XIX REUNIÓN ESPAÑOLA SOBRE CRIPTOLOGÍA Y SEGURIDAD DE LA INFORMACIÓN



Tenerife, 18-20 Marzo 2026

Editores:

Marcos Rodriguez-Vega

Daniel Escanez-Exposito

Jorge Garcia-Diaz

Pino Caballero-Gil

Financiado por la Unión Europea NextGenerationEU

GOBIERNO DE ESPAÑA · MINISTERIO PARA LA TRANSFORMACIÓN DIGITAL Y DE LA FUNCIÓN PÚBLICA · SECRETARÍA DE ESTADO DE TELECOMUNICACIONES E INFRAESTRUCTURAS DIGITALES

Plan de Recuperación, Transformación y Resiliencia

incibe_ INSTITUTO NACIONAL DE CIBERSEGURIDAD

XIX REUNIÓN ESPAÑOLA SOBRE CRIPTOLOGÍA Y SEGURIDAD DE LA INFORMACIÓN



Tenerife, 18-20 Marzo 2026

Editores:

Marcos Rodriguez-Vega

Daniel Escanez-Exposito

Jorge Garcia-Diaz

Pino Caballero-Gil

Financiado por la Unión Europea NextGenerationEU

GOBIERNO DE ESPAÑA · MINISTERIO PARA LA TRANSFORMACIÓN DIGITAL Y DE LA FUNCIÓN PÚBLICA · SECRETARÍA DE ESTADO DE TELECOMUNICACIONES E INFRAESTRUCTURAS DIGITALES

Plan de Recuperación, Transformación y Resiliencia

incibe_ INSTITUTO NACIONAL DE CIBERSEGURIDAD

Reunión Española sobre Criptología y Seguridad de la Información (19ª. 2026. Tenerife, España)

XIX Reunión Española sobre Criptología y Seguridad de la Información: XIX RECSI, Tenerife 18-20 Marzo 2026 / Editores: Marcos Rodriguez-Vega, Daniel Escanez-Exposito, Jorge Garcia-Diaz, Pino Caballero-Gil – Tenerife: Universidad de La Laguna, Servicio de Publicaciones, 2026. 1 recurso en línea (XX p.)
ISBN XXX-XX-XXXXX-XX-X

Edita: UNIVERSIDAD DE LA LAGUNA. Servicio de Publicaciones

Imagen de la portada: "Armas Núñez, J. (2014). Luz e icono : la vidriera artística en las iglesias canarias. RIULL. Repositorio institucional. San Cristóbal de La Laguna: Servicio de publicaciones de la Universidad de La Laguna, p. 54."



CC BY-NC-ND 4.0

Atribución No Comercial Sin Derivadas 4.0 Internacional

Usted es libre de:

Compartir, copiar y redistribuir el material en cualquier medio o formato bajo los siguientes términos:

Atribución: Usted debe dar crédito de manera adecuada, brindar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.

No Comercial: Usted no puede hacer uso del material con propósitos comerciales.

Sin Derivadas: Si remezcla, transforma o crea a partir del material, no podrá distribuir el material modificado.

Tenerife, 2026

Índice de Artículos

AI-Driven Privacy Risk Analysis for Mobile Apps and Social Networks
Daniel Blanco-Aza, Antonio Robles-Gómez, Rafael Pastor-Vargas, Llanos Tobarra, Pedro Vidal-Balboa and Mariano Méndez-Suárez **11**

Marco para la adquisición remota y automatizada de evidencias digitales en entornos corporativos e industriales
David García Fernández, Llanos Tobarra, Antonio Robles-Gómez and Rafael Pastor-Vargas **15**

Quasi-perfect (de)compression of elliptic curve points in the highly 2-adic scenario
Dimitri Koshelev **21**

PaXport: Privacy Pitfalls in the Spanish Age Verification System
David Balbás and Diego Castejón-Molina **23**

Smart Rural IoT Lab: Laboratorio experimental para la evaluación de la ciberseguridad en entornos rurales inteligentes
Daniel Sernández-Iglesias, Llanos Tobarra Abad, Rafael Pastor-Vargas, Antonio Robles-Gómez, Pedro Vidal-Balboa and João Sarraipa **29**

About Formal Security Models for the Post-Quantum ML-KEM
Inés Araujo Cañas and Luis Hernández Encinas **34**

Building Blocks Descentralizados del Pasaporte Digital de Producto para Mejorar la Soberanía de la Cadena de Suministro y las Prácticas de Economía Circular
Santiago de Diego and Ivan Gutiérrez **40**

Detección de Buffer Overflow utilizando Lógica difusa y Machine Learning
Alfonso Gotor Burguete, Bruce Azevedo, Robson Albuquerque and Luis Villalba **44**

Generación mejorada de ruido para oráculos de descifrado de la cifra Kyber
Francesc Sebé **54**

Metodología de Auditoría de Juguetes
Diego Narciandi-Rodríguez, Jose Aveleira-Mata, Pablo de la Hera Martínez, Iván Gómez Muñoz, Jonatan García González and Miguel Martín Redondo **59**

TrustAIoT: de la confiabilidad sociotécnica a la detección temprana ecosistémica
Carlos Mario Braga Ortuño, Manuel A. Serrano and Eduardo Fernandez-Medina . . . **65**

DP to BDP: Noise Recalibration for Correlation-Resilient Privacy Guarantees
Patricia Guerra-Balboa, Martin Lange, Javier Parra-Arnau and Thorsten Strufe . . . **70**

Índice de Artículos

AI-Driven Privacy Risk Analysis for Mobile Apps and Social Networks
Daniel Blanco-Aza, Antonio Robles-Gómez, Rafael Pastor-Vargas, Llanos Tobarra, Pedro Vidal-Balboa and Mariano Méndez-Suárez **11**

Marco para la adquisición remota y automatizada de evidencias digitales en entornos corporativos e industriales
David García Fernández, Llanos Tobarra, Antonio Robles-Gómez and Rafael Pastor-Vargas **15**

Quasi-perfect (de)compression of elliptic curve points in the highly 2-adic scenario
Dimitri Koshelev **21**

PaXport: Privacy Pitfalls in the Spanish Age Verification System
David Balbás and Diego Castejón-Molina **23**

Smart Rural IoT Lab: Laboratorio experimental para la evaluación de la ciberseguridad en entornos rurales inteligentes
Daniel Sernández-Iglesias, Llanos Tobarra Abad, Rafael Pastor-Vargas, Antonio Robles-Gómez, Pedro Vidal-Balboa and João Sarraipa **29**

About Formal Security Models for the Post-Quantum ML-KEM
Inés Araujo Cañas and Luis Hernández Encinas **34**

Building Blocks Descentralizados del Pasaporte Digital de Producto para Mejorar la Soberanía de la Cadena de Suministro y las Prácticas de Economía Circular
Santiago de Diego and Ivan Gutiérrez **40**

Detección de Buffer Overflow utilizando Lógica difusa y Machine Learning
Alfonso Gotor Burguete, Bruce Azevedo, Robson Albuquerque and Luis Villalba **44**

Generación mejorada de ruido para oráculos de descifrado de la cifra Kyber
Francesc Sebé **54**

Metodología de Auditoría de Juguetes
Diego Narciandi-Rodríguez, Jose Aveleira-Mata, Pablo de la Hera Martínez, Iván Gómez Muñoz, Jonatan García González and Miguel Martín Redondo **59**

TrustAIoT: de la confiabilidad sociotécnica a la detección temprana ecosistémica
Carlos Mario Braga Ortuño, Manuel A. Serrano and Eduardo Fernandez-Medina . . . **65**

DP to BDP: Noise Recalibration for Correlation-Resilient Privacy Guarantees
Patricia Guerra-Balboa, Martin Lange, Javier Parra-Arnau and Thorsten Strufe . . . **70**

Comparadores Lexicográficos en el Modelo R1CS
Marc Guzmán-Albiol, Marta Bellés-Muñoz, Carla Brugulat-Rica, Rafael Genés-Durán and Jose L. Muñoz-Tapia **74**

Un resumen de '"Meta-feature-based Data Preprocessing for
Noemí DeCastro-García and David Escudero García **78**

Existence conditions of codes for content protection in medical environments
Marcel Fernandez, Sebastià Martín Molleví, Daniel Escanez-Exposito and Pino Caballero **82**

On the Inconsistency of Linear Complexity Tests: a Surprising Case Study in the Rabbit Battery-TestU01
Jorge Munilla, Darren Hurely-Smith and Julio Hernandez-Castro **88**

Identifying Cybersecurity Vulnerabilities in Water and Resource Recovery Facilities (WRRFs)
Sandra Bonillo **94**

Auditoría de Seguridad en Dispositivos IoT BLE: Técnicas de Sniffing Pasivo y Ataques GATT
Jose Aveleira-Mata, Diego Narciandi-Rodríguez, Martín Bayón-Gutiérrez, Natalia Prieto-Fernández, Isaías Garcia-Rodríguez and Carmen Benavides **97**

Detección de Imágenes Generadas por Modelos de Difusión usando Información Espectral y Semántica mediante Mecanismos de Atención
Daniel Povedano Álvarez, Andrés Sáanchez Casas, Alexandra Carolina Molina Ospina and Luis Javier García Villalba **102**

Un Marco Multicapa de Ciberengaño con IA y Orquestación Zero-touch Multidominio
Pedro Beltrán-López, Pantaleone Nespoli and Manuel Gil Pérez **108**

Microagregación híbrida asistida por modelos de lenguaje
Sara Lanuza-Orna and Alberto Blanco-Justicia **112**

Patient-centric drug monitoring with certified awareness: a blockchain-enabled IoT platform
Jaume Ramis-Bibiloni, M. Magdalena Payeras-Capellà, Bartomeu Alorda-Ladaria and Macià Mut-Puigserver **118**

Seguridad del paciente en IoMT: Una revisión sistemática de la literatura sobre IDS basados en el comportamiento
Jordi Doménech, Isabel V. Martin-Faus, Saber Mhiri and Josep Pegueroles **122**

Further Development of the Toolbox for the Design of Non-Binary Pseudorandom Sequence Generators
Slobodan Petrovic **126**

NOVADEF: Una plataforma de monitorización de seguridad ubicua para actividades de perfilado de atacantes basadas en el engaño
Alberto García Pérez, Pedro Beltrán-López, José María Jorquera Valero, Antonio López Martínez, Manuel Gil Pérez and Gregorio Martínez Pérez **130**

Generadores Pseudoaleatorios sobre F_{p^n} con Realimentacion Dinamica Inducida por Curvas Elipticas
Guillermo Cotrina Cuenca and Alberto Peinado **136**

Cibernética inversa: detección estructural de ciberataques por ruptura de simetría ecosistémica
Carlos Mario Braga Ortuño, Manuel A. Serrano and Eduardo Fernandez-Medina . . . **142**

Modeling and Training Robust Intrusion Detection Systems Using Deep Neural Networks: A Synthesis of 'Best Practices'
Victor Manuel González-Gorrín and Josep Prieto-Blázquez **148**

Maritime cybersecurity: protecting digital seas
Ferney Martinez Ossa, Luis Enrique Sánchez, Antonio Santos-Olmo, David G.Rosado and Eduardo Fernández-Medina **154**

When Do Membership Inference Attacks Succeed? An Empirical Study of Overfitting in Fine-Tuned Large Language Models (LLMs)
Jean Dié, Najeeb Jebreel, David Sánchez and Josep Domingo-Ferrer **158**

Mejora de la anonimización de texto mediante explicabilidad orientada al riesgo de re-identificación
Benet Manzanares-Salor and David Sanchez **164**

Predicción de Carga Segura y Privada en Smart Grids mediante Incentivos Basados en Blockchain
Joan Ferré-Queralt, Jordi Castellà-Roca and Alexandre Viejo **168**

Resumen extendido de "Calyptography: Secure Secret Storage Inspired by Cryptography and Steganography"
Daniel Lerch-Hostalot, Jordi Puiggali and David Megías **172**

Detección de Contenido Sexual en Dispositivos Móviles para la Preservación de la Privacidad
Sandra Pérez Arteaga, Ana Lucila Sandoval Orozco and Luis Javier García Villalba . . **176**

Viabilidad de un intercambio de clave ML-KEM autenticado en entornos IoT mediante LoRa
Rodolfo Leonardo Sumoza Matos, Esteban Eduardo Mocskos and Raúl Durán Díaz . . **182**

Multi-Solution Grover Search on Small Reversible Hash Functions: Simulation and Noisy Quantum Hardware Experiments
Jon Hernández-Bueno, Julen Bernabé-Rodriguez **188**

Inventarios criptográficos para una migración postcuántica ágil y segura
Juan Banga Pardo, Santiago de Vicente Poutás and Xavier Martinez Luaña **194**

The impact of aggregation algorithms on property inference attacks in federated learning
Adolf Felip Garcia, Antonina Serebriakova, Gorka Pineda Burgueño, Cristina Pérez-Solà and Victor Garcia-Font **200**

QUILLAQUA: Arquitectura por capas y Huella Digital de Seguridad en IoT agrícola
Diego Bustamante, Antonio Santos-Olmo, David G. Rosado, Luis E Sánchez and Eduardo Fernández-Medina **206**

Extendiendo X-Road con Validación de Código de Metaservicios: Una Contribución Orientada a la Seguridad
Romualdo Victor Figueroa, Luis Enrique Sánchez Crespo, Antonio Santos-Olmo, David G. Rosado and Eduardo Fernández-Medina **213**

Estudio comparativo del rendimiento de los estándares de firma digital post-cuánticos
Pablo Aguado Gómez and Luis Hernández **220**

FAIR+: Protocolo Federado, Anónimo e Interoperable para Resoluciones colectivas
Macià Mut-Puigserver, Magdalena Payeras-Capellà, Rosa Pericàs Gornals, Jaume Ramis-Bibiloni and Jordi Castellà-Roca **226**

Diseño y Desarrollo de un Sistema de Respuesta a Ciberamenazas Basado en Large Language Models y Retrieval Augmented Generation
Rodrigo Tavares de Pina Simões, Luis Pérez Miguel, Xavier Larriva-Novo, David Álvarez Muñiz, Carmen Sanchez-Zas and Victor Villagrá **232**

A Side-Channel Protected and High-Performance Hardware Implementation for EdDSA25519
Pablo Navarro-Torrero, Eros Camacho-Ruiz, Macarena C. Martínez-Rodríguez and Piedad Brox **238**

NoRDEx: Un protocolo descentralizado optimista de no repudio para intercambios de datos
Fernando Román-García, Antonio Alarcón, Juan Hernández-Serrano and Oscar Esparza **242**

Nonlinear Approximations in Linear Cryptanalysis and Hadamard matrices
Jose Andres Armario **246**

¿Sobremuestreo o submuestreo? Cómo mejorar las métricas de eficiencia en la detección de amenazas basada en aprendizaje automático
Jordi Doménech, Olga León, Isabel V. Martin-Faus, Enrica Zola and Josep Pegueroles **251**

Quantum Entanglement and Communication
Guillermo Morales-Luna **257**

Statistic measures in LLM Watermark detection
Jose Luis Martin-Navarro **263**

An extended approach towards root-based attacks against PLWE instances via trace operators
Beatriz Barbero-Lucas, Iván Blanco-Chacón, Raúl Durán-Díaz, Rodrigo Martín Sánchez-Ledesma and Rahinatou Njah Nchiwo **268**

Anticipando las amenazas cibernéticas: un marco de preparación en cuatro pasos
Jeimy Cano **272**

EIP 712-based Off-Chain Oracles: Benefits and Caveats
Daniel Morales and Isaac Agudo **279**

DiReCT Voting: Dispute Resolution with Cast Timeliness
Jose Luis Martin-Navarro, Antonio Manuel Larriba Flor and Damián López **284**

TRITON: Hacia un Marco de Ciberseguridad para Redes de Agua Inteligentes
Isaac David Torres Paredes, Antonio Santos-Olmo, David G Rosado, Luis Enrique Sánchez and Eduardo Fernández-Medina **288**

The Sponge Mesozoic: The Explosion in Sponge-Based Constructions
Mario Marhuenda Beltran **294**

Privacidad online: evaluando la incertidumbre del adversario en técnicas de perturbación de perfiles
César Gil, Javier Parra-Arnau and Jordi Forné **298**

Resumen Extendido: Rewardable Naysayer Proofs
Gennaro Avitabile, Luisa Siniscalchi and Ivan Vinsconti **304**

De reseñas a reputación: un textit{pipeline} auditable, explicable y robusto para servicios en línea
Elmira Saeedi Taleghani, Ronald Ivan Maldonado Valencia, Ana Lucila Sandoval Orozco and Luis Javier García Villalba **309**

Metodología de optimización para la selección de contramedidas en procesos de gestión de riesgos de ciberseguridad
Carmen Sanchez-Zas, Javier Montesinos Martí, Xavier Larriva-Novo, Sonia Solera Cotanilla and Victor Villagrá **315**

HoneyV: Laboratorio Automatizado de Análisis de Malware Basado en Honeypots
Antonio Lara-Gutierrez, Jose A. Onieva and Carmen Fernandez-Gago **321**

Algebraic Zero Knowledge Contingent Payment
Javier Gomez-Martinez, Dimitrios Vasilopoulos, Pedro Moreno-Sanchez and Dario Fiore **327**

Diseño y desarrollo de un sistema predictivo de gestión de amenazas de seguridad utilizando CWE, CVE y CAPEC
Joaquin Sierra Granados, José Luis Ruiz-Catalán, Ferney Martínez Ossa, Ángel Suárez-Bárcena, Antonio Santos-Olmo and Luis Enrique Sánchez **331**

Towards a European Data and AI Ecosystem: Convergences between the DGA, DA, AIA, and GDPR
Ignacio Miguel Rodríguez, Jordi Castellà Roca, M. Magdalena Payeras Capellà and Alexandre Viejo **337**

Fusión Audio-Visual con Mecanismos de Atención para la Detección de Contenido Sensible en Vídeo
Daniel Povedano Álvarez, Ana Lucila Sandoval Orozco and Luis Javier García Villalba **343**

Microagregación con Privacidad Diferencial: Ruido Adaptativo a Nivel de Clúster
Jordi Soria-Comas, David Megías, Jordi Castellà-Roca, Carles Anglés-Tafalla, Guillermo Navarro-Arribas and Julián Salas **349**

Generación de Código Malicioso con IA Generativa y Modelos de Lenguaje
Luis Alberto Martinez Hernandez, Sandra Pérez Arteaga, Ana Lucila Sandoval Orozco and Luis Javier García Villalba **355**

Random Graph Models for the Lightning Network: A Comparative Topology Study
Yasser Hayali, Cristina Pérez-Solà and Jordi Herrera **361**

Revision de tácticas, técnicas y procedimientos (TTPs) en MITRE ATT&CK para Compromiso del Correo Empresarial (BEC)
Sebastien Kanj, Olga León, Enrica Zola, Oriol Rosés and Josep Pegueroles **366**

Topological Safeguard for Evasion Attack based on the Interpretability of Artificial Neural Network Behavior (Extended Abstract)
Xabier Echeberria Barrio **370**

Is Hardware Wallet Fingerprinting Even Possible?
Domènec Madrid Hernández and Jordi Herrera Joancomartí **374**

Explorando mejoras en el protocolo Albatross: una validación experimental en escenarios distribuidos
Cristian González García, Ana I. Gómez, Jesús R. Abascal and Domingo Gomez . . . **380**

Desarrollo Seguro de Software en BizDevOps: Incorporación Temprana de Metas y Controles
Alejandra Selva-Mora, Christian Quesada-López, Antonio Santos-Olmo, Luis Enrique Sánchez and Eduardo Fernández-Medina **386**

PINO: Privacidad en Incidencias Notificadas On-Chain
Carles Anglés-Tafalla, Jordi Castellà-Roca, Alexandre Viejo, Jordi Soria-Comas, Guillermo Navarro-Arribas and Julián Salas **392**

ZWP-TRUST: Trusted Utility-Smart Telemetry Using Zero Knowledge Proof, Reversible Watermarking, and Paillier Encryption
Tanya Koohpayeh Araghi, Muhammad Yasir Masood, Helena Rifà-Pous, David Megías, Joan Ferré-Queralt, Jordi Castellà-Roca and Alexandre Viejo **398**

Diseño y desarrollo de un sistema de control de acceso y presencia en eventos mediante técnicas de reconocimiento facial y OCR
Sonia Solera-Cotanilla, Javier Martínez, Manuel Álvarez-Campana, Carmen Sánchez-Zas and Xavier Larriva-Novo **405**

Helm-ET: Reducing Exposure to Lateral Movement in Kubernetes Artifacts
Jose Luis Martin-Navarro **411**

Inteligencia Artificial aplicada al análisis de códigos convolucionales y sus representaciones ISO
Lucía Mallo Fernández, David Escudero García, Noemí DeCastro-García and Ángel Luis Muñoz Castañeda **417**

Reimagining Proof-of-Work with Elliptic Curve Discrete Logarithms
Ghazaleh Keshavarzkalhori, Biel Castellarnau, Jordi Herrera-Joancomarti and Cristina Pérez-Solà **423**

M3Suite: Plataforma de Aprendizaje Federado Privado, Modular y Usable
Idoia Gamiz, Julen Bernabé, Kristina Stemikovskaya and Miguel Esteras **429**

Quantum Sieving to solve the LWE Problem
Diego Rojas Rodríguez, Miguel Ángel González de la Torre and Luis Hernández Encinas **435**

Análisis de FAEST como Protocolo de Firma Digital Postcuántica
Fernando Contreras, Miguel Ángel González de la Torre and Luis Hernández Álvarez . **441**

DataSecOnt: Ontología para la Seguridad de los Datos de Google Play
Alejandra Gavino-Dias, M. Mercedes Martínez-González and Alejandro Pérez-Fuente . **447**

FHE en Detección de Anomalías: Aplicando Cifrado a la Evaluación de Modelos de IA
Julen Bernabé-Rodríguez, M. Carmen Palacios-Prados, Marina Ruiz del Olmo and Maite Álvarez Piernavieja **453**

Integridad y auditoría para DNS: auditabilidad a largo plazo para IoT y redes distribuidas
Daniel Díaz-Sánchez, Florina Almenares and Carlos García-Rubio **460**

Un protocolo basado en blockchain para la entrega equitativa de mensajes con acuse de recibo
Gerardo-Francisco Chévez-Alvarado, M. Francisca Hinarejos, Josep-Lluis Ferrer-Gomila and Andreu-Pere Isern-Deyà **464**

Tweak Meets Hashing: Improving Padding-Free Sponges
Mario Marhuenda Beltran **468**

Un marco de trabajo para la trazabilidad y el control de riesgos en la ciberseguridad de la cadena de suministro de software
Carlos Villarrubia, Antonio Santos-Olmo, Juan Manuel Vara and Eduardo Fernández-Medina **473**

Providing economic incentives to reveal the existence of a CRQC
Roger Sala-Mimó and Jordi Herrera-Joancomartí **479**

Análisis de mecanismos de privacidad basados en la perturbaci´on de los enlaces en grafos
Víctor Rubio-Jornet, Javier Parra-Arnau and Jordi Forné **485**

Evaluación comparativa de esquemas ZKP para credenciales anónimas en clave eIDAS2.0
Alfonso Camblor-García, Xavier Martínez-Luaña, Manuel Fernández-Veiga and Pablo Santos-Cabaleiro **491**

Verificaciones de perfiles digitales: construcción de perfiles en entornos online mediante OSINT
Antonio Calvo Picón, Luis Jesús Arévalo Rosado and José Carlos Sancho Núñez . . . **498**

Niños y adultos mayores ante la vulnerabilidad digital: la urgencia de programas de capacitación para su protección
José Agustín Portas Yáñez, Ana Lucila Sandoval Orozco and Luis Javier García Villalba **504**

Interpretable Neuro-Symbolic Graph Neural Network for Anomaly Detection
Fernando Ares-Robledo, Helena Rifà-Pous and Robert Clarisó **510**

A Complete Description of the Hermitian Dual of Dihedral Codes and Derived Quantum Codes
Miguel Sales Cabrera, Xaro Soler-Escrivà and Víctor Sotomayor **516**

Rompiendo Barreras en el Entorno Médico: un Framework Seguro Basado en Identidad para el Control de Acceso
Antonio López Martínez, Montassar Naghmouchi, Maryline Laurent, Joaquín García Alfaro, Manuel Gil Pérez and Antonio Ruiz Martínez **521**

Estudio de la ciclicidad de códigos grupo binarios sobre grupos no abelianos de orden menor que 24
Beatriz García García, Consuelo Martínez López and Ignacio F. Rúa **525**

A Taxonomy and Security Analysis of Rollup Architectures in Ethereum Blockchain
Adrià Torralba-Agell, Muoi Tran and Cristina Pérez-Solà **529**

Secuencias binomiales sobre cuerpos primos
Miguel Beltrá Vidal, Sara Díaz Cardell, Verónica Requena Arévalo and Joan Josep Climent **535**

Migration Strategies for Ethereum Externally Owned Accounts under the Quantum Threat
Marco López and Isaac Agudo **541**

Uso de LLMs para el análisis de vulnerabilidades en Solana "Programs""""
Laura De Haro García and Isaac Agudo **547**

Hacia nuevas formas de componer aleatoriamente en privacidad diferencial
Àlex Miranda-Pascual, Javier Parra Arnau and Thorsten Strufe **553**

Códigos Grupo LDOI con Matriz de Adyacencia Binaria
Fabián Molina and Consuelo Martínez **559**

Hacia un indicador de privacidad para aplicaciones móviles basado en OWASP MAS
Amador Aparicio, M. Mercedes Martínez-González, Alejandro Pérez-Fuente and Pablo A. Criado-Lozano **565**

Towards an integrated risk analysis security framework according to a systematic analysis of existing proposals
Antonio Santos-Olmo, Luis Enrique Sanchez, David G. Rosado, Carlos Blanco Bueno, Haralambos Mouratidis and Eduardo Fernandez-Medina **571**

Security analysis of multilevel secret sharing schemes based on autoparatopisms of Latin squares
Manuel González-Regadera, Raúl Falcón and María Dolores Frau **575**

Slice Property Inference in B5G Networks
Albert Garcia-Llagostera, Helena Rifà-Pous and Julián Salas **580**

Comunicación segura de vehículos usando criptografía basada en identidades
Pablo Escapa Gordón, Vicente Matellán Olivera and Adriana Suárez Corona **586**

Evaluación de riesgos en el procesamiento automatizado de Documentos Nacionales de Identidad
José Carlos Sancho Núñez, Antonio Calvo Picón, Sergio Guijarro Domínguez and Víctor González Morales **590**

EVALUACIÓN COMPARATIVA DE MODELOS DE APRENDIZAJE AUTOMÁTICO UTILIZADOS PARA LA DETECCIÓN DE ANOMALÍAS EN LOGS DE ACTIVE DIRECTORY
Allysson Noronha and Robson Albuquerque **596**

VESTAL: Marco holístico de ciberseguridad personal
Angel Suarez-Barcena, José Luis Ruiz Catalán, Antonio Santos-Olmo Parra and Eduardo Fernandez-Medina **602**

Análisis de Vulnerabilidades en Protocolo de Comunicación DNP 3.0 en Sistemas ICS/SCADA de Infraestructuras Críticas Eléctricas
Jacinto Pérez García, Ana Lucila Sandoval Orozco and Luis Javier García Villalba . . **609**

Optimización del test de primalidad para la generación de claves de firmas en anillo
Álvaro Larrey-Gómez, José Luis Salazar Riaño and Guillermo Azuara Guillén **615**

Mejoras para el cifrado broadcast para mensajes cortos en WiFi
José Luis Salazar-Riaño, Julián Fernández-Navajas and Guillermo Azuara-Guillén . . **621**

A Linguistic Intelligence Framework for Proactive Ransomware Detection
Muhammad Junaid Iqbal and Jordi Serra-Ruiz **625**

Transformer borgeanos: inyectividad y confidencialidad en modelos de lenguaje
Mariano Caruso **631**

QZKP basada en Codificacion Conjugada para la ´
Jorge Garcia-Diaz, Daniel Escanez-Exposito, Pino Caballero-Gil **636**

AI-Driven Privacy Risk Analysis for Mobile Apps and Social Networks

Daniel Blanco-Aza*
Universidad Nacional de Educación a Distancia (UNED)
Programa de Doctorado en Sistemas Inteligentes
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
d.blanco@scc.uned.es

Antonio Robles-Gómez
Universidad Nacional de Educación a Distancia (UNED)
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
arobles@scc.uned.es

Rafael Pastor-Vargas
Universidad Nacional de Educación a Distancia (UNED)
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
rpastor@scc.uned.es

Llanos Tobarra
Universidad Nacional de Educación a Distancia (UNED)
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
llanos@scc.uned.es

Pedro Vidal-Balboa
Consorcio CA UNED en Ponferrada / INTECCA
Ponferrada, Spain
pvidal@intecca.uned.es

Mariano Méndez-Suárez
ESIC University, ESIC Business & Marketing School
Madrid, Spain
mariano.mendez@esic.university

***Abstract*—Mobile apps and social networks collect and process large volumes of personal data, raising persistent privacy concerns. We present *SafeMountain*, an AI-driven framework that unifies static and dynamic analysis with NLP-based interpretation of privacy policies and Terms of Service to quantify privacy risk. The pipeline correlates requested permissions, third-party libraries, and code/runtime behaviors with textual disclosures to detect inconsistencies, signs of non-compliance, and potential leakage. In prior work [1], we introduced the framework architecture, a multilingual (EN/ES) corpus for policy analysis, and a reproducible traffic-light risk score aligned with regulatory requirements. We also conducted illustrative evaluations on popular social applications, showing the framework's ability to surface permission–policy mismatches, high-risk permission usage, and third-party components with privacy impact. Here, we provide an overview of the approach and summarize the obtained results, including dataset curation, analysis modules, and scoring rules, and we outline current limitations and ongoing extensions.**

***Index Terms*—SafeMountain, Mobile Apps, AI Techniques, PRISMA Methodology, Privacy Risk Framework, Social Networks Privacy**

I. Introduction

Mobile applications and social networks collect, process, and share large amounts of personal data. This raises persistent privacy concerns for users and complicates how developers and organizations demonstrate compliance with data-protection regulations. Prior research provides valuable but specialized tools (e.g., checking description–permission consistency in Android apps, contrasting privacy policies with platform labels, inspecting third-party libraries (TPLs), or assessing permissions and privacy risks via static/dynamic analyses) yet these approaches are rarely integrated into a unified, objective way to quantify privacy risk across technical and textual evidence. As a result, practitioners still lack reproducible assessments that can be applied at scale to heterogeneous apps and multilingual policy documents.

In this work, we summarize the results obtained with *SafeMountain*, an AI-driven framework that integrates static and dynamic analysis with NLP-based understanding of privacy policies and Terms of Service to quantify privacy risk in mobile apps and social platforms [1]. The framework correlates requested permissions, TPLs, and observed code/runtime behaviors with multilingual policy disclosures to identify inconsistencies, indicators of non-compliance, and potential leakage. We released a bilingual (EN/ES) corpus to support policy analysis and implementation artifacts to facilitate reproducibility [2], [3]. Illustrative evaluations on popular social applications showed the framework's ability to surface permission–policy mismatches, high-risk permission usage, and TPLs with privacy impact.

The remainder of this paper is organized as follows: Section II reviews related work and positions our approach; Section III summarizes the research methodology and datasets; Section IV presents the risk assessment model and scoring process; Section V describes the SafeMountain framework; Section VI summarizes experiments and findings; and Section VII concludes and outlines future work.

II. Related Work

Research on privacy analysis for mobile apps and social platforms spans several complementary strands, typically focusing on either textual disclosures (policies, labels) or technical evidence (permissions, code and network behavior). On the textual side, *AC-Net* infers Android permissions from app descriptions, highlighting description–permission consistency gaps [4]. *ATLAS* targets inconsistencies between iOS App Store privacy labels and the privacy policies inferred via NLP, surfacing systematic mismatches but without observing runtime behavior [5]. Beyond labels, methods such as *PermPress* [6] and *PolicyChecker* [7] analyze privacy policies against regulatory completeness or policy–permission alignment, while *PolicyLint* [8] detects internal contradictions in policies.

On the technical side, several tools inspect code, libraries, and runtime data handling. *ATPChecker* [9] evaluates TPLs by combining static analysis and NLP to detect discrepancies between declared practices and binary behavior. *PRADroid* [10] estimates privacy risk for Android by combining permission analysis with static information-flow reasoning. *PriVot* [11] joins policy classification with TLS interception to compare stated disclosures against observed traffic, though it faces challenges with certificate pinning and policy ambiguity. *XPOChecker* [12] studies cross-user over-delivery (XPO) risks via bi-directional slicing, targeted test generation, and dynamic tracing, quantifying severity/likelihood of over-sharing events. Complementary lines include active-learning for leak detection on iOS (*iABC-AL*) [13] and broader corpora/resources for policy NLP [14].

Prior work typically targets a single evidence channel (text-only or code/network-only) and lacks a unified, reproducible quantification that bridges technical behaviors with policy disclosures across languages. In our previous publication [15], we reported a comprehensive survey and gap analysis of these tool families and their

*Corresponding author.

limitations, motivating a multi-evidence approach. Building on that, our prior study introduced *SafeMountain* as an integrated framework that correlates permissions, TPLs, and static/dynamic findings with multilingual (EN/ES) policy/ToS analysis, supports dataset curation and evidence alignment, and produces an objective traffic-light risk score that summarizes non-compliance indicators.

SafeMountain further adds obfuscation-resilient signals by combining APK public-string mining aligned with PII taxonomies with robust attribution of TPLs directly from APKs, even under repackaging or identifier renaming. These additions mitigate decompilation limits and naming erasure, improve attribution between host and library code, and reinforce the cross-validation of textual disclosures against technical evidence.

III. Research Methodology and Datasets

A. Overview

We follow a structured, reproducible methodology with six phases: (P1) systematic state-of-the-art review (PRISMA); (P2) dataset generation and multilingual corpus compilation; (P3) design and development of the modular *SafeMountain* framework; (P4) evaluation and comparative analysis; (P5) experimentation on real-world social applications; and (P6) deployment and continuous validation.

B. Dataset Construction (P2)

For training and evaluation we compiled (i) Android APKs from social-media–related categories selected by popularity in major app marketplaces, (ii) associated privacy policies and Terms of Service, and (iii) app/store metadata. Collection was automated via scripts (scraping/APIs); texts were preprocessed to build bilingual corpora (EN/ES) with manual annotations supporting compliance-oriented assessments.

C. Analysis Pipeline (P3)

SafeMountain is designed according to privacy–engineering principles articulated in NISTIR 8062 [16]. The pipeline integrates, end to end, data collection and normalization of app artifacts and metadata; static analysis of APKs to extract permissions, sensitive APIs, and code/manifest indicators (using decompilation outputs to drive rule-based detectors and dynamic analysis to observe runtime and network behaviour in complementary fashion [17]). In parallel, third-party library (TPL) attribution is performed with strategies resilient to obfuscation and informed by scalable identification methods [18]. Textual evidence is incorporated through *DeepMountain*, an EN/ES policy/ToS NLP component that segments and classifies disclosures, yielding coverage signals and evidence spans suitable for assessment. Finally, cross-channel consistency checks align what the policy states with what the app is capable of or does at runtime, and contrast host disclosures with those implicit in embedded libraries, drawing on prior analyses of policy–behavior and policy–label mismatches [9].

D. Evaluation Protocol (P4–P5)

We combine quantitative and qualitative analyses, plus an application study on popular social apps. Quantitatively, we evaluate classifiers and policy–behaviour consistency detectors on benchmark datasets from Phase 2 using standard metrics (accuracy, precision, recall, F1, ROC–AUC) and compare against representative tools from Phase 1 to gauge coverage, multilingual robustness, and coherence of the unified scoring. Qualitatively, expert/user reviews assess interpretability and utility, and the feedback guides iterative refinement of models, thresholds, and reports. Finally, executing the pipeline end-to-end on widely used social applications validates real-world behaviour and surfaces permission–policy mismatches, privacy-relevant TPLs, and potential leakage signals.

Table I
Risk level determination based on probability and impact.

Impact / *Prob.*	*Very limited*	*Limited*	*Significant*	*Very significant*
Very high	Medium	High	Very high	Very high
High	Low	High	Very high	Very high
Low	Low	Medium	High	Very high
Improbable	Low	Low	Medium	Very high

E. Deployment and Continuous Validation (P6)

We plan staged release of corpus and code, with periodic refresh of app/policy snapshots and regression tests. User feedback and regulatory updates drive iterative rule tuning (thresholds/weights), language expansion, and resilience improvements in dynamic observation and TPL attribution. This enables longitudinal tracking of privacy signals and reproducible re-evaluation under versioned assets.

IV. Risk Assessment Model and Scoring Process

Ensuring privacy protection in mobile applications requires a structured, objective, and reproducible assessment. Our approach is grounded in the privacy–engineering principles of NISTIR 8062 [16] and the risk–impact guidance of the Spanish Data Protection Agency (AEPD) [19], aligning with the General Data Protection Regulation (GDPR) [20] and retaining flexibility to accommodate regulatory frameworks such as the Digital Services Act (DSA) [21].

The assessment relies on a characterization dataset comprising EN/ES privacy policies and terms of service, together with app permissions and metadata. This corpus feeds NLP models for classification and compliance analysis, following prior work that leverages annotated corpora to link GDPR obligations with permissions declared in code [14]. Based on these correlations (declared practices and rights versus declared capabilities), the method estimates alignment with regulatory expectations and the potential level of risk, incorporating GDPR/AEPD risk factors to evaluate both *probability* and *impact*.

Consistent with AEPD guidance, we distinguish *inherent* from *residual* risk and consider contextual origins (legal obligations, data nature, technological complexity, and affected rights/freedoms). Compliance documentation covers data type/sensitivity, processing scale/purposes (including retention and geographic scope), security/governance measures, and operational context. Each factor is rated by *impact* (very significant, significant, limited, very limited) and *probability* (very high, high, low, improbable); when multiple indicators converge, an accumulated coefficient may be applied. Overall levels follow Table I.

For practical traceability, risk factors are grouped along GDPR–aligned categories: processing operations; data types; extent and scope; data–subject categories (e.g., vulnerable groups); technical aspects (e.g., encryption, access control); data collection and generation (accuracy, consent, sources); side effects; controller/processor category; data transfers and sharing; and security breaches.

Risk is operationalized with a traffic–light system (Green, Yellow, Red) per dimension. The dimensions follow NISTIR 8062 objectives (*Predictability*, *Manageability*, and *Disassociability*) and are evaluated from observable characteristics in policies/ToS, permissions, metadata, and, when applicable, runtime behavior. The global score is assigned conservatively: if any dimension is Red, the overall score is Red; if all are Green, the overall score is Green; otherwise, Yellow. When available, historical breach reports, known attack vectors, and publicly disclosed vulnerabilities are incorporated to reinforce probability estimation.

By systematically combining standards, regulatory factors, and observable evidence, the framework yields a quantifiable and contextualized privacy–risk assessment that supports accountability and transparency for users and authorities.

V. SafeMountain Framework

Building on the risk-assessment methodology, *SafeMountain* delivers a unified, modular pipeline to produce automated, standards-aligned privacy diagnostics for mobile and web apps (especially social networks). The toolchain combines existing resources and proprietary modules: *Libloom* [18], *Privado Scan* [22], *MobSF* [17], and *ATPChecker* [9], orchestrated end-to-end.

A data-collection component (*AppCollector*) automates marketplace harvesting, storing app/library metadata, APKs and versioned binaries in a structured workspace. For TPL attribution, *Libloom* builds profiles (package tree + obfuscation-resilient fuzzy method signatures) for libraries and target APKs, matches candidate hierarchies, and compares signatures to score presence and extent of use under obfuscation (enriching APK metadata with detailed TPL usage).

DeepMountain automates deep-learning workflows over multilingual (EN/ES) corpora of privacy policies, Terms of Service, and code artifacts from the project dataset. It supports hyperparameter optimization (learning rate, batch size, dropout, depth) guided by validation metrics, and tracks training/inference (convergence, misclassification patterns, semantic drift). Traditional and transformer-based NLP are integrated to link text and code corpora, enabling permission–policy alignment, compliance characterization against required policy elements, and auxiliary ToS fairness analysis.

Static/dynamic analyses complement the text pipeline. *Privado Scan* maps personal/sensitive data flows by locating sources and sinks and constructing a data map across 108 identifiable data types. *MobSF* performs static decompilation-based inspection (code, manifest, resources) and dynamic execution (network/API/system-call monitoring, behavioral anomalies), producing CI/CD-ready reports that cover insecure practices, permission misconfigurations, outdated dependencies, and malware indicators. *ATPChecker* examines privacy compliance in TPLs and host apps by analyzing bytecode (call graphs, flow analysis) and comparing it with automatically processed privacy policies to extract ADUP (flagging declared-vs-observed discrepancies as potential non-compliance).

To strengthen coverage under obfuscation, the framework adds public-string extraction from APK resources (via AAPT2) to infer potentially handled PII without full decompilation. This string-pool analysis, aligned with the 108-category scheme, offers a lightweight, scalable signal when metadata is stripped, complementing static/dynamic findings; limitations include missing bytecode-embedded or runtime-generated strings and possible resource obfuscation.

Sensitivity-aware analysis and objective risk assessment are combined for clear diagnostics. Each of the 108 personal data types is labeled with a sensitivity level (Very High / High / Moderate / Low) based on data characteristics, potential harms, and minimum thresholds, with contextual adaptation (setting, purpose, demographics) and automated alignment to evolving regulatory standards. Objective assessment scores *predictability*, *manageability*, and *disassociability* and summarizes them with a Green/Yellow/Red traffic-light scheme to surface strong alignment, partial gaps, or critical deficiencies.

The final implementation targets public deployment as a web platform and mobile app, offering transparent explanations, intuitive dashboards, color-coded indicators, and tailored reports; continuously updated educational content fosters proactive privacy management. Planned AI-driven enhancements include automatic categorization, policy/ToS summarization, detection of deceptive configurations and hidden permissions, context-based detection of sensitive content, stronger location-privacy protection, real-time bullying/harassment detection, proactive breach prevention, and protection of health data in social media. To support transparency and reproducibility, the source code is publicly available [3], and the multilingual dataset used for training and evaluation is accessible via the institutional repository [2].

VI. Experiments and Discussion

We evaluated *SafeMountain* end-to-end with an automated pipeline spanning large-scale app harvesting to consolidated privacy diagnostics. *AppCollector* assembled the top-200 Google Play social-networking apps, recording names, package IDs, version details, and APKs; in parallel, TPL metadata was pulled from Maven Central for attribution. XAPK packages (APKs with OBB) hindered static/dynamic tooling, prompting alternative re-assembly strategies. Beyond APKs and runtime traces, each app's privacy policy and ToS were crawled, normalized to EN/ES plain text, and classified by deep-learning models into GDPR-derived categories. We cross-referenced (i) manifest-declared permissions, (ii) disclosures in policies/ToS (via EN/ES NLP and LLMs such as LLaMA 2 for nuanced permission mentions), and (iii) permissions/sensitive behaviors invoked in code/runtime; reverse engineering (e.g., Androguard) linked permissions to API/bytecode use to surface omissions and potential non-compliance. The corpus is periodically refreshed to train/evaluate DistilBERT, RoBERTa, and BiLSTM with standard metrics (accuracy, precision, recall, F1, ROC–AUC) and techniques (SMOTE, attention masking, class balancing).

Static/dynamic tools are complementary. *Privado Scan* maps personal-data flows (sources→sinks) across 108 categories but shows clear limits on obfuscated proprietary apps (being tuned for unobfuscated repos), motivating stronger deobfuscation. *MobSF* adds static inspection (manifest, permissions, dependencies, insecure patterns, malware indicators) plus dynamic execution (network traffic, API/system calls, behavioral anomalies), revealing over-privileged configurations, outdated libraries, insecure transmissions, and undocumented server interactions. *ATPChecker* contrasts TPL bytecode behavior with declared practices, flagging discrepancies, yet struggles with reflection, dynamic loading, very large binaries (notably XAPK), and English-centric policy patterns. To strengthen TPL attribution under obfuscation we integrate *LibLoom*, which matches package hierarchies and fuzzy method signatures; effective use requires a substantial local mirror of library versions. Given Maven Central's scale (hundreds of thousands of libraries; millions of versions), we prioritize major/minor releases and schedule updates.

On widely used Android apps, a representative case, *Instagram Lite* (com.instagram.lite), exhibited high-risk permissions (`RECORD_AUDIO`, `ACCESS_FINE_LOCATION`, `READ_CONTACTS`), exported components with insufficient protection, and traffic consistent with non-trivial data sharing. *SafeMountain* correlates signals across modules and summarizes them via a traffic-light score while retaining provenance for audit. Integrated evidence (policies/ToS, permissions, components, TPLs, runtime traces) consistently exposed mismatches between declared and observed behavior, yielding actionable, reproducible diagnostics for audits, store vetting, and privacy-aware engineering.

Ethically, findings are reported as *potential* risks with transparent justification to avoid automation bias and reputational overreach; documentation supports critical interpretation. Practically, dynamic analysis on non-rooted devices is constrained by OS protections and privacy policies; priorities include controlled environments, lightweight instrumentation, and collaboration on safe interfaces.

Threats to validity span internal, external, construct, and reliability dimensions. Internally, popularity-based sampling may under-represent niche/regional apps; we mitigate via periodic refreshes, category expansion (e.g., finance, health), and comparisons against lower-ranked random samples. TPL coverage can miss micro-releases; gaps are flagged and mirrors updated regularly. Externally, Android-first tooling limits generalization; the architecture is modular to extend across platforms and languages. Construct risks (metric adequacy; heuristic scoring) are tempered by publishing rubrics, using domain

indicators (e.g., undocumented-permission rate, policy–code mismatch), and bilingual manual checks with active-learning fine-tuning to reduce NLP ambiguity. Reliability concerns (tool/version drift, stochastic training, nondeterministic runtime traces) are addressed with containerized pinned builds, fixed seeds and repeated runs, and replicated emulator analyses with aggregation. Overall, results support a multi-tool, multi-evidence approach to obtain nuanced, scalable, and reproducible privacy findings.

VII. Conclusions and Future Work

This paper reviewed methodologies, tools, and frameworks that leverage ML/NLP to assess privacy risks and support regulatory compliance (notably the GDPR), finding substantial progress but a persisting gap: the absence of a unified tool that both *quantifies* and *details* the full spectrum of risks in mobile apps and social networks. We argued for an integrated, holistic approach and outlined a detailed, quantifiable, and user-friendly framework aligned with international requirements and practical engineering principles to empower users with clear, actionable insights. The proposed solution systematically addresses issues from data sensitivity and processing practices to legal/regulatory conformity, integrating static and dynamic analyses with NLP-based policy evaluation and deep learning for automated categorization and summarization, enabling plain-language communication of complex risks.

Building on this, *SafeMountain* consolidates risk quantification/visualization, multilingual policy interpretation, and code-based discrepancy detection into a unified AI-driven system, grounded in privacy-engineering principles and regulatory guidelines and summarized via an interpretable traffic-light scheme. Experimental results validate detection of policy discrepancies, third-party library risks, and delivery of actionable insights for users, developers, and regulators. Its modular, multilingual design supports extensibility and adaptability to evolving threats and domains.

Future work will advance the framework for mobile apps and social networks by improving accuracy, scalability, and usability, and by exploring high-sensitivity domains (e.g., gaming, health). Overall, this study refines the understanding of privacy risk assessment, identifies strengths and open challenges, and points to a unified, state-of-the-art solution sustained by interdisciplinary collaboration and continuous refinement to help all users navigate data privacy with confidence.

Acknowledgments

The authors would like to acknowledge the support of the Spanish Government (through the Cybersecurity Institute of Spain, INCIBE), with the Strategic Research Project "Analysis of mobile applications from the perspective of data protection: Cyber-protection and Cyber-risks of citizen information" and the International Chair "Smart Rural IoT and Secured Environments", within the context of the Recovery, Transformation and Resilience Plan financed by the European Union (NextGenerationEU/PRTR). The authors also thank the support of the UNED Associated Center Consortium in Ponferrada, INTECCA, as well as the UNED CiberCSI research group, and the UNED CiberGID innovation group.

References

[1] D. Blanco-Aza, A. Robles-Gómez, R. Pastor-Vargas, L. Tobarra, P. Vidal-Balboa, and M. Méndez-Suárez, "Comprehensive AI-driven privacy risk assessment in mobile apps and social networks," *Cluster Computing*, 2025.

[2] E. García-Arias, A. Robles-Gómez, R. Pastor-Vargas, L. Tobarra, P. Vidal-Balboa, D. Blanco-Aza, and A. Valtuille-Pacios, "Dataset de política de privacidad y términos de uso de aplicaciones móviles en inglés y castellano," 2025.

[3] D. Blanco-Aza, A. Robles-Gómez, R. Pastor-Vargas, L. Tobarra, P. Vidal-Balboa, E. García-Arias, and A. Valtuille-Pacios, "SafeMountain," https://github.com/uned-cibergid-projects/safemountain, 2025, accessed May 21, 2025.

[4] Y. Feng, L. Chen, A. Zheng, C. Gao, and Z. Zheng, "AC-Net: Assessing the Consistency of Description and Permission in Android Apps," *IEEE Access*, vol. 7, pp. 57 829–57 842, 2019.

[5] A. Jain, D. Rodriguez, J. M. del Alamo, and N. Sadeh, "ATLAS: Automatically Detecting Discrepancies Between Privacy Policies and Privacy Labels," *arXiv*, 2023. [Online]. Available: http://arxiv.org/abs/2306.09247

[6] M. S. Rahman, P. Naghavi, B. Kojusner, S. Afroz, B. Williams, S. Rampazzi, and V. Bindschaedler, "PermPress: Machine Learning-Based Pipeline to Evaluate Permissions in App Privacy Policies," *IEEE Access*, vol. 10, pp. 89 248–89 269, 2022.

[7] A. Xiang, W. Pei, and C. Yue, "PolicyChecker: Analyzing the GDPR Completeness of Mobile Apps' Privacy Policies," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2023, pp. 3373–3387.

[8] B. Andow, S. Y. Mahmud, J. Whitaker, W. Enck, B. Reaves, W. Wang, K. Singh, and T. Xie, "PolicyLint: Investigating Internal Privacy Policy Contradictions on Google Play," https://www.usenix.org/conference/usenixsecurity19/presentation/andow, 2019.

[9] K. Zhao, X. Zhan, L. Yu, S. Zhou, H. Zhou, X. Luo, H. Wang, and Y. Liu, "Demystifying Privacy Policy of Third-Party Libraries in Mobile Apps," in *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, 2023, pp. 1583–1595.

[10] Y. Yang, X. Du, and Z. Yang, "PRADroid: Privacy Risk Assessment for Android Applications," in *2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP)*, 2021, pp. 90–95.

[11] H. N. Atapattu, W. S. N. Fernando, J. P. A. K. Somasiri, P. M. K. Lokuge, A. N. Senarathne, and M. Tissera, "A Sensitive Data Leakage Detection and Privacy Policy Analyzing Application for Android Systems (PriVot)," in *2021 3rd International Conference on Advancements in Computing (ICAC)*, 2021, pp. 300–304.

[12] S. Li, Z. Yang, N. Hua, P. Liu, X. Zhang, G. Yang, and M. Yang, "Collect Responsibly But Deliver Arbitrarily?: A Study on Cross-User Privacy Leakage in Mobile Apps," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2022, pp. 1887–1900.

[13] A. J. Bhatt, C. Gupta, and S. Mittal, "iABC-AL: Active learning-based privacy leaks threat detection for iOS applications," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 7, pp. 769–786, 2021.

[14] S. Liu, F. Zhang, B. Zhao, R. Guo, T. Chen, and M. Zhang, "APPCorp: A corpus for Android privacy policy document structure analysis," *Frontiers of Computer Science*, vol. 17, no. 173320, 2023.

[15] D. Blanco-Aza, A. Robles-Gómez, R. Pastor-Vargas, L. Tobarra, P. Vidal-Balboa, and M. Méndez-Suárez, "Privacy Analysis in Mobile Apps and Social Networks using AI Techniques," in *Proceedings of The Fourth Intelligent Cybersecurity Conference (ICSC 2024)*, Valencia, Spain, 2024.

[16] S. Brooks, M. Garcia, N. Lefkovitz, S. Lightman, and E. Nadeau, "An Introduction to Privacy Engineering and Risk Management in Federal Systems," *NIST Interagency/Internal Report 8062*, 2017.

[17] C. Anwar, C. Sumerli, N. Rahayu, and K. Kraugusteeliana, "The Application of Mobile Security Framework (MOBSF) and Mobile Application Security Testing Guide to Ensure the Security in Mobile Commerce Applications," *Jurnal Sistim Informasi dan Teknologi*, vol. 5, pp. 97–102, 2023.

[18] J. Huang, B. Xue, J. Jiang, W. You, B. Liang, J. Wu, and Y. Wu, "Scalably Detecting Third-Party Android Libraries With Two-Stage Bloom Filtering," *IEEE Transactions on Software Engineering*, vol. 49, no. 4, pp. 2272–2284, 2023.

[19] Agencia Española de Protección de Datos, "Gestión del riesgo y evaluación de impacto en tratamientos de datos personales," 2021, accessed May 20, 2025. [Online]. Available: https://www.aepd.es/guias/gestion-riesgo-y-evaluacion-impacto-en-tratamientos-datos-personales.pdf

[20] E. Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation)," pp. 1–88, 2016. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj

[21] "Regulation (eu) 2022/2065 of the european parliament and of the council of 19 october 2022 on a single market for digital services (digital services act)," https://eur-lex.europa.eu/eli/reg/2022/2065/oj, 2022, accessed May 20, 2025.

[22] S. Sharma, "Building an Automated Machine for Discovering Privacy Violations at Scale," https://www.usenix.org/conference/enigma2023/presentation/sharma, 2023.

Marco para la adquisición remota y automatizada de evidencias digitales en entornos corporativos e industriales

David García-Fernández

Programa de Doctorado en Tecnologías Industriales,
Escuela Internacional de Doctorado (EIDUNED)
Universidad Nacional de Educación a Distancia (UNED)
Madrid 28040, España
dgarcia@scc.uned.es

Llanos Tobarra, Antonio Robles-Gómez,
Rafael Pastor-Vargas

Departamento de Sistemas de Comunicación y Control,
ETSI Informática
Universidad Nacional de Educación a Distancia (UNED)
Madrid 28040, España
{llanos,arobles,rpastor}@scc.uned.es

***Resumen*—La creciente complejidad de los entornos corporativos e industriales plantea retos inéditos en la adquisición de evidencias digitales, especialmente aquellas de carácter volátil. Este trabajo presenta un marco de adquisición remota, automatizada y sin agentes, diseñado para operar de forma segura en infraestructuras críticas y entornos híbridos. Se sintetiza el estado del arte en herramientas y marcos de referencia, y se propone una arquitectura definida como código basada en Ansible que permite garantizar trazabilidad, reproducibilidad y alineamiento con normas internacionales. A partir de este planteamiento se analizan las ventajas, limitaciones y riesgos del enfoque, y se establecen líneas futuras centradas en su validación en entornos industriales reales e integración con técnicas de inteligencia artificial. El objetivo es consolidar un proceso forense digital trazable, eficiente y legalmente defendible.**

***Index Terms*—análisis forense digital, adquisición remota, automatización, memoria volátil, ICS (Sistemas de Control Industrial), cadena de custodia**

I. Introducción

El análisis forense digital se ha consolidado como un componente esencial en la investigación de incidentes de ciberseguridad. La digitalización masiva, la proliferación de servicios en la nube y la integración de tecnologías operativas en entornos industriales han multiplicado la superficie de ataque y la complejidad en la gestión de evidencias. La disciplina de Respuesta a Incidentes y Análisis Forense Digital (DFIR) adquiere un papel crítico, no solo para determinar el alcance de un ataque, sino también para garantizar la validez legal de la evidencia recopilada.

La volatilidad de datos en memoria RAM, procesos activos, sesiones efímeras y contenedores requiere mecanismos de captura rápidos, reproducibles y mínimamente intrusivos, cuya relevancia ha sido ampliamente reconocida en la literatura [1], [2], [3]. La gestión manual resulta costosa y propensa a errores, lo que abre la puerta a soluciones que incorporen automatización, trazabilidad y mecanismos de orquestación [2]. En entornos industriales, la situación es aún más crítica debido a la heterogeneidad tecnológica y la falta de interfaces estandarizadas, como señalan Cook et al. [4] y Stoyanova et al. [5].

Los estándares internacionales, como ISO/IEC 27037 [6] e ISO/IEC 27041 [7], así como las guías publicadas por ENISA específicas para sistemas de control industrial [8], subrayan la necesidad de asegurar la integridad, autenticidad y trazabilidad de la evidencia, estableciendo principios rectores que deben guiar cualquier proceso de adquisición. A nivel europeo, este marco se refuerza con regulaciones como la Directiva NIS2 [9], el Reglamento e-Evidence [10] y el Segundo Protocolo del Convenio de Budapest [11], que imponen obligaciones explícitas en materia de conservación y acceso a datos electrónicos, garantizando la cooperación transfronteriza y la auditabilidad de los procesos.

En este escenario regulatorio y técnico, se proyecta un marco de adquisición forense agentless, automatizado y definido como código, capaz de integrarse en entornos corporativos e industriales sin añadir riesgos adicionales. El empleo de infraestructuras reproducibles y orquestadas mediante herramientas abiertas como Ansible [12], en línea con propuestas recientes en IIoT [13], permite reducir la intervención manual, mejorar la escalabilidad y facilitar la integración con procedimientos de gestión de incidentes. Este enfoque no solo responde a la necesidad de rapidez y eficiencia, sino que también se alinea con las exigencias normativas y las buenas prácticas internacionales, ofreciendo un camino hacia investigaciones forenses más fiables, transparentes y legalmente defendibles.

I-A. Preguntas de investigación y contribuciones

A pesar de la relevancia práctica de la adquisición automatizada de evidencias digitales, la contribución científica de los enfoques *agentless* no ha sido formulada de manera explícita en la literatura existente. Para cubrir este vacío, este trabajo se articula en torno a las siguientes preguntas de investigación (PI):

- **PI1**: ¿Es viable, desde el punto de vista técnico y operativo, un enfoque *agentless* para la adquisición remota de evidencias digitales volátiles en entornos corporativos e industriales simulados, manteniendo la integridad y la trazabilidad?
- **PI2**: ¿Cómo se compara el enfoque propuesto, en términos de tiempo de adquisición e impacto operativo, con una plataforma forense basada en agentes como Velociraptor?

- **PI3**: ¿Hasta qué punto un entorno ICS virtualizado permite validar este tipo de arquitecturas y cuáles son sus limitaciones respecto a despliegues industriales reales?

A partir de estas preguntas, las principales contribuciones de este trabajo son las siguientes:

1. El diseño de un marco de adquisición forense definido como código, basado en Ansible, que permite la adquisición remota, automatizada y sin agentes de evidencias digitales.
2. La validación experimental del marco en tres entornos complementarios: un laboratorio corporativo virtualizado, un entorno en la nube pública y un entorno ICS virtualizado.
3. Una comparación cuantitativa y cualitativa con una solución forense basada en agentes (Velociraptor), que permite identificar ventajas y limitaciones de ambos enfoques.
4. Una discusión crítica sobre el alcance real y las limitaciones del enfoque propuesto en contextos industriales.

El resto del artículo se organiza de la siguiente manera. La Sección II presenta una revisión sintética del estado del arte en herramientas, marcos y enfoques aplicables a la adquisición forense digital, incluyendo entornos industriales. La Sección III describe la metodología seguida y la arquitectura propuesta, mientras que la Sección IV recoge la discusión de resultados y las principales conclusiones derivadas del trabajo.

II. Estado del arte

El estado del arte en adquisición forense digital puede organizarse en tres bloques principales, atendiendo al tipo de soluciones disponibles y a su grado de madurez tecnológica. En primer lugar, destacan las soluciones comerciales, que ofrecen suites integradas con capacidades avanzadas de adquisición remota, captura de memoria volátil y procesamiento simultáneo de múltiples dispositivos. Herramientas como Magnet AXIOM Cyber, Cellebrite UFED, OpenText EnCase Forensic o Belkasoft Evidence Center X representan opciones consolidadas en el mercado, orientadas a entornos corporativos con necesidades de escalabilidad y soporte especializado [14], [15], [16], [17].

En segundo lugar, se encuentran las herramientas abiertas, que han alcanzado una amplia adopción tanto en ámbitos académicos como profesionales. Proyectos como Autopsy/The Sleuth Kit, CAINE, Volatility o SIFT constituyen un ecosistema diverso para la adquisición y análisis de evidencias digitales, con especial énfasis en la memoria RAM y en la transparencia metodológica. Aunque estas herramientas carecen en muchos casos de capacidades nativas para adquisición remota y orquestación, su modularidad y extensibilidad facilitan su integración en flujos de trabajo más amplios [18], [19], [20]. Destaca además Velociraptor como plataforma avanzada de código abierto, que combina la recolección de datos con monitorización de endpoints y ejecución distribuida, aunque basada en agentes residentes [21].

El tercer bloque lo constituyen los marcos emergentes, que buscan superar las limitaciones de las herramientas tradicionales mediante enfoques innovadores. Entre ellos se incluyen AutoDFBench, orientado a la validación de código forense generado por inteligencia artificial [22]; MF-Ledger, que aplica tecnología blockchain para garantizar integridad y trazabilidad de la cadena de custodia [23]; o SOLVE-IT, que sistematiza el conocimiento forense en una base de datos semántica capaz de recomendar metodologías adaptadas al caso de uso [24]. Estos marcos reflejan una tendencia hacia la automatización, la reproducibilidad y la integración con paradigmas como DevOps o MLOps.

Finalmente, el dominio industrial (ICS) introduce desafíos específicos. La criticidad operativa, la heterogeneidad de dispositivos y la prevalencia de protocolos propietarios limitan la aplicación de metodologías forenses convencionales. Investigaciones recientes han evidenciado estas limitaciones y han propuesto soluciones específicas, como ICSREF para la ingeniería inversa de binarios de PLCs [26], PEM para la adquisición remota de memoria en PLCs [25], o recolectores automatizados en redes eléctricas inteligentes [27]. A nivel de síntesis, Cook et al. [4] y Stoyanova et al. [5] destacan la necesidad de metodologías reproducibles y adaptadas a la naturaleza crítica de estos entornos. El estado del arte en ICS se encuentra en una fase emergente, donde la investigación académica comienza a sentar las bases de marcos especializados, todavía lejos de la consolidación alcanzada en el ámbito corporativo.

II-A. Soluciones comerciales

Entre las soluciones de carácter comercial destacan varias plataformas consolidadas en el ámbito del análisis forense digital, ampliamente utilizadas tanto en entornos corporativos como en cuerpos policiales y agencias de seguridad. Magnet AXIOM Cyber se ha posicionado como una de las herramientas más completas al permitir la adquisición y análisis de evidencias desde *endpoints*, servicios en la nube y dispositivos móviles. Además, integra la utilidad Magnet RAM Capture para la obtención de memoria volátil y se complementa con Magnet AUTOMATE, que facilita la orquestación de procesos de adquisición y análisis en grandes volúmenes de dispositivos, mejorando así la escalabilidad [14].

En una línea similar, Cellebrite UFED, inicialmente orientada a la extracción de datos en dispositivos móviles, ha ampliado su alcance hacia estaciones de trabajo mediante su integración con Cellebrite Endpoint Inspector. Esta combinación habilita la adquisición en vivo de memoria y el acceso remoto a múltiples nodos en paralelo, lo que resulta especialmente útil en investigaciones distribuidas que requieren rapidez de intervención [15]. Por su parte, OpenText EnCase Forensic constituye una referencia consolidada en cuerpos policiales y organizaciones gubernamentales. Sus capacidades incluyen tanto la adquisición de datos volátiles como persistentes, así como la ejecución remota mediante de agentes ligeros. EnCase destaca por su robustez en escenarios judiciales y por la fiabilidad de su cadena de custodia, aunque sus costes de licencia y su dependencia de agentes residentes limitan su aplicabilidad en determinados contextos [16].

Finalmente, Belkasoft Evidence Center X ofrece compatibilidad con entornos protegidos e integra un capturador propio de memoria RAM (Belkasoft Live RAM Capturer), capaz de operar incluso en presencia de protecciones anti-*debugger*. Además, soporta adquisición remota mediante agentes desplegables y proporciona automatización avanzada para procesos de adquisición, análisis y generación de informes [17]. En

conjunto, estas herramientas comerciales representan soluciones maduras que ofrecen cobertura funcional y soporte especializado, pero comparten limitaciones importantes: su dependencia de agentes residentes en los sistemas objetivo y los elevados costes de adquisición y mantenimiento, factores que restringen su adopción en entornos sensibles o con restricciones presupuestarias.

II-B. Herramientas abiertas

Dentro del ámbito de las soluciones de código abierto, destacan diversos proyectos que han contribuido significativamente al avance del análisis forense digital, especialmente en contextos académicos y de investigación aplicada. Volatility se ha consolidado como el marco de referencia para el análisis forense de memoria RAM, siendo ampliamente adoptado por la comunidad profesional y científica. Este framework permite examinar en profundidad el estado interno de un sistema a partir de una imagen de memoria, ofreciendo capacidades para identificar procesos activos, conexiones de red, módulos cargados o indicios de malware en ejecución. Aunque no realiza la adquisición de memoria por sí mismo, Volatility es compatible con múltiples formatos de imágenes y puede integrarse con herramientas externas para la captura, formando así un componente esencial dentro de flujos de análisis automatizados [20].

Autopsy, junto con The Sleuth Kit, representa otra de las soluciones de referencia en el ámbito forense. The Sleuth Kit proporciona un conjunto de utilidades de línea de comandos orientadas al análisis de sistemas de archivos y estructuras de disco, mientras que Autopsy añade una interfaz gráfica que facilita el examen de evidencias y la generación de informes estructurados. Su integración con Volatility permite combinar el análisis de memoria con el estudio de artefactos de disco, ofreciendo una visión más completa del sistema analizado. No obstante, carece de capacidades nativas de adquisición remota, aunque admite cierto grado de automatización mediante su función de ingestión programada [18].

En la misma línea, SIFT Workstation —desarrollado por el SANS Institute— agrupa un amplio conjunto de utilidades forenses preconfiguradas para análisis de sistemas de archivos, memoria y redes. Integra herramientas como Volatility y LiME para la captura y el análisis de memoria, y proporciona un entorno homogéneo que facilita la replicación de procedimientos en laboratorios y ejercicios de formación. Sin embargo, SIFT no incorpora mecanismos de orquestación remota ni automatización nativa, lo que limita su aplicabilidad en escenarios distribuidos [20].

CAINE (*Computer Aided INvestigative Environment*) ofrece un entorno Linux diseñado específicamente para tareas forenses, con un amplio repertorio de herramientas preinstaladas que abarcan desde la adquisición de memoria —mediante utilidades como LiME o WinPmem— hasta el análisis de sistemas de archivos, registros y redes. Gracias a su diseño modular y su carácter educativo, CAINE resulta especialmente útil en entornos de formación y en laboratorios experimentales, aunque no dispone de un sistema centralizado de automatización o control remoto [19].

Por último, Velociraptor constituye una de las plataformas más avanzadas de código abierto para la monitorización de *endpoints*, el análisis forense digital y la respuesta a incidentes. Basado en agentes desplegables, permite la adquisición remota de datos, incluida la memoria volátil, mediante un lenguaje de consultas declarativo denominado *Velociraptor Query Language* (VQL). Esta arquitectura ofrece una gran flexibilidad y escalabilidad, permitiendo gestionar miles de clientes desde una única instancia del servidor. Sin embargo, su dependencia de agentes residentes y su complejidad de configuración pueden limitar su aplicabilidad en entornos industriales (ICS), donde las políticas de seguridad restringen la instalación de componentes persistentes [21].

En conjunto, estas herramientas de código abierto constituyen la base del ecosistema forense contemporáneo. Aunque presentan limitaciones en automatización y orquestación remota, su transparencia, capacidad de personalización y amplia adopción las convierten en pilares fundamentales para la investigación, la formación y el desarrollo de marcos experimentales reproducibles.

II-C. Marcos emergentes

En los últimos años han surgido marcos de trabajo orientados a reforzar la transparencia, la reproducibilidad y la automatización de los procesos forenses, como respuesta al aumento de la complejidad de los entornos digitales y a la incorporación progresiva de técnicas de inteligencia artificial.

AutoDFBench se centra en la validación de herramientas forenses y de código generado por IA, proporcionando conjuntos de pruebas automatizados que permiten evaluar la corrección, robustez y trazabilidad de los resultados frente a una verdad de referencia. Este tipo de marcos resulta especialmente relevante en escenarios donde se emplean técnicas de aprendizaje automático, cuya opacidad puede dificultar la justificación técnica y legal de los resultados obtenidos, fomentando la transparencia metodológica y la comparabilidad de resultados entre soluciones impulsadas por IA [22].

Por su parte, MF-Ledger propone el uso de tecnologías blockchain para registrar de forma inmutable las operaciones realizadas sobre las evidencias digitales a lo largo de la cadena de custodia. Mediante el almacenamiento de eventos forenses como transacciones verificables, este enfoque refuerza la integridad y auditabilidad del proceso, facilitando revisiones independientes y aportando garantías adicionales en contextos regulatorios exigentes [23].

Finalmente, SOLVE-IT adopta un enfoque basado en grafos de conocimiento para sistematizar procedimientos, técnicas y herramientas forenses, permitiendo recomendar metodologías adaptadas al tipo de caso y al entorno operativo. Este tipo de sistemas resulta especialmente útil para apoyar la toma

Tabla I: Comparativa visual de herramientas forenses.

Herramienta	Adq. memoria	Adq. remota	Automatización	Multidispositivo
Magnet AXIOM Cyber	🟢	🟢	🟢	🟢
Cellebrite UFED	🟡	🟢	🟡	🟢
OpenText EnCase Forensic	🟢	🟢	🟡	🟢
Belkasoft Evidence Center X	🟢	🟢	🟢	🟢
Autopsy + TSK	🔴	🔴	🟢	🔴
CAINE	🟢	🟡	🟡	🟡
Volatility	🔴	🔴	🟡	🔴
SIFT	🟢	🔴	🟡	🔴
Velociraptor	🟢	🟢	🟢	🟢

de decisiones en investigaciones complejas y para reducir la dependencia de conocimiento experto no documentado [24].

En conjunto, estos marcos emergentes representan un avance hacia la estandarización, la reproducibilidad y la auditabilidad de las investigaciones digitales. Si bien su adopción práctica aún es incipiente, sientan las bases de una nueva generación de herramientas forenses más transparentes, interoperables y confiables, alineadas con los principios de verificabilidad y trazabilidad exigidos por las normas internacionales y las agencias de ciberseguridad.

II-D. Entornos industriales: desafíos y herramientas específicas

El análisis forense digital en entornos industriales se enfrenta a desafíos particulares derivados de la criticidad operativa de los Sistemas de Control Industrial (ICS), su elevada heterogeneidad tecnológica y los estrictos requisitos de disponibilidad continua. Estas características dificultan la aplicación directa de metodologías forenses tradicionales, ya que cualquier intervención no planificada puede afectar al funcionamiento del proceso industrial o introducir riesgos operativos inaceptables [4].

La adquisición de evidencias digitales en Controladores Lógicos Programables (PLCs) y otros dispositivos industriales resulta especialmente compleja debido a la ausencia de interfaces estandarizadas y a la naturaleza propietaria de muchos de estos sistemas. En la literatura se han propuesto enfoques basados en métodos físicos invasivos, como el acceso mediante *Joint Test Action Group* (JTAG), así como alternativas orientadas a la adquisición remota de información a través de protocolos industriales o mecanismos específicos de los fabricantes, con el objetivo de reducir el impacto sobre la operación [4], [25].

En este contexto han surgido herramientas y marcos especializados que abordan parcialmente estas limitaciones. *Industrial Control Systems Reverse Engineering Framework* (ICSREF) se centra en la ingeniería inversa de binarios de PLCs para analizar la lógica de control y detectar modificaciones no autorizadas, mientras que propuestas como PEM exploran la viabilidad de la adquisición remota de memoria en dispositivos industriales. De forma complementaria, algunos trabajos plantean el uso de recolectores forenses automatizados en infraestructuras como redes eléctricas inteligentes, con el fin de minimizar la intervención manual y el impacto operativo [26], [25], [27].

Desde el punto de vista normativo, estándares como la familia IEC 62443 y las recomendaciones publicadas por ENISA subrayan la necesidad de preservar la disponibilidad del sistema y de priorizar enfoques forenses no intrusivos, alineados con los requisitos de seguridad y continuidad operativa [28], [8]. Estas directrices refuerzan la conveniencia de soluciones que integren automatización y trazabilidad sin comprometer el funcionamiento de los entornos industriales.

III. Metodología

Este trabajo adopta la metodología *Design Science Research Methodology* (DSRM) [29], [30] como marco de referencia para el diseño y evaluación de un artefacto tecnológico orientado a la adquisición forense digital. DSRM proporciona una estructura sistemática que permite vincular la identificación del problema con la definición de objetivos, el desarrollo de la solución y su validación experimental, resultando especialmente adecuada para investigaciones de carácter aplicado.

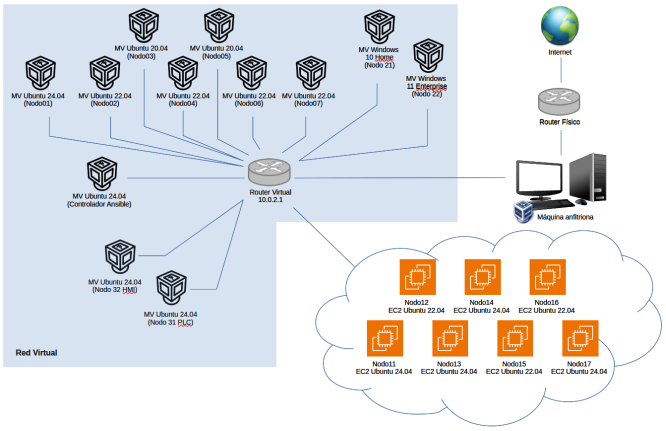


Figura 1: Arquitectura *agentless* para la adquisición remota y automatizada de evidencias digitales.

El problema abordado se centra en la dificultad de realizar una adquisición remota y fiable de evidencias digitales, especialmente de carácter volátil, en entornos corporativos e industriales caracterizados por su heterogeneidad, criticidad operativa y restricciones sobre la instalación de software. Frente a este contexto, el objetivo del trabajo es diseñar un marco de adquisición automatizado y sin agentes que reduzca la intrusividad, minimice el error humano y preserve la integridad y trazabilidad de las evidencias obtenidas, ajustándose a los requisitos de preservación, fiabilidad y repetibilidad establecidos en las normas ISO/IEC 27037 e ISO/IEC 27041.

Para alcanzar este objetivo se implementa una arquitectura basada en Ansible, que permite orquestar de forma remota la ejecución de herramientas forenses portables y definir el proceso de adquisición como código. Este enfoque facilita la reproducibilidad de los experimentos, la auditoría de las acciones realizadas y la adaptación del marco a distintos escenarios, manteniendo un equilibrio entre automatización, control y simplicidad operativa.

III-A. Arquitectura del marco propuesto

La solución planteada sigue una arquitectura centralizada y *agentless*, en la que un nodo controlador orquesta las tareas de adquisición forense sobre múltiples nodos remotos mediante Ansible (Figura 1). Todos los procesos de adquisición se definen como código, lo que garantiza repetibilidad, trazabilidad y auditabilidad.

Los nodos objetivo se acceden utilizando mecanismos estándar de administración remota (SSH o WinRM), sin necesidad de instalar software persistente. Las utilidades forenses portables se transfieren de forma temporal, se ejecutan para la adquisición de evidencias y, cuando es posible, se eliminan al finalizar el proceso.

El flujo general de adquisición consta de los siguientes pasos:

1. Selección de los nodos objetivo a partir del inventario de Ansible.

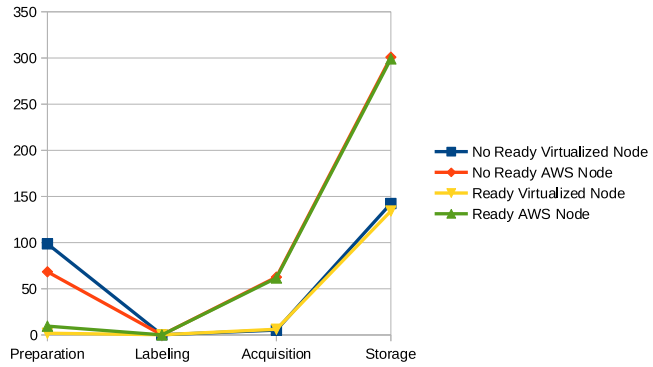


Figura 2: Gráfico comparativo de tiempos de adquisición entre los diferentes entornos experimentales

2. Verificación del entorno y creación de la estructura de directorios del caso.
3. Despliegue temporal de las herramientas forenses necesarias.
4. Ejecución de la adquisición de evidencias (volátiles y/o no volátiles).
5. Cálculo de valores hash criptográficos para garantizar la integridad.
6. Recolección centralizada de las evidencias y de los registros de ejecución.
7. Eliminación opcional de artefactos temporales en los nodos analizados.

La validación del prototipo se plantea en tres entornos complementarios. En primer lugar, un laboratorio virtual local permite verificar el funcionamiento básico del sistema sobre múltiples máquinas Linux y Windows interconectadas, simulando un entorno corporativo controlado. A continuación, se despliega la misma infraestructura en Amazon Web Services (AWS) para comprobar su comportamiento en un entorno de nube pública, con condiciones de latencia y escalabilidad diferentes. Finalmente, se crea un entorno industrial virtualizado que integra un PLC implementado con OpenPLC y un SCADA configurado con Node-RED. Este último escenario sirve para evaluar la aplicabilidad del enfoque en un contexto con restricciones propias de los sistemas ICS, donde la intrusión debe reducirse al mínimo y la disponibilidad no puede verse comprometida. Aunque este escenario permite reproducir interacciones de red y restricciones operativas realistas, no representa la totalidad de las particularidades de sistemas industriales reales. Los resultados deben interpretarse como un estudio de viabilidad y no como una solución lista para su despliegue directo en plantas industriales reales.

III-B. Métricas de evaluación

Para realizar una evaluación objetiva y comparable, durante los experimentos se recogieron las siguientes métricas:

- **Tiempo de adquisición (s)**: tiempo total de ejecución por nodo (Figura 2).
- **Volumen de datos**: tamaño total de las evidencias transferidas.
- **Fiabilidad**: tasa de éxito de las ejecuciones repetidas.

La integridad de las evidencias se verifica mediante el cálculo de valores hash criptográficos (SHA-256) inmediatamente tras la adquisición y tras su transferencia al repositorio central. Adicionalmente, se comprueba la consistencia de los resultados frente a una herramienta consolidada en el sector, Velociraptor, que representa un modelo opuesto basado en agentes residentes. La comparación permite identificar con claridad las ventajas y limitaciones de ambos enfoques en distintos escenarios operativos.

El trabajo concluye con la documentación completa del proceso, desde la configuración de los entornos y los *playbooks* utilizados hasta la descripción detallada de los resultados. Todo el material se organiza para facilitar la replicación del experimento, para que otros investigadores o profesionales puedan reproducir y ampliar el estudio con nuevas herramientas o entornos. Este componente de transparencia se considera un valor esencial del proyecto, en línea con las buenas prácticas de investigación reproducible.

IV. Discusión

Los resultados obtenidos confirman que la automatización sin agentes constituye una alternativa sólida frente a los enfoques tradicionales basados en agentes residentes. La solución propuesta demuestra que es posible ejecutar procesos de adquisición forense remota de forma fiable y con un impacto mínimo sobre los sistemas analizados, ya que al no requerir la instalación de software persistente se reduce la superficie de ataque y se evitan interferencias con los procesos de negocio o de control industrial, algo especialmente relevante en entornos sensibles o críticos, reforzando además la validez técnica y legal del procedimiento mediante una trazabilidad completa de cada ejecución.

El uso de Ansible como núcleo de orquestación aporta un equilibrio adecuado entre flexibilidad y control. Su enfoque declarativo permite definir la adquisición como código, lo que facilita la repetición exacta de cada proceso y la verificación posterior de los pasos ejecutados. Los *playbooks* actúan como una descripción transparente y auditable del procedimiento, versionable y revisable como cualquier otro artefacto de software, favoreciendo la coherencia entre entornos y su integración en metodologías de infraestructura como código o flujos de integración continua.

El enfoque también contribuye a simplificar la adopción de estándares y buenas prácticas internacionales. La posibilidad de documentar automáticamente las acciones realizadas, conservar los registros de ejecución y aplicar funciones de verificación criptográfica facilita el cumplimiento de normas como ISO/IEC 27037 e ISO/IEC 27041. Esta alineación no se limita a la teoría, ya que el sistema produce evidencias con metadatos trazables, comprobaciones de integridad y registros reproducibles, elementos imprescindibles para asegurar su admisibilidad y confianza.

Otro aspecto relevante es la eficiencia operativa, ya que las pruebas realizadas muestran que los tiempos de adquisición y la estabilidad del sistema mejoran de forma significativa cuando se trabaja sobre nodos previamente preparados, sin que ello comprometa la flexibilidad del entorno. El modelo *agentless* aprovecha la ejecución paralela de Ansible para operar sobre múltiples equipos de forma simultánea, lo que reduce los tiempos de respuesta ante incidentes y amplía el alcance de la adquisición sin requerir infraestructura adicional. En comparación con herramientas basadas en agentes, el

enfoque se muestra más ligero, más fácil de desplegar y menos dependiente de configuraciones previas.

La comparación con Velociraptor permite establecer una distinción clara entre ambos enfoques. En términos de tiempo medio de adquisición, no se aprecian diferencias significativas en nodos no preparados: la captura de una imagen de memoria de 4 GB requiere aproximadamente 245 s con Velociraptor frente a 246 s utilizando Ansible; sin embargo, en nodos previamente preparados se reduce el tiempo hasta 142 s, mostrando una mejora sustancial.

Más allá de los tiempos, Velociraptor ofrece una mayor granularidad y capacidad analítica mediante su lenguaje de consultas VQL, a costa de una infraestructura más compleja y del uso de agentes residentes, mientras que el marco basado en Ansible prioriza la rapidez de despliegue, la transparencia y una baja intrusividad al no requerir componentes persistentes. En consecuencia, el enfoque sin agentes resulta especialmente adecuado en contextos con fuertes restricciones operativas o donde la disponibilidad del sistema es crítica —como ocurre habitualmente en redes industriales—, mientras que herramientas como Velociraptor pueden aportar mayor valor en fases posteriores, una vez estabilizado el entorno y cuando se requiere un análisis más exhaustivo, siendo ambas aproximaciones complementarias dentro de una estrategia forense integral.

La aplicabilidad del marco a entornos industriales representa uno de sus mayores valores añadidos. El laboratorio ICS virtualizado confirma que es posible adaptar los principios de adquisición forense a infraestructuras que emplean protocolos industriales y dispositivos de control sin interferir en su funcionamiento. Aunque la validación se realiza sobre entornos simulados, los resultados apuntan a que este modelo puede evolucionar hacia una integración real en sistemas ICS, siempre que se respeten las limitaciones operativas y normativas impuestas por estándares como la IEC 62443 y las guías de ENISA para infraestructuras críticas. La capacidad de ejecutar adquisiciones controladas y auditables en este tipo de redes abre una línea prometedora para futuras investigaciones.

En conjunto, la solución desarrollada combina automatización, trazabilidad y cumplimiento normativo en un marco reproducible y técnicamente sostenible. El modelo *agentless* demuestra que la adquisición remota de evidencias puede realizarse de forma segura, verificable y alineada con las exigencias de integridad y transparencia que demandan los entornos corporativos e industriales. El trabajo sienta así una base práctica para la evolución de metodologías forenses más eficientes y menos invasivas, orientadas tanto a la respuesta a incidentes como a la mejora continua de las capacidades de investigación digital.

Agradecimientos

Los autores desean agradecer el apoyo del Gobierno de España (a través del Instituto Nacional de Ciberseguridad de España, INCIBE), con el Proyecto de Investigación Estratégico “Análisis de aplicaciones móviles desde la perspectiva de la protección de datos: Ciberprotección y Ciberriesgos de la información ciudadana”, así como la Cátedra Internacional “Smart Rural IoT y Entornos Seguros”, en el contexto del Plan de Recuperación, Transformación y Resiliencia financiado por la Unión Europea (NextGenerationEU/PRTR). Asimismo, los autores agradecen el apoyo del grupo de investigación UNED CiberCSI y del grupo de innovación UNED CiberGID.

Referencias

[1] Michael W. Graves: ”Digital Archaeology: The Art and Science of Digital Forensics”, Addison-Wesley, 2014.
[2] John Sammons: ”The Basics of Digital Forensics.2nd Ed.”, Syngress, 2014.
[3] Ken Zatyko: Çommentary: Defining digital forensics”, *Forensic Magazine*, vol. February/March, pp. 1-5, 2007.
[4] Marco M. Cook et al.: .A Survey on Industrial Control System Digital Forensics: Challenges, Advances and Future Directions”, *IEEE Communications Surveys & Tutorials*, vol. 25, n. 3, pp. 1705-1747, 2023.
[5] Maria Stoyanova et al.: .A Survey on the Internet of Things (IoT) Forensics: Challenges, Approaches and Open Issues”, *IEEE Communications Surveys & Tutorials*, vol. 22, n. 2, pp. 1191-1221, 2020.
[6] International Organization for Standardization: ÏSO/IEC 27037:2012 Information technology — Security techniques — Guidelines for identification, collection, acquisition and preservation of digital evidence”, 2012.
[7] International Organization for Standardization: ÏSO/IEC 27041:2015 - Information technology — Security techniques — Guidance on assuring suitability and adequacy of incident investigative methods”, 2015.
[8] ENISA: ”Good practice guide for CERTs in the area of Industrial Control Systems - Computer Emergency Response Capabilities considerations for ICS”, 2021.
[9] Unión Europea: ”Directiva (UE) 2022/2555 del Parlamento Europeo y del Consejo, de 14 de diciembre de 2022, relativa a medidas para un elevado nivel común de ciberseguridad en toda la Unión”, 2022.
[10] Unión Europea: Reglamento (UE) 2023/1543 del Parlamento Europeo y del Consejo, de 12 de julio de 2023, sobre las órdenes europeas de investigación en materia de pruebas electrónicas en procesos penales”, 2023.
[11] Consejo de Europa: ”Segundo Protocolo Adicional al Convenio sobre la Ciberdelincuencia relativo a la mejora de la cooperación y la revelación de pruebas electrónicas”, 2022.
[12] .Ansible Documentation”, 2025.
[13] Mohammad Zahid et al.: .Agentless Approach for Security Information and Event Management in Industrial IoT”, *Electronics*, vol.12, n. 8, pp. 1831, 2023.
[14] , Magnet Forensics: ”Digital Forensics Tools: The Ultimate Guide”, 2024.
[15] Cellebrite: Çellebrite”, 2025.
[16] Opentext: .OpenText EnCase Forensic”, 2025.
[17] Belkasoft: ”Belkasoft Evidence Center X”, 2025.
[18] Sleuth Kit Labs: .Autopsy User Documentation”, 2025.
[19] CAINE Project: ÇAINE Manual and Policies”, 2025.
[20] SalvationDATA: ”The Top 20 Open Source Digital Forensic Tools for 2024”, 2024.
[21] Velociraptor Team: ”Velociraptor’s Documentation site”, 2025.
[22] Z. Yuan et al.: .AutoDFBench: A Framework for AI Generated Digital Forensic Code Validation”, *Proceedings of the 2025 ACM Conference on Digital Forensics*, 2025.
[23] Abdullah Ayub Khan et al.: ”MF-Ledger: Blockchain Hyperledger Sawtooth-Enabled Novel and Secure Multimedia Chain of Custody Forensic Investigation Architecture”, *IEEE Access*, vol. 9, pp. 103637-103650, 2021.
[24] Christopher Hargreaves et al.: ”SOLVE-IT: A proposed digital forensic knowledge base inspired by investigation techniques”, *Forensic Science International: Digital Investigation*, 2025.
[25] N. Zubair et al.: ”PEM: Remote forensic acquisition of PLC memory in industrial control systems”, *Forensic Science International: Digital Investigation*, vol. 40, n. 04, 2022.
[26] Anastasis Keliris y Michail Maniatakos: ÏCSREF: A Framework for Automated Reverse Engineering of Industrial Control Systems Binaries”, *26th Network and Distributed System Security Symposium (NDSS)*, 2019.
[27] New York State Energy Research and Development Authority (NYSERDA): Ïndustrial Control Systems Automated Digital Forensics Harvester”, *Prepared by Electric Power Research Institute (EPRI)*, 2022.
[28] INCIBE: .EI IEC 62443-4-2, la necesidad de securizar los componentes”, 2022.
[29] Alan Hevner et al.: ”Design Science in Information Systems Research”, *Management Information Systems Quarterly*, vol. 28, pp. 75-, n. 03, 2004.
[30] Ken Peffers et al.: .A design science research methodology for information systems research”, *Journal of Management Information Systems*, vol. 24, pp. 45-77, n. 01, 2007.

Quasi-perfect (de)compression of elliptic curve points in the highly 2-adic scenario

Dimitri Koshelev
University of Lleida, Department of Mathematics
Catalonia, Spain
dimitri.koshelev@gmail.com

Jordi Pujolàs
University of Lleida, Department of Mathematics
Catalonia, Spain
jordi.pujolas@udl.cat

***Abstract*—In this short note, a new laconic, secure, and efficient (de)compression method is provided for points of any elliptic curve over any highly 2-adic finite field of large characteristic. Such fields are ubiquitous in modern elliptic curve cryptography, whereas they severely slow down the conventional x-coordinate (de)compression technique. In comparison with the current state of the art in the given research direction, the new method requires neither complicated mathematical formulas nor conditions on the curve, nor reliable pseudo-random generators. Thereby, the present work is universal and much more implementation-friendly, which justifies its existence, despite the absence of interesting mathematics behind it.**

***Index Terms*—(D)DoS attacks, elliptic curve cryptography, highly 2-adic finite fields, Müller's square-root algorithm, point (de)compression**

I. Introduction

With the reader permission, the full introduction on *(de)compression of elliptic curve points* (in the *highly 2-adic* setting) is omitted to avoid repetitions with the fresh comprehensive article [1]. All the necessary details on the topic can be found in that article and especially in its introduction. Nevertheless, let's survey very briefly the state of affaires.

As usual, we are given an elliptic curve $E\colon y^2 = f(x) := x^3 + ax + b$ over a finite field $\mathbb{F}_q$ of large characteristic. In the past, *ECC (elliptic curve cryptography)* used to be mostly employed when the 2-adicity $\nu := \nu_2(q-1)$ is small or even $\nu \leqslant 2$. Therefore, the classical x-coordinate (de)compression method [2, Appendix D.2.1] was an ideal solution for compact point representation. Indeed, the follow-up decompression stage extracts $y = \sqrt{f(x)} \in \mathbb{F}_q$, which is readily done by *Tonelli–Shanks' algorithm* [3] or just by one exponentiation in $\mathbb{F}_q$. However, the base fields for many modern elliptic curves (see, e.g., [4]) are often highly 2-adic (i.e., $\nu \gg 2$) owing to advanced *zk-SNARK (zero knowledge succinct non-interactive argument of knowledge)* applications. As is known, Tonelli–Shanks' algorithm becomes very slow over such fields, since it requires $O(\ell + \nu^2)$ multiplications in $\mathbb{F}_q$, where $\ell := \lceil \log_2(q) \rceil$.

Alternatively, there is *Müller's algorithm* [5] (look also at [6]) whose cost is close to that of a general field exponentiation (notably for the large ν), namely $\approx 2\ell - \nu$ multiplications in $\mathbb{F}_q$. Unfortunately, this algorithm does not function in constant time, which implies that in some rare situations it "freezes" for a quite long time. In a nutshell, [1, Section 3.1] explains how to leverage the given effect to mount a *(D)DoS (distributed denial-of-service) attack* on a decompressor based on the naive usage of Müller's algorithm. By the way, on the Internet page [7] there is a short discussion about the analogous issue in the hash-to-curve setting, although it can be seemingly fixed by means of a resilient hash function. In addition, [1, Section 3.2] proposes an efficient countermeasure (following a similar idea as in [8]), namely novel (de)compression for which Müller's algorithm turns out to be completely deterministic. Nonetheless, the invented technique is not universal: It is relevant if and only if the order of the group $E(\mathbb{F}_q)$ is even. Thus, a lot of useful prime-order curves (including NIST P-224 [2, Section 3.2.1.2] and MNT 2-cycles [9]) remain uncovered.

Recall that Müller's algorithm of finding $\sqrt{f(x)}$ needs an additional value $u \in \mathbb{F}_q^*$ such that $g(x,u) := u^2 - f(x)$ is a quadratic non-residue in $\mathbb{F}_q$. Searching for such u makes the execution variable-time and so long-time for a "poisoned" value of x. In order to treat this trouble, it is reasonable to iterate u prior to sending x. Fortunately, the non-constant-time behavior of a compressor does not pose any threat. In other words, it is impossible to prepare a (D)DoS attack as earlier, since the compression stage (unlike the decompression one) is independent of any data received from a public channel. In particular, a generator for the next values of u must not be cryptographically strong (as one of the solutions from [1, Table 1]). Thereby, there is no additional risk of relying mistakenly on a predictable generator. In fact, one can simply increment $u := u + 1$, starting with a certain fixed value $u_0 \in \mathbb{F}_q^*$ (e.g., $u_0 = 1$), as it is frequently done for hashing to elliptic curves without secret inputs [10, Section 3.2] or for generating a transparent point basis [11, Section 5.1], [12] with a view towards multi-scalar multiplication.

Hereafter, $u = u_0 + i$ for $i \in \mathbb{N}_{<2^m} := [0, 2^m) \cap \mathbb{N}$, where $m \in \mathbb{N}$ is an auxiliary parameter. According to [5, Theorem 3.5], the probability of being a (non-)square for $g(x,u)$ amounts to $\approx 1/2$ for the random u and for any x such that $f(x) \neq 0$. Hence, the new compression fails with probability $\approx 1/2^{2^m}$ for the general x. At the same time, it is necessary to transmit/store not only this coordinate but also the index i, not to mention one bit for the sign $\pm$ of the coordinate y. To sum up, $L := \ell + m + 1$ bits are required for compressing a point from $E(\mathbb{F}_q) \setminus E[2]$. We thus get a kind of trade-off. It is satisfactory in the sense that m is relatively negligible compared to ℓ of cryptographic size, i.e., $\ell \approx L$. For instance, already for $m = 7$, we achieve the standard 128-bit security level, that is, the probability of solving the discrete logarithm problem on E is not less. On the other hand, the most popular types of computer networks/memory operate with bytes rather than bits. So, the case $m + 1 = 8$ ideally fits them whenever $8 \mid \ell$, which is usually true in

practice (e.g., for NIST P-224). In this respect, $\ell + 1$ bits do not constitute a more optimal representation than L bits at least if we are not talking about point packets.

The compression and decompression under consideration are formalized in Algorithms 1, 2, respectively. As is customary, $\mathrm{sign} : \mathbb{F}_q^* \to \mathbb{F}_2$ is an arbitrary cheap function such that $\mathrm{sign}(y) \neq \mathrm{sign}(-y)$. Besides, $\left(\frac{g}{q}\right) = g^{(q-1)/2}$ stands for the Legendre symbol[1] in $\mathbb{F}_q$ and $M(x, u)$ does for Müller's algorithm of extracting the square root of $f(x)$ with the help of u such that $\left(\frac{g(x,u)}{q}\right) = -1$. The bit complexity of determining the Legendre symbol is known to be only (sub-)quadratic in ℓ due to the *(binary) Euclidean algorithm* [13, Section 12.3]. Consequently, the compressor succeeds in finding promptly the desired u with overwhelming probability (unless the parameter $m \approx 0$). To be more precise, the given entity computes on average two $\left(\frac{\cdot}{q}\right)$ as the total overhead, while the decompressor calls one $M(\cdot, \cdot)$ apart from exactly two $\left(\frac{\cdot}{q}\right)$. Finally, the maximum four 2-torsion $\mathbb{F}_q$-points on E can be separately processed by adding to L one more bit[2]. We do not elaborate on this, since small-order points do not occur in ECC protocols by trivial security reasons.

Algorithm 1: New point compression

Data: A point $(x, y) \in E(\mathbb{F}_q) \setminus E[2]$;
Result: The triple $\mathrm{com}(x, y) \in \mathbb{F}_q \times \mathbb{N}_{<2^m} \times \mathbb{F}_2$;
begin
  $f := y^2$;
  $u := u_0$;
  for $i := 0$ **to** $2^m - 1$ **do**
    $g := u^2 - f$;
    if $\left(\frac{g}{q}\right) \in \{-1, 0\}$ **then**
      $\beta := \mathrm{sign}(y)$;
      return (x, i, β).
    end
    $u := u + 1$;
  end
  return fail.
end

ACKNOWLEDGEMENTS

This research is a result of the strategic project "Advances in post-quantum cryptography applied to the development of a coupon system" (C039/24), resulting from an agreement between the Spanish National Cybersecurity Institute (INCIBE) and University of Lleida. This initiative is carried out in the scope of the funds from the Recovery, Transformation and Resilience Plan, funded by the European Union (Next Generation). The paper is also part of the R&D+i project PID2021-124613OB-I00 funded by MICIU/AEI/10.13039/501100011033 and FEDER, EU. Besides, the first author was supported by Ethereum Foundation through the grant FY25-1896.

[1]This symbol is well defined for any odd prime power q, having nothing to do with the Jacobi symbol.

[2]In reality, L bits are sufficient, because $L > \ell$ or, equivalently, $2^L \gg \#E(\mathbb{F}_q)$ and thereby the points of $E(\mathbb{F}_q)[2]$ can be easily codified by any quartet (duet or one) of the unoccupied L-bit strings.

Algorithm 2: New point decompression

Data: A triple $(x, i, \beta) \in \mathbb{F}_q \times \mathbb{N}_{<2^m} \times \mathbb{F}_2$;
Result: The point $(x, y) \in E(\mathbb{F}_q) \setminus E[2]$ such that $\mathrm{com}(x, y) = (x, i, \beta)$;
begin
  $f := x^3 + ax + b$;
  $u := u_0 + i$;
  $g := u^2 - f$;
  if $\left(\frac{f}{q}\right) \in \{-1, 0\}$ **or** $\left(\frac{g}{q}\right) = 1$ **then**
    return fail.
  end
  if $g = 0$ **then**
    $y := u$;
  else
    $y := M(x, u)$;
  end
  if $\mathrm{sign}(y) \neq \beta$ **then**
    $y := -y$;
  end
  return (x, y).
end

REFERENCES

[1] D. Koshelev, "Point (de)compression for elliptic curves over highly 2-adic finite fields," *Advances in Mathematics of Communications*, vol. 19, no. 5, pp. 1539–1559, 2025. [Online]. Available: https://doi.org/10.3934/amc.2025008

[2] L. Chen, D. Moody, A. Regenscheid, A. Robinson, and K. Randall, "Recommendations for discrete logarithm-based cryptography: elliptic curve domain parameters (NIST Special Publication 800-186)," 2023. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-186/final

[3] D. Shanks, "Five number-theoretic algorithms," in *Proceedings of the Second Manitoba Conference on Numerical Mathematics*, ser. Congressus Numerantium, R. S. D. Thomas and H. C. Williams, Eds., vol. 7. Winnipeg: Utilitas Mathematica Publishing Inc., 1973, pp. 51–70.

[4] D. F. Aranha, Y. El Housni, and A. Guillevic, "A survey of elliptic curves for proof systems," *Designs, Codes and Cryptography*, vol. 91, no. 11, pp. 3333–3378, 2023. [Online]. Available: https://doi.org/10.1007/s10623-022-01135-y

[5] S. Müller, "On the computation of square roots in finite fields," *Designs, Codes and Cryptography*, vol. 31, no. 3, pp. 301–312, 2004. [Online]. Available: https://doi.org/10.1023/B:DESI.0000015890.44831.e2

[6] R. J. Lambert, "Method to calculate square roots for elliptic curve cryptography," 2013, United States patent No. 9148282B2. [Online]. Available: https://patents.google.com/patent/US9148282B2/en

[7] C. C. Liang, "Non-constant time hash to point attack vector," 2020. [Online]. Available: https://github.com/thehubbleproject/hubble-contracts/issues/171

[8] D. Koshelev, "Hashing to elliptic curves through Cipolla–Lehmer–Müller's square root algorithm," *Journal of Cryptology*, vol. 37, no. 2, p. article 11, 2024. [Online]. Available: https://doi.org/10.1007/s00145-024-09490-w

[9] A. Guillevic, "Pairing-friendly curves," 2021. [Online]. Available: https://members.loria.fr/AGuillevic/pairing-friendly-curves

[10] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297–319, 2004. [Online]. Available: https://doi.org/10.1007/s00145-004-0314-9

[11] J. Baylina and M. Bellés, "4-bit window Pedersen hash on the Baby Jubjub elliptic curve," 2019. [Online]. Available: https://iden3-docs.readthedocs.io/en/latest/_downloads/4b929e0f96aef77b75bb5cfc0f832151/Pedersen-Hash.pdf

[12] D. Koshelev, "Generation of two "independent" points on an elliptic curve of j-invariant $\neq 0, 1728$," 2023. [Online]. Available: https://eprint.iacr.org/2023/785

[13] V. Shoup, *A computational introduction to number theory and algebra*, 2nd ed. Cambridge: Cambridge University Press, 2008. [Online]. Available: https://doi.org/10.1017/CBO9780511814549

PAXPORT: Privacy Pitfalls in the Spanish Age Verification System

David Balbás
IMDEA Software Institute
david.balbas@imdea.org

Diego Castejón-Molina
IMDEA Software Institute
Universidad Politécnica de Madrid
diego.castejon@imdea.org

***Abstract*—The European Digital Identity (EUDI) wallet is intended to enable privacy-preserving access to online services across the EU. Spain's first deployed protocol for age-restricted content, referred to here as PAXPORT, uses government-issued, single-use verifiable credentials that enable users to prove to streaming platforms that they are over 18. In line with previous analysis of EUDI wallet proposals, we argue that PAXPORT fails to provide meaningful privacy guarantees: credential issuance and presentation remain linkable, enabling government-level traceability and user deanonymization. The protocol also presents usability and scalability issues, making it a deficient template for future implementations of the EUDI wallet. We present a minimal modification, replacing standard signatures with partially blind signatures, that guarantees unlinkability. While this fix is practical, we argue that future deployments must adopt anonymous credential-based solutions, ideally incorporating zero-knowledge proofs, to meet the stated privacy goals of the EUDI initiative.**

***Index Terms*—privacy, anonymous credentials, verifiable credentials, digital identity, age verification**

I. INTRODUCTION

The European Union (EU) is moving towards a more widespread adoption of European digital identities to regulate the access to content and services in the internet. For this purpose, the EU has recently introduced an interoperable digital identity framework called the European Digital Identity (EUDI) wallet [21]. These initiatives aim to empower users with control over their identity data while prioritizing usability and user privacy.

The government of Spain recently presented the first protocol that relies on the EUDI wallet to restrict access to adult content streaming services only to adults. In this work, we brand this protocol as PAXPORT. PAXPORT involves four parties: the government (*registration authority*, a *credential issuer*) (which is, in practice, also the government), the citizen (*user)* and the streaming platform (*service provider*). Each citizen register their digital identity with the government, and can use this identity to authenticate and receive a batch of single-use verifiable credentials that prove that their age is over 18. At a later stage, when they connect to the adult content streaming service, the platform requires them to present a valid verifiable credential. If the credential is valid and has not expired (credentials are initially planned to expire after 30 days), the user is granted access to the adult content. Otherwise, access is denied.

Given that this protocol is the first European attempt to *regulate* the access to digital services and content, it may be used in both Spain and in other EU countries as a template for credential issuance. Our main goal in this paper is to explain why this is not a good idea. PAXPORT, as presented by the Spanish government, has a major flaw that should be avoided in future iterations: it enables traceability of verifiable credentials and deanonymization of users. We also argue that the current solution falls short in several other aspects that hinder its large-scale deployment. Most of our observations go in line with similar issues that have been raised by the cryptographic community [19] on different design proposals for the EUDI wallet.

Our paper is structured as follows. First, we describe the PAXPORT protocol from a cryptographic standpoint, introduce the system goals, and explain the main design flaw. Second, we show how, with minor adjustments, the same protocol can be enhanced to provide unlinkability, preventing the government from tracking citizens' actions and minimizing the privacy loss caused by potential data breaches. We do so by leveraging partially blind signature schemes, which can readily replace standard signatures proposed in PAXPORT for credential issuance. Third, we analyse the limitations of our solution and discuss its framing within the broader EUDI wallet framework. We remark that our solution is meant to be a simple patch that allows to deploy a variant of PAXPORT that prevents user tracing by the government. We strongly suggest that legislators should instead direct their efforts to implement a solution based on anonymous credentials, which are a well-researched topic [16], [3], [8], [5], [10], [13], [1], [11], [15], [20].

II. PAXPORT AND ITS PRIVACY PITFALL

In this section, we provide an overview of the PAXPORT ecosystem, as described in [18]. Our analysis focuses on the cryptographic aspects of PAXPORT. Similar to anonymous credential systems [8], PAXPORT involves four parties:

- **User**: The party holding a physical, state-provided identity.
- **Registration authority**: The party responsible for registering the digital identity of *users* who possess a valid, state-provided physical identity.
- **Credential issuer**: The party that issues a digital credential indicating whether a registered *user* is an adult.
- **Service provider**: The party that grants access to a specific service only to those users who hold a credential confirming that they are adults.

In PAXPORT, the registration authority and the credential issuer are both controlled by the government, although this may not be the case for all credentials. For instance, a university may give credentials to those users who graduate. PAXPORT involves three interactive protocols between the aforementioned parties, that we denote as *identity registration*, *credential issuance* and *credential presentation*. The latter two

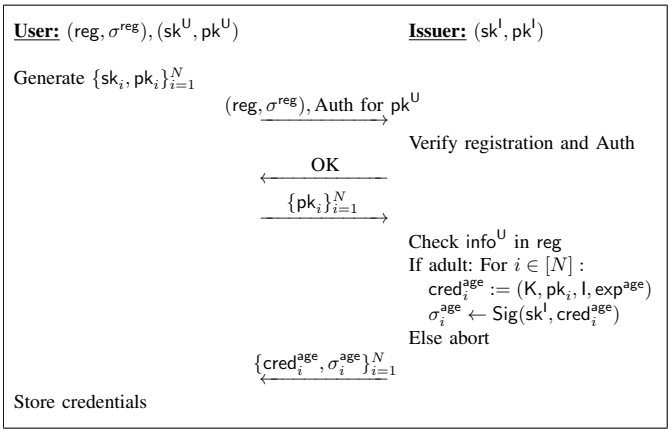


Figure 1. Age credential issuance protocol.

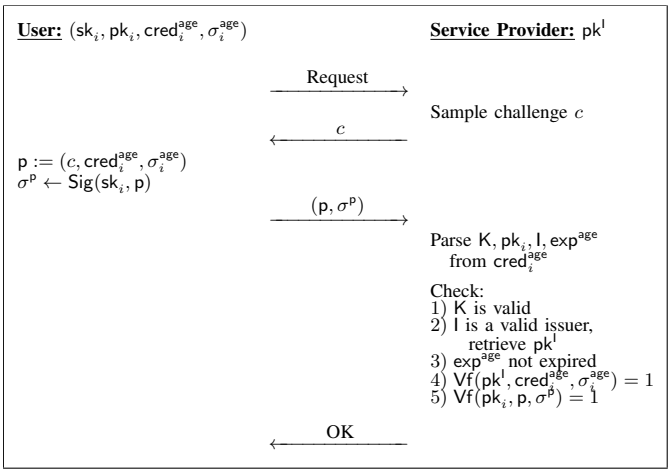


Figure 2. Age credential presentation protocol.

follow the OID4VCI and OID4VP standards for issuance and presentation of credentials, respectively. [24]

Identity registration. In Spain, everyone aged fourteen or older are required to possess a state-provided identity card called the *Documento Nacional de Identidad (DNI)*. The DNI includes a secure enclave (SE) capable of generating key pairs and signing messages. The DNI requires a password to compute a signature or to generate a new key pair. The password can only be modified using the biometric data stored on the card. The registration authority holds a key pair denoted $(\mathsf{sk}^{\mathsf{RA}}, \mathsf{pk}^{\mathsf{RA}})$ To register the key pair $(\mathsf{sk}^{\mathsf{U}}, \mathsf{pk}^{\mathsf{U}})$ in the DNI:

1) The user physically goes to a police station and inserts the DNI into a registration machine. The registration machine prompts the user to authenticate using their password.
2) The SE of the DNI provides the public key pk^{U} and certain user information $\mathsf{info}^{\mathsf{U}}$ (e.g., name, surname, date of birth, address) to the registration machine.
3) The registration machine prepares a certificate $\mathsf{reg} := (\mathsf{info}^{\mathsf{U}}, \mathsf{pk}^{\mathsf{U}}, \mathsf{RA}, \exp^{\mathsf{reg}})$ containing the user's information $\mathsf{info}^{\mathsf{U}}$, the public key pk^{U}, the registration authority identity RA, and the expiration date $\exp^{\mathsf{reg}}$, and produces a signature σ^{reg} using $\mathsf{sk}^{\mathsf{RA}}$.
4) The tuple $(\mathsf{reg}, \sigma^{\mathsf{reg}})$ is stored in the SE of the DNI, completing the identity registration process.

Credential issuance. The user, owning a registration certificate $(\mathsf{reg}, \sigma^{\mathsf{reg}})$ and key pair $(\mathsf{sk}^{\mathsf{U}}, \mathsf{pk}^{\mathsf{U}})$, connects to the age credential issuer to obtain a credential proving that they are an adult. The issuer holds a key pair denoted as $(\mathsf{sk}^{\mathsf{I}}, \mathsf{pk}^{\mathsf{I}})$. The credential issuance proceeds as follows (we outline the protocol flow in Figure 1):

1) The user generates N key pairs, denoted as $\{(\mathsf{sk}_i, \mathsf{pk}_i)\}_{i=1}^N$.
2) The user connects to the issuer's server. To authenticate, the issuer requires $(\mathsf{reg}, \sigma^{\mathsf{reg}})$ and a proof that the user knows sk^{U}. This is done via a standard challenge-response authentication protocol where the user signs a challenge sent by the issuer.
3) Upon successful authentication, the user sends the batch of public keys generated in step (1), $\{\mathsf{pk}_i\}_{i=1}^N$.
4) The issuer extracts the date of birth from the user's information field, $\mathsf{info}^{\mathsf{U}}$, in reg. If the user is an adult, then for each $i \in [1, N]$, the issuer builds the age verification credential as $\mathsf{cred}_i^{\mathsf{age}} := (\mathsf{K}, \mathsf{pk}_i, \mathsf{I}, \exp^{\mathsf{age}})$, where K is an identifier for age verification credential (known by all parties a-priori) following the description in [18], pk_i is the ephemeral public key sent by the user, I is the issuer's identity, and $\exp^{\mathsf{age}}$ is the credential's expiration date.
5) The issuer signs each credential using sk^{I}, such that $\sigma_i^{\mathsf{age}} \leftarrow \mathsf{Sig}(\mathsf{sk}^{\mathsf{I}}, \mathsf{cred}_i^{\mathsf{age}})$ and returns the set $\{\mathsf{cred}_i^{\mathsf{age}}, \sigma_i^{\mathsf{age}}\}_{i=1}^N$ to the user, finalizing the credential issuance.

Credential presentation. Consider a user that owns a credential key pair $(\mathsf{sk}_i, \mathsf{pk}_i)$ and an age-verification credential $(\mathsf{cred}_i^{\mathsf{age}}, \sigma_i^{\mathsf{age}})$. To access a service that require age verification, user and service provider proceed as follows (we outline the protocol flow in Figure 2):

1) The user connects to the server and the service provider sends a random challenge c.
2) The user produces the presentation, $\mathsf{p} := (c, \mathsf{cred}_i^{\mathsf{age}}, \sigma_i^{\mathsf{age}})$ and signs it using their secret key, $\sigma^{\mathsf{p}} \leftarrow \mathsf{Sig}(\mathsf{sk}_i, \mathsf{p})$.
3) The user sends the tuple $(\mathsf{p}, \sigma^{\mathsf{p}})$ to the service provider and deletes the credential $(\mathsf{pk}_i, \mathsf{sk}_i, \mathsf{cred}_i^{\mathsf{age}}, \sigma_i^{\mathsf{age}})$.[1]
4) The service provider parses K, pk_i, I, and $\exp^{\mathsf{age}}$ from $\mathsf{cred}_i^{\mathsf{age}}$ and checks that:
   a) K is the identifier of the age-verification credential.
   b) I is the identifier of the age-verification issuer, and retrieves the public key pk^{I}.
   c) The credential expiration date $\exp^{\mathsf{age}}$ has not passed.
   d) The credential is valid, $\mathsf{Vf}(\mathsf{pk}^{\mathsf{I}}, \mathsf{cred}_i^{\mathsf{age}}, \sigma_i^{\mathsf{age}}) = 1$
   e) The presentation is valid, $\mathsf{Vf}(\mathsf{pk}_i, \mathsf{p}, \sigma^{\mathsf{p}}) = 1$.
5) If all checks pass, user is granted access to the service.

Expiration dates are intended to be short, the initial proposal being 30 days. As credentials are ephemeral, a user may run out of valid credentials before the expiration date. Thus, the credential issuance protocol should be run upon user request.

A. System Goals

The specification available for PAXPORT [18] does not convey the intended security and privacy goals. Based on the specifications, and the eIDAS regulation [21], we infer the goals and why the proponents would argue that they hold.

[1]The deletion of the credential at this step is not mentioned in PAXPORT specifications, but we observe that it is needed for forward security, since reusing the same credential may lead to session correlation attacks.

- **Registration unforgeability**: No party other than the registration authority can issue registrations certificates. PAXPORT satisfies this notion since the RSA signature scheme used for identity registration is unforgeable under chosen message attacks.
- **Credential unforgeability**: No party other than the issuer can issue age-verification credentials. PAXPORT satisfies this notion since the EdDSA scheme used for credential issuance is unforgeable under chosen message attacks.
- **Presentation soundness**: The service provider should not convinced that a presentation is valid unless: (i) the credential was produced by the issuer and; (ii) the presentation is signed by the user. We have already established that credential unforgeability holds. The EdDSA signature scheme used by the user to sign the presentation is unforgeable under chosen message attacks. Therefore, PAXPORT satisfies presentation soundness.
- **Selective disclosure**: No information is given to the service provider except what is strictly required. Since the only information contained in the credential is whether the user is an adult or not (and not their date of birth or any other information in $\mathsf{info}^{\mathsf{U}}$), PAXPORT satisfies this property.
- **Privacy with respect to the service provider**: The service provider should neither learn the identity of the user from the presentation $(\mathsf{p}, \sigma^{\mathsf{p}})$ nor be able to link different access requests from the same user.[2] Intuitively, PAXPORT satisfies this property because:
  - The presentation does not contain the field $\mathsf{info}^{\mathsf{U}}$ that allows to identify the user.
  - The public key pk_i is ephemeral and single-use.
  - Signatures themselves do not contain any information about the user, given that the relation between a signature σ^{age} and the user is never known to the service provider. Note that such an assumption breaks if there is a collusion between the issuer and the service provider.

B. PAXPORT *enables user tracking*

The security notion underlying PAXPORT is insufficient because it considers only the information that the service provider directly receives from the user, ignoring the broader context of information available within the ecosystem. In particular, it overlooks the possibility that the credential issuer may leak information that could enable the service provider, or any other entity, to link presentations to users.

In the literature on anonymous credential systems, the usual privacy notion is called user unlinkability. This property requires that, even if the registration authority, issuer, and service provider fully collude and share all the information they possess, they still cannot determine which user generated a particular credential presentation, or in general correlate two presentations of credentials.

Definition of unlinkability. Formally defining unforgeability for PAXPORT requires to state the latter as a cryptographic primitive with a concrete syntax and to introduce a security game. This is beyond the scope of this work, but we introduce an informal game-based notion below. The game involves a challenger and an adversary $\mathcal{A}$ who proceed as follows:

[2]We remark that deanonymization may occur by other means, such as by tracking of IP addresses. However, this is out of the scope of our analysis.

1) The challenger randomly samples a challenge bit $b \leftarrow\$ \{0, 1\}$.
2) The adversary $\mathcal{A}$ has full access to the service provider, the issuer, and the registration authority. It generates keys pk^{I} and $\mathsf{pk}^{\mathsf{RA}}$ for the latter two.
3) Two users U_0, U_1 obtain a registration certificate from the registration authority.
4) Each user obtains a batch of age-verification credentials from the issuer.
5) $\mathcal{A}$ queries either u_0 or u_1 to present a credential to the service provider. $\mathcal{A}$ can make multiple queries to the users.
6) At some point, $\mathcal{A}$ issues a challenge. Then, user u_b makes a presentation to the service provider.
7) Finally (and after potentially more presentation queries from $\mathcal{A}$), the adversary outputs a bit b'.

The adversary wins the game if $b = b'$. We say that PAXPORT achieves (computational) unlinkability if the winning probability of any polynomial-time adversary is $\Pr[b = b'] = 1/2 + \mathsf{negl}(\lambda)$ for a negligible function $\mathsf{negl}(\lambda)$.

PAXPORT does not satisfy unlinkability. The system trivially allows credential presentations to be traced back to their owners through the following process:

1) During credential issuance, the issuer receives $\mathsf{info}^{\mathsf{U}}$ and $\{\mathsf{pk}_i\}_{i=1}^{N}$ from each user. The issuer can create a key-value store where each pk_i serves as a key and the associated $\mathsf{info}^{\mathsf{U}}$ as its value.
2) The service provider later receives a tuple $(\mathsf{p}, \sigma^{\mathsf{p}})$, which can be parsed as $\mathsf{p} = (c, \mathsf{cred}_i^{\mathsf{age}}, \sigma_i^{\mathsf{age}})$, where $\mathsf{cred}_i^{\mathsf{age}} = (\mathsf{K}, \mathsf{pk}_i, \mathsf{I}, \mathsf{exp}^{\mathsf{age}})$.
3) The service provider can forward pk_i to the issuer, who looks it up in the key-value store and returns the corresponding $\mathsf{info}^{\mathsf{U}}$, uniquely identifying the user.

One might argue that collusion between the issuer and the service provider is unlikely. However, the core issue with PAXPORT is that the protocol imposes *no technical barrier* preventing the issuer (i.e. the government) from creating such a key-value mapping. Even if the issuer refrains from doing so intentionally, a security breach in the credential issuance infrastructure could expose this information to external attackers, who could exploit it, for instance to blackmail users based on their use of age-verification credentials.

The fundamental problem, therefore, is that the data *exists at all* beyond the user's device. We argue that PAXPORT (and, in general, any future implementation of the EUDI wallet) should be redesigned with user unlinkability as a foundational principle, thereby eliminating all data that enables user tracing. This would protect users privacy even in the event of a severe data breach, or even if the system is misused by a potentially authoritarian government.

III. A SIMPLE FIX FOR PAXPORT

Despite the important privacy flaw in PAXPORT, there are simple ways to fix it. In this section, we introduce a fix that requires minimal changes to the scheme, relies only on standard cryptographic primitives, and can be prototyped and implemented promptly. As we argue in the discussion section, our proposal is not a fully optimal solution, but rather an example for how the current implementation can be improved with minor modifications. Optimal designs can be obtained

$\mathsf{Blind}_{\mathcal{A}}(\lambda)$

$(\mathsf{pk}, t, m_0, m_1) \leftarrow \mathcal{A}(1^\lambda), b \leftarrow\$ \{0,1\}$
$\forall a \in \{0,1\}\colon\ (\rho_U^a, \mathsf{st}_a) \leftarrow \mathsf{User}(\mathsf{pk}, t, m_a)$
$(\rho_S^b, \rho_S^{1-b}) \leftarrow \mathcal{A}(\rho_U^b, \rho_U^{1-b})$
$\forall a \in \{0,1\}\colon\ \sigma_a \leftarrow \mathsf{Derive}(\mathsf{st}_a, t, \rho_S^a)$
if $\exists a$ s.t. $\mathsf{Vf}(\mathsf{pk}, t, m_a, \sigma_a) = 0$ **then**
  $\sigma_0 = \sigma_1 = \bot$
$b' \leftarrow \mathcal{A}(\sigma_0, \sigma_1)$
return $b = b'$

Figure 3. Security game for partial blindness under chosen keys.

by leveraging zero knowledge proofs, but there are further barriers to their adoption such as a lack of standardized schemes.

Our proposal consists of modifying the credential issuance by using *partially blind digital signatures*, which are standardized in ISO/IEC 18370-2:2016 [17] and which we define below.

Definition 1. *A partially blind signature scheme [7] with tag space $\mathcal{T}$ and message space $\mathcal{M}$, is a tuple of algorithms with the following syntax, adopted from [2]:*

- *A key generation algorithm* $\mathsf{KGen}(1^\lambda)$ *that takes the security parameter λ and outputs a public/secret key pair* $(\mathsf{pk}, \mathsf{sk})$.
- *A user algorithm* $\mathsf{User}(\mathsf{pk}, t, m)$ *that inputs a public key* pk*, a tag $t \in \mathcal{T}$ and a message $m \in \mathcal{M}$ and outputs a blind user message ρ_U and a state* st.
- *A signer algorithm* $\mathsf{Signer}(\mathsf{sk}, t, \rho_U)$ *inputs a secret key* sk*, a tag $t \in \mathcal{T}$ and a user message ρ_U and outputs a signer message ρ_S.*
- *A derivation algorithm* $\mathsf{Derive}(\mathsf{st}, t, \rho_S)$ *inputs a state* st*, a tag $t \in \mathcal{T}$ and a signer message ρ_S and outputs a signature σ.*
- *A verification algorithm* $\mathsf{Vf}(\mathsf{pk}, t, m, \sigma)$ *outputs* 1 *if σ is a valid signature on m for tag t under the public key* pk*, and outputs* 0 *otherwise.*

A secure partially blind signature scheme satisfies *partial blindness* and *one-more unforgeability*. Partial blindness is a privacy notion that prevents the signer (or any observer) to link a signature σ for message m and tag t to any signing session for users using the same tag t. One-more unforgeability is a security notion that ensures that no adversary can output $k+1$ valid signatures given that they have obtained k valid signatures from the signer. We present the details for partial blindness in Figure 3, as we use it later to argue that our construction satisfies unlinkability. A signature scheme satisfies partial blindness if for any polynomial-time adversary $\mathcal{A}$ playing the game in Figure 3 it holds that $\Pr[b = b'] = 1/2 + \mathsf{negl}(\lambda)$.

Credential issuance with partially blind signatures. As in the original protocol, we assume that the user owns a registration certificate $(\mathsf{reg}, \sigma^{\mathsf{reg}})$ and key pair $(\mathsf{sk}^{\mathsf{U}}, \mathsf{pk}^{\mathsf{U}})$. The user connects to the credential issuer to obtain a credential proving that they are an adult. The issuer holds a blind signature key pair denoted as $(\mathsf{sk}^{\mathsf{I}}, \mathsf{pk}^{\mathsf{I}}) \leftarrow \mathsf{KGen}(1^\lambda)$. The credential issuance with partially blind signatures proceeds:

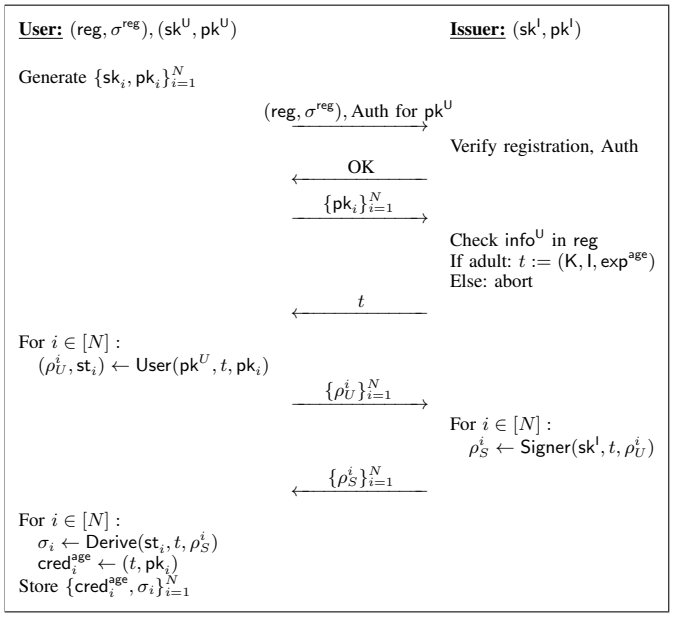


Figure 4. Credential issuance with partially blind signatures.

1) The user generates N key pairs, $\{(\mathsf{sk}_i, \mathsf{pk}_i)\}_{i=1}^N$.
2) The user connects to the issuer's server. The issuer requires $(\mathsf{reg}, \sigma^{\mathsf{reg}})$ and authenticates the user on pk^{U}.
3) Upon successful authentication, the issuer extracts the date of birth from the user's information field, $\mathsf{info}^{\mathsf{U}}$, in reg. If the user is an adult, the issuer sets a tag to $t := (\mathsf{K}, \mathsf{I}, \mathsf{exp}^{\mathsf{age}})$ and sends it to the user, or aborts otherwise. The tag is shared among all users with age-verification credentials with the same expiration date.
4) The user receives the tag t and, for each pk_i in $\{\mathsf{pk}_i\}_{i=1}^N$, computes $(\rho_U^i, \mathsf{st}_i) \leftarrow \mathsf{User}(\mathsf{pk}^{\mathsf{U}}, t, \mathsf{pk}_i)$, and sends the batch of blind messages $\{\rho_U^i\}_{i=1}^N$ to the issuer.
5) For each ρ_U^i in $\{\rho_U^i\}_{i=1}^N$, the issuer computes $\rho_S^i \leftarrow \mathsf{Signer}(\mathsf{sk}^{\mathsf{I}}, t, \rho_U^i)$, and returns the batch of blind signatures $\{\rho_S^i\}_{i=1}^N$ to the user.
6) For each ρ_S^i in $\{\rho_S^i\}_{i=1}^N$, the user computes $\sigma_i \leftarrow \mathsf{Derive}(\mathsf{st}_i, t, \rho_S^i)$. Finally, the user sets $\mathsf{cred}_i^{\mathsf{age}} := (t := (\mathsf{K}, \mathsf{I}, \mathsf{exp}^{\mathsf{age}}), \mathsf{pk}_i)$ for each pk_i in $\{\mathsf{pk}_i\}_{i=1}^N$.

We summarize the protocol in Figure 4. The protocol requires minimal modifications with respect to the original PAXPORT issuance protocol (Figure 1) and requires one additional round of interaction.

Regarding credential presentation, the service provider now parses $\mathsf{cred}_i^{\mathsf{age}} := (t, \mathsf{pk}_i)$ and checks whether $\mathsf{Vf}(\mathsf{pk}^{\mathsf{I}}, t, \mathsf{pk}_i, \sigma_i) = 1$.

A. Security Analysis

Regarding security, introducing partially blind signatures requires to revise credential unforgeability, presentation soundness and user unlinkability. Regarding credential unforgeability and presentation soundness, we argue that we are substituting a digital signature, secure under unforgeability under chosen message attacks, with a partially blind signature

scheme, secure under one-more unforgeability. Since one-more unforgeability implies unforgeability under chosen message attacks [4], [14], PAXPORT with partially blind signatures for credential issuance satisfies credential unforgeability and presentation soundness. Regarding user unlinkability, we state the property in Claim 1 and argue why it holds below.

Claim 1. *Assume that the partially blind signature scheme achieves partial blindness under chosen keys. Then, our proposed modification for* PAXPORT *satisfies computational unlinkability for all users with the same issuer* I *and same expiration date* $\mathsf{exp}^{\mathsf{age}}$.

Proof: (sketch). Assume that there is an adversary $\mathcal{A}$ that is able to win the unlinkability game for PAXPORT with a probability significantly greater than $1/2$. We show how to build an adversary $\mathcal{B}$ that uses $\mathcal{A}$ to break partial blindness under chosen keys (Figure 3). For simplicity, we let $N = 1$. Setting arbitrary N would require an additional guessing step in the proof involving a linear security loss in N.

1) $\mathcal{B}$ initializes $\mathcal{A}$, who outputs the registration authority public key $\mathsf{pk}^{\mathsf{RA}}$ and the credential issuer key pair pk^{I}.
2) $\mathcal{B}$ samples user pairs $(\mathsf{sk}^{u_0}, \mathsf{pk}^{u_0})$ and $(\mathsf{sk}^{u_1}, \mathsf{pk}^{u_1})$ and interacts with $\mathcal{A}$ for each key pair the identity registration protocol, obtaining $(\mathsf{reg}_0, \sigma^{\mathsf{reg}_0})$ and $(\mathsf{reg}_1, \sigma^{\mathsf{reg}_1})$.
3) Prior to interacting with $\mathcal{A}$ on protocol credential issuance, $\mathcal{B}$ samples key pairs $(\mathsf{pk}_0, \mathsf{sk}_0)$ and $(\mathsf{pk}_1, \mathsf{sk}_1)$.
4) $\mathcal{B}$ runs credential issuance as user 0, using $(\mathsf{reg}_0, \sigma^{\mathsf{reg}_0})$. When $\mathcal{A}$ outputs $t := (\mathsf{K}, \mathsf{I}, \mathsf{exp}^{\mathsf{age}})$ in step 3, $\mathcal{B}$ initializes the challenger of partial blindness on chosen keys by providing the tuple $(\mathsf{pk}^{\mathsf{I}}, t, \mathsf{pk}_0, \mathsf{pk}_1)$ and receives (ρ_U^0, ρ_U^1). In step 4, $\mathcal{B}$ sends ρ_U^0 to $\mathcal{A}$, receiving ρ_U^0.
5) $\mathcal{B}$ runs credential issuance as user 1, using $(\mathsf{reg}_1, \sigma^{\mathsf{reg}_1})$, and in step 4, sends ρ_U^1 to $\mathcal{A}$, receiving ρ_S^1.
6) $\mathcal{B}$ returns (ρ_S^0, ρ_S^1) to the challenger, who replies with (σ_0, σ_1). Now, $\mathcal{B}$ invokes $\mathcal{A}$ on input $(t := (\mathsf{K}, \mathsf{I}, \mathsf{exp}^{\mathsf{age}}), \mathsf{pk}_0, \sigma_0)$, receiving a bit b'.
7) $\mathcal{B}$ forwards the bit b' to the challenger.

It is easy to see that $\mathcal{B}$ is a polynomial-time algorithm and that it perfectly simulates the unlinkability game to $\mathcal{A}$. If $\mathcal{A}$ makes a correct guess for in which of the two credential issuance processes pk_0 was used, then the bit b' forwarded to the challenger is also a correct guess for the shuffling of the blind messages. Hence, $\mathcal{B}$ has the same success probability than $\mathcal{A}$, breaking partial blindness for chosen keys.

We conclude that if the partially blind signature scheme satisfies partial blindness under chosen keys, then PAXPORT achieves user unlinkability. ∎

Computational and statistical unlinkability. In some anonymous credential schemes, unlinkability is achieved unconditionally, meaning that user privacy will be preserved even if other security properties, such as unforgeability, are broken by powerful or future adversaries. In practice, this is a great feature to aim for, as it guarantees the preservation of user privacy against quantum adversaries, against a break of the underlying cryptography, or even against a leakage of the secret keys. In our case, this depends on the choice of the partially blind signature scheme. If the latter satisfies statistical partial blindness under chosen keys, then our proposed modification of PAXPORT will satisfy statistical unlinkability too.

IV. DISCUSSION

A. Framing within the EU Digital Identity Wallet

As we mention, the PAXPORT solution can be framed as a preliminary and bounded-scope implementation of the EUDI wallet. While PAXPORT is only capable of proving one statement, that is "I am over 18", the EUDI wallet should support more complex statements over different user attributes, such as the possession of a valid driving licence, the (lack of) a criminal record, or the possession of an academic degree. Current proposals for implementations of the EUDI Wallet rely on batch issuance of verifiable credentials, similarly to PAXPORT, and therefore suffer from the same privacy pitfalls. Indeed, these solutions enable selective disclosure but fail to provide unlinkability [23], in the event of a collusion between issuers and verifiers, offering weak privacy guarantees for users.

Privacy is not the only technical concern in the current PAXPORT and other EUDI Wallet designs. There are several other issues with the current solutions, including but not limited to:

- *Non-transferability and device binding.* Credentials should be linked to a device and never be shared with other users. This is not an explicit security goal of the PAXPORT technical specification, but it seems critical to it. In practice, key exfiltration can be prevented (under some assumptions) by storing keys in secure hardware enclaves, also minimizing the risk of security breaches.
- *Storage of cryptographic keys.* For solutions that employ secure enclaves, we note that the storage of these enclaves is reduced, and flooding them with tens or hundreds of one-time keys is not ideal. Besides, hardware attestation for these keys may introduce linkability issues [19].
- *Denial of credentials.* These designs rely on a central party, i.e., the government, to operate as the registration authority and credential issuer. As such, a malicious government could arbitrarily deny digital identities to and credentials to specific citizens, preventing their access to any service that requires EUDI credentials.
- *Server overhead.* The solution requires a frequent interaction with the issuer's server due to the short living time of the one-time credentials. This puts a large overhead in the server and a temporary server unavailability may have a significant impact in the service.
- *Issuer disclosure.* Hiding the issuer's identity from the service provider is not possible, as it is part of the tag. In multi-issuer settings (such as the EU with the multiple member states and institutions), disclosing the issuer may be a significant source of information which is not needed by the service provider.
- *Attestation of complex statements.* In the event of having credentials with multiple attributes, these have to be either fully disclosed or not disclosed at all. This prevents users from attesting complex more statements while disclosing the minimal amount of information. For example, an age credential cannot be used to prove "I am over 65" unless one includes a specific attribute for this (or unless the full date of birth is disclosed, deanonymizing the user). It is also impossible to prove disjunctive statements such as "I have a bachelor's degree or a superior technical diploma".

Limitations of our solution. As mentioned, while our solution does fix the main privacy flaw of PAXPORT, it is still suboptimal and suffers from most of the limitations we just mentioned. The issuer anonymity limitation is the simplest one to fix: we can simply use partially blind ring signatures [9] where the ring of signers consists of all valid issuers in the EU. In addition, denial of credentials can be also mitigated by accepting registration authorities and credential issuers other than the government e.g., a certificate or a credential signed by an independent notary. We also note that the security of our solution has not been analyzed with sufficient rigour, as we did not develop a formal cryptographic modelling with corresponding security proofs.

B. The golden standard: anonymous credential systems.

Following many rounds of interaction between the academic community and the discussion group for the EUDI Wallet [19], the latter now states an interest in leveraging advanced cryptography, in particular anonymous credential-based systems [6]. These systems are designed with unlinkability as a foundational principle, preventing the design flaws of the current schemes. We argue that this is the right design pattern to pursue for PAXPORT and for the EUDI Wallet; while our proposed fix is simple and practical, these systems should be redesigned more deeply in future iterations.

Among the possible ways of realizing anonymous credentials, the EUDI Wallet is currently evaluating systems based on zero-knowledge proofs [22]. Zero-knowledge proofs [12] allow users to prove statements without revealing any additional information beyond the truth of the statement itself. Such protocols enable holders can generate privacy-preserving proofs that a certain predicate on their credential holds, as well as proving the possession of a valid credential without disclosing it. Nevertheless, there exist multiple challenges towards deploying these systems at scale. These include requirements to rely only on standardized cryptography, lack of secure hardware support for pairing-based cryptography, and efficiency and security trade-offs of proof systems and their implementations. The deployment of zero-knowledge proofs within the EUDI framework is thus expected to be slow, but we encourage both researchers and practitioners to support this or any effort to migrate to an anonymous credential-based system, moving away from the single-use verifiable credential setting.

ACKNOWLEDGEMENTS

This work is supported by the PICOCRYPT project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 101001283), partially supported by projects PRODIGY (TED2021-132464B-I00), ESPADA (PID2022-142290OB-I00), and EX2024-001471-M funded by MCIN/AEI/10.13039/501100011033/ and the European Union NextGenerationEU / PRTR, and partially funded by Ministerio de Universidades (FPU21/00600).

REFERENCES

[1] Omid Mir and Balthazar Bauer and Scott Griffy and Anna Lysyanskaya and Daniel Slamanig. "Aggregate Signatures with Versatile Randomization and Issuer-Hiding Multi-Authority Anonymous Credentials".

[2] Shuichi Katsumata and Michael Reichle and Yusuke Sakai. "Practical Round-Optimal Blind Signatures in the ROM".

[3] Jan Camenisch and Maria Dubovitskaya and Kristiyan Haralambiev and Markulf Kohlweiss. "Composable and Modular Anonymous Credentials: Definitions and Practical Constructions".

[4] David Pointcheval and Jacques Stern. "Provably Secure Blind Signature Schemes".

[5] Jan Camenisch and Anna Lysyanskaya. "Signature Schemes and Anonymous Credentials from Bilinear Maps".

[6] Jan Camenisch and Anna Lysyanskaya. "An Identity Escrow Scheme with Appointed Verifiers".

[7] Masayuki Abe and Tatsuaki Okamoto. "Provably Secure Partially Blind Signatures".

[8] Jan Camenisch and Anna Lysyanskaya. "An Efficient System for Nontransferable Anonymous Credentials with Optional Anonymity Revocation".

[9] Javier Herranz and Fabien Laguillaumie. "Blind Ring Signatures Secure Under the Chosen-Target-CDH".

[10] Aisling Connolly and Pascal Lafourcade and Octavio Perez-Kempner. "Improved Constructions of Anonymous Credentials from Structure-Preserving Signatures on Equivalence Classes".

[11] Olivier Sanders. "Efficient Redactable Signature and Application to Anonymous Credentials".

[12] Shafi Goldwasser and Silvio Micali and Charles Rackoff. "The Knowledge Complexity of Interactive Proof-Systems (Extended Abstract)".

[13] Georg Fuchsbauer and Christian Hanser and Daniel Slamanig. "Structure-Preserving Signatures on Equivalence Classes and Constant-Size Anonymous Credentials".

[14] David Pointcheval and Jacques Stern. "Security Arguments for Digital Signatures and Blind Signatures".

[15] Olivier Sanders and Jacques Traoré. "Compact Issuer-Hiding Authentication, Application to Anonymous Credential".

[16] Daniel Bosk and Davide Frey and Mathieu Gestin and Guillaume Piolle. "Hidden Issuer Anonymous Credential".

[17] "Information technology — Security techniques — Blind digital signatures — Part 2: Discrete logarithm based mechanisms". (2016).

[18] Ministerio para la Transformacion Digital y de la Funcion Publica. "Presentacion del ecosistema de verificacion de edad". (2024). https://digital.gob.es/especificaciones_tecnicas

[19] Carsten Baum et al.. "Cryptographers' Feedback on the EU Digital Identity's ARF". (2024).

[20] Rebekah Mercer and Kaoutar El Khiyaoui and Angelo De Caro and Elli Androulaki. "Efficient Aggregate Anonymous Credentials for Decentralized Identity". (2025). https://eprint.iacr.org/2025/1724

[21] European Parliament and Council of the European Union. "Regulation (EU) 2024/1183 amending Regulation (EU) 910/2014 as regards establishing the European Digital Identity Framework". (2024). https://eur-lex.europa.eu/eli/reg/2024/1183/oj

[22] "Specification for ZKP Implementation in EUDI Wallet". (2025). https://github.com/eu-digital-identity-wallet/eudi-doc-standards-and-technical-specifications/blob/main/docs/technical-specifications/ts4-zkp.md

[23] Anja Lehmann. "EU Digital Identity and Anonymous Credentials - A Happy End?".

[24] OpenID Digital Credentials Protocols Workgroup. "OpenID for Verifiable Presentations 1.0".

Smart Rural IoT Lab: Laboratorio experimental para la evaluación de la ciberseguridad en entornos rurales inteligentes

Daniel Sernández-Iglesias*
Universidad Nacional de Educación a Distancia (UNED)
Programa de Doctorado en Tecnologías Industriales (EIDUNED)
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
dsernandez@scc.uned.es

Llanos Tobarra
Universidad Nacional de Educación a Distancia (UNED)
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
llanos@scc.uned.es

Rafael Pastor-Vargas
Universidad Nacional de Educación a Distancia (UNED)
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
rpastor@scc.uned.es

Antonio Robles-Gómez
Universidad Nacional de Educación a Distancia (UNED)
Departamento de Sistemas de Comunicación y Control, ETSI Informática
Madrid, Spain
arobles@scc.uned.es

Pedro Vidal-Balboa
Consorcio CA UNED en Ponferrada / INTECCA
Ponferrada, Spain
pvidal@intecca.uned.es

João Sarraipa
Instituto de Desenvolvimento de Novas Tecnologias (UNINOVA)
Caparica, Portugal
jfss@uninova.pt

***Resumen*—Las zonas rurales presentan importantes barreras para la adopción de tecnologías IoT debido a la baja conectividad, escasa infraestructura y recursos técnicos limitados. Este trabajo resume los resultados presentados previamente [1], donde se introdujo el *Smart Rural IoT Lab*, una plataforma modular y reproducible orientada a la investigación en ciberseguridad y comunicaciones para entornos rurales inteligentes. El laboratorio integra hardware de bajo coste y herramientas de código abierto para desplegar escenarios basados en LoRaWAN, MQTT, HTTP, UDP y CoAP, permitiendo la captura de tráfico real y la generación de datasets. Se presentan seis escenarios experimentales que reflejan casos de uso rurales (monitorización ambiental, control de accesos o seguimiento de activos) y se analiza el conjunto de datos generado mediante Tcpdump y Zeek. Los resultados validan la viabilidad de arquitecturas autónomas y sin dependencia de la nube, contribuyendo a la investigación en ciberseguridad, edge computing e inteligencia artificial aplicada a entornos rurales.**

***Index Terms*—IoT, smart rural, laboratorio IoT, redes de sensores, dataset**

I. Introducción

El crecimiento del IoT ha impulsado soluciones de automatización, sensorización y control remoto en sectores industriales y urbanos. Sin embargo, la traslación del modelo *Smart City* al medio rural sigue siendo incompleta. Las denominadas *Smart Villages* [2], [3] buscan replicar las ventajas de la digitalización en comunidades rurales, pero su aplicación real continúa centrada en núcleos habitados, sin atender zonas despobladas o de difícil acceso.

En estos entornos, la ausencia de cobertura estable, la escasez de infraestructura y la falta de personal in situ agravan el riesgo de incidentes de seguridad que afectan a servicios esenciales como el riego, la ganadería o la conservación del patrimonio [4], [5]. Los ataques de denegación de servicio o la manipulación de sensores pueden causar daños económicos y medioambientales, evidenciando la necesidad de sistemas IoT resilientes y seguros.

La *Tabla I* resume las diferencias clave entre las implementaciones típicas de IoT urbanas y rurales, mostrando por qué se requieren enfoques distintos para garantizar un diseño, pruebas y seguridad efectivos en las implementaciones rurales.

Estas diferencias justifican el desarrollo de infraestructuras experimentales específicas. En este contexto, nuestro trabajo previo [1] presentó el *Smart Rural IoT Lab*, que aborda esta brecha mediante una arquitectura modular que combina tecnologías de largo y corto alcance (LoRaWAN y WiFi) con herramientas abiertas (Node-RED, InfluxDB, MongoDB, Grafana y Docker) para garantizar autonomía, replicabilidad e independencia de servicios en la nube. Su diseño permite estudiar la eficacia de diferentes protocolos y metodologías de seguridad en escenarios reales y controlados. Además, el laboratorio permite generar datasets de tráfico IoT real, superando la falta de conjuntos de datos específicos para entornos rurales.

El objetivo general es ofrecer un entorno reproducible, de bajo coste y abierto, que facilite la investigación aplicada en seguridad IoT, análisis de tráfico y aprendizaje automático, contribuyendo al desarrollo de soluciones sostenibles y seguras para el ámbito rural.

La estructura del presente trabajo es la siguiente: la Sección II revisa los trabajos relacionados sobre herramientas y bancos de pruebas así como los conjuntos de datos más empleados en el ámbito de la ciberseguridad. La Sección III describe la arquitectura y la metodología del Smart Rural IoT Lab, incluyendo los protocolos de comunicación, los escenarios experimentales y el proceso de generación del conjunto de datos. La Sección IV presenta y analiza los resultados experimentales obtenidos durante la validación del laboratorio. Finalmente, la Sección V resume las principales contribuciones de este trabajo y expone las líneas futuras de investigación.

II. Trabajo Relacionado

En los últimos años se han desarrollado numerosos laboratorios y plataformas experimentales orientados al estudio del IoT y la ciberseguridad. La mayoría de los laboratorios IoT existentes se orientan a entornos urbanos o industriales, donde la conectividad y los recursos computacionales son abundantes. Herramientas de simulación como

Tabla I
Entornos IoT Urbanos vs. Rurales: Diferencias Clave.

Aspecto	Entornos IoT Urbanos	Entornos IoT Rurales
Conectividad	Alto ancho de banda, baja latencia	Intermitente, bajo ancho de banda, alta latencia
Suministro de energía	Estable y continuo	A menudo limitado, alimentado por energía solar o baterías
Infraestructura	Densa, centralizada	Dispersa, descentralizada
Acceso físico	Mantenimiento regular posible	Acceso remoto, infrecuente
Soporte de ciberseguridad	Redes gestionadas, monitorización centralizada	Supervisión limitada, nodos distribuidos y fuera de línea
Factores ambientales	Controlados (interior, urbano)	Impredecibles (terreno, clima, obstáculos)

*Autor de correspondencia.

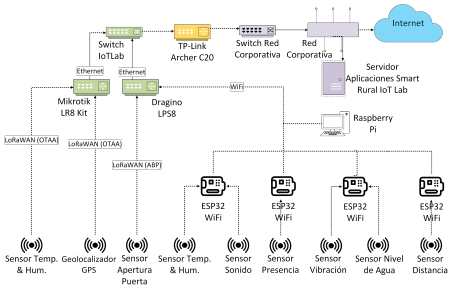


Figura 1. Smart Rural IoT Lab: Topología Física

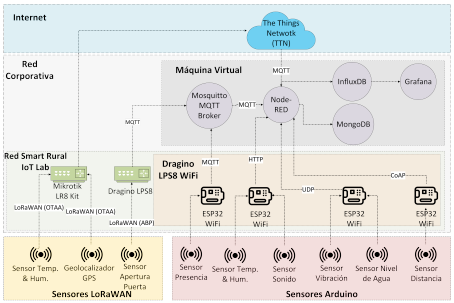


Figura 2. Smart Rural IoT Lab: Topología Lógica

GNS3 [6] y CupCarbon [7] ofrecen gran flexibilidad para diseñar topologías y flujos de tráfico, pero carecen de componentes físicos que reproduzcan la variabilidad de las comunicaciones reales. Estas soluciones, resultan adecuadas para escenarios controlados, pero no reflejan las condiciones de conectividad ni la heterogeneidad de dispositivos que caracterizan a los entornos rurales. En este contexto, han surgido propuestas híbridas como GothX [8] y el Gotham Testbed [9], que combinan nodos virtuales y físicos para generar tráfico tanto benigno como malicioso. Estos bancos de pruebas han sido relevantes en la investigación sobre detección de intrusiones, ya que permiten evaluar el comportamiento de los sistemas frente a distintos ataques. Sin embargo, su arquitectura se orienta a infraestructuras centralizadas y de alta capacidad.

Paralelamente, se han producido avances significativos en la generación de datasets IoT para investigación en seguridad. Datasets como CIC-IDS2017 [10], UNSW-NB15 [11], ToN-IoT [12] y Edge-IIoTset [13] han sido fundamentales para la investigación en detección de intrusiones, pero todos fueron obtenidos bajo redes estables y de alta capacidad. En resumen, la mayoría de estos datasets prioriza la eficiencia energética o el rendimiento, relegando la ciberseguridad a un segundo plano.

El Smart Rural IoT Lab aborda la carencia de herramientas para el desarrollo de soluciones IoT en el ámbito rural. Su función principal es servir como un banco de pruebas dedicado a la investigación científica y a la validación de la seguridad de las soluciones IoT aplicadas en entornos rurales, contribuyendo a cubrir la brecha entre la investigación en IoT urbano y rural.

III. Metodología

El Smart Rural IoT Lab se ha diseñado como un entorno modular y reproducible que combina dispositivos físicos de bajo coste, protocolos de comunicación heterogéneos y herramientas de software abiertas para simular condiciones reales de despliegue rural. Su arquitectura reproduce el ciclo completo de vida de los datos IoT, desde su captura hasta su almacenamiento y visualización.

III-A. Arquitectura física y lógica del laboratorio

La arquitectura física integra dispositivos y pasarelas distribuidos en dos dominios de comunicación:

1. Largo alcance (LoRaWAN): Orientado a la monitorización de variables ambientales a gran distancia con mínimo consumo energético.
2. Corto alcance (WiFi): Empleado en escenarios de comunicación local donde se requiere mayor frecuencia de muestreo o transmisión de datos en tiempo real.

La Figura 1 muestra la topología física: sensores Dragino (LHT65, LGT-92, LDS02) y de bajo coste (DHT11, PIR, etc.), microcontroladores Arduino ESP32 y gateways LoRaWAN (Mikrotik LR8, Dragino LPS8) junto con routers WiFi locales y una Raspberry Pi 3 Model B para la captura de tráfico.

La Figura 2 representa la topología lógica del laboratorio, donde se muestra la interacción entre los dispositivos de adquisición de datos, transporte, almacenamiento y visualización.

III-B. Selección de protocolos IoT

La selección de los protocolos de comunicación se realizó considerando los requisitos característicos del entorno rural: eficiencia energética, tolerancia a latencias altas y compatibilidad con dispositivos de bajo consumo. En este contexto, el laboratorio soporta los siguientes protocolos:

- LoRaWAN: Tecnología LPWAN de bajo consumo y amplio alcance [14]–[16].
- MQTT: Protocolo publish/subscribe ideal para redes multisensor [17]–[19].
- HTTP: Protocolo request/response compatible con servicios web [20], [21].
- UDP: Protocolo sin conexión para comunicaciones rápidas.
- CoAP: Protocolo ligero basado en UDP optimizado para dispositivos con recursos restringidos.

Estos protocolos fueron seleccionados no sólo por su relevancia técnica, sino también por su complementariedad, permitiendo reproducir diferentes escenarios de comunicación rural: desde nodos remotos con baja frecuencia de transmisión (LoRaWAN) hasta redes locales de sensores (WiFi + MQTT/CoAP).

III-C. Backend

El backend combina únicamente software open source:

- Mosquitto MQTT Broker [17]: Encargado de la mensajería IoT.
- Node-RED [22]: Orquestación de flujos.
- InfluxDB [23]: Base de datos de series temporales.
- MongoDB [24]: Almacenamiento basado en documentos.
- Grafana [25]: Visualización en tiempo real.
- Tcpdump y Zeek [26], [27]: captura y procesado de tráfico.

Todos los servicios se despliegan con Docker [28], lo que facilita su portabilidad.

III-D. Escenarios experimentales

Se definieron seis escenarios que cubren casos de uso típicos del ámbito rural (Tabla II).

Cada escenario es procesado por un flujo de Node-RED personalizado para decodificar y almacenar los mensajes, generando tráfico reproducible para análisis de rendimiento y ciberseguridad.

III-E. Captura de tráfico y dataset

La captura de tráfico se centra en las cuatro configuraciones basadas en WiFi (MQTT, HTTP, UDP y CoAP), seleccionadas por su completa observabilidad, sin cifrado ni infraestructura de terceros. Para ello, se utiliza una Raspberry Pi 3 Model B como nodo pasivo de captura de tráfico y puerta de enlace para los microcontroladores Arduino ESP32, desempeñando un papel clave, al permitir la monitorización pasiva, sin interferir con las operaciones normales.

El trafico de red se capturó en formato *PCAP* usando Tcpdump y se procesó con Zeek, una herramienta de análisis de tráfico de red que procesa el tráfico capturado y produce una colección de

archivos de registro estructurados en formato de texto plano como *.log* o *JSON*.

Estos logs incluyen información de protocolos, tiempos de sesión, IPs, puertos y detalles específicos (MQTT topics, URIs HTTP, métodos CoAP, etc.). Esta generación de datasets es una de las contribuciones principales del laboratorio, proporcionando un dataset realista, útil para respaldar una amplia gama de actividades de investigación, incluyendo el desarrollo de modelos impulsados por IA para la detección de anomalías y la ciberseguridad, particularmente en entornos de IoT.

IV. Resultados y Discusión

IV-A. Validación funcional

Cada escenario fue ejecutado de manera independiente para comprobar la correcta interacción entre sensores, gateways y servicios del backend.

En los escenarios LoRaWAN + TTN y LoRaWAN ABP, las transmisiones desde sensores Dragino (LHT65, LGT-92 y LDS02) fueron recibidas sin pérdida significativa, demostrando la interoperabilidad entre gateways Mikrotik LR8 y Dragino LPS8. En el primer caso, los datos se publicaron en The Things Network y se procesaron a través de un broker MQTT público. En el segundo, toda la infraestructura operó sin conexión a Internet, validando la autonomía de la red local. En ambos escenarios los flujos Node-RED decodificaron correctamente los paquetes LoRaWAN y almacenaron los valores en InfluxDB y MongoDB, confirmando la robustez del pipeline.

En los escenarios basados en WiFi (MQTT, HTTP, UDP y CoAP) se verificó el funcionamiento de las comunicaciones de corto alcance. Los microcontroladores ESP32 establecieron conexión con el punto de acceso Dragino LPS8 y transmitieron lecturas periódicas. Los flujos Node-RED filtraron, limpiaron y almacenaron los datos, actualizando en tiempo real los paneles Grafana.

El uso combinado de InfluxDB y MongoDB garantizó persistencia y flexibilidad. La plataforma almacenó 97.914 lecturas válidas, confirmando la estabilidad del sistema incluso con variabilidad temporal.

Durante la ejecución del escenario CoAP, se observaron ligeras limitaciones en las bibliotecas Arduino disponibles. Este hallazgo resalta la necesidad de revisar las librerías de protocolos ligeros cuando se aplican a plataformas embebidas.

En conjunto, los seis escenarios demostraron que es posible emular con bajo coste las principales topologías de comunicación rural, tanto con infraestructura pública como con despliegues privados, y generar tráfico reproducible para análisis posteriores.

IV-B. Dataset

Como se comentó anteriormente, el tráfico de red se capturó mediante Tcpdump en la Raspberry Pi 3 Model B y posteriormente se procesó con Zeek para extraer registros en formato .log. Se realizaron

Tabla II
Smart Rural IoT Lab: Escenarios.

Escenario	Protocolo	Aplicación Práctica
LoRaWAN + TTN	LoRaWAN + MQTT	Geolocalización y condiciones ambientales (infraestructura pública)
LoRaWAN + ABP	LoRaWAN + MQTT	Monitorización local del acceso a instalaciones sin dependencia de la nube
WiFi + MQTT	MQTT	Detección de movimiento en alojamientos de ganado o salas de maquinaria
WiFi + HTTP	HTTP	Monitorización ambiental en invernaderos o almacenes
WiFi + UDP	UDP	Monitorización de vibración/nivel de agua en tanques o silos
WiFi + CoAP	CoAP	Detección de distancia en estaciones de ordeño o instalaciones rurales

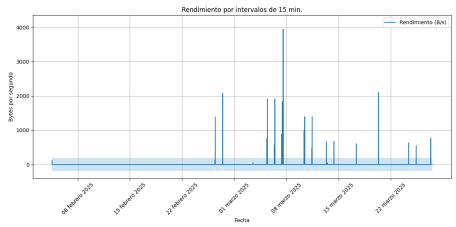


Figura 3. Smart Rural IoT Lab: Rendimiento (15min.)

dieciséis sesiones experimentales con duraciones entre 32 minutos y 5 horas, sumando más de 160.000 paquetes y un total de 33 MB de datos. Zeek generó 103 archivos de log que incluyen información detallada sobre protocolos, duración de sesiones, direcciones IP, puertos, tamaños de payload y campos específicos de capa de aplicación (topics MQTT, URIs HTTP, métodos CoAP, etc.).

El dataset final contiene 2.326 flujos de red con una duración media de 354 segundos. Aproximadamente el 76 % de los flujos fueron no reconocidos (S0-no acknowledged), comportamiento esperado en protocolos sin confirmación como UDP y CoAP.

La Figura 3 muestra el rendimiento (throughput) agregado (bytes/segundo) calculado en intervalos de 15 minutos durante todas las sesiones. Las oscilaciones reflejan las transmisiones de sensores, alternados con periodos de inactividad, lo que reproduce fielmente la naturaleza intermitente de las comunicaciones IoT.

La estructura final del dataset mantiene un directorio por sesión con:

- Archivos .pcap originales (captura bruta).
- Logs procesados por Zeek (conn.log, dns.log, http.log, etc.).
- Documentación README con las condiciones experimentales.

Este enfoque proporciona trazabilidad y reproducibilidad, permitiendo comparar la actividad de diferentes protocolos o escenarios en el tiempo.

IV-C. Indicadores de seguridad pasiva

Aunque el tráfico capturado corresponde exclusivamente a escenarios benignos, el análisis efectuado con Zeek permitió identificar múltiples indicadores pasivos de seguridad. Los registros *weird.log* mostraron diversas inconsistencias de protocolo y cierres anómalos de sesión, habituales en redes IoT debido a la naturaleza no persistente de las conexiones o a la pérdida de paquetes en entornos inalámbricos. Los tres tipos más frecuentes (possible split routing, data before established e inappropriate FIN) presentaron 132 ocurrencias cada uno. Si bien estos comportamientos no implican necesariamente actividad maliciosa, pueden servir como indicadores de comportamiento base frente a futuras inyecciones de tráfico adversario.

Además, los registros DNS se analizaron mediante el cálculo de entropía de Shannon en las cadenas de consulta, con el fin de detectar posibles dominios generados algorítmicamente (DGAs). La entropía media fue de 3.42 bits/carácter (mínimo 1.5, máximo 4.18), valores coherentes con tráfico legítimo de dispositivos IoT que emplean dominios estructurados o estáticos. Este tipo de métricas podrán servir en futuras fases para identificar tráfico anómalo o detectar malware beacons en despliegues reales.

En paralelo, los registros ssh.log permitieron extraer información detallada sobre los algoritmos criptográficos empleados, mecanismos de intercambio de claves y MAC utilizados. Estos metadatos, junto con los registros TLS/SSL y HTTP, aportan una visión complementaria del estado de las comunicaciones dentro del entorno experimental.

En conjunto, estos resultados demuestran la capacidad del Smart Rural IoT Lab para generar y analizar tráfico real de forma granular, permitiendo extraer indicadores pasivos de ciberseguridad sin necesidad de alterar el comportamiento de los dispositivos. Esta información constituye una base sólida para el entrenamiento de

modelos de detección de intrusiones basados en aprendizaje profundo y para el desarrollo de mecanismos de monitorización adaptativa en redes IoT rurales.

IV-D. Discusión general

Los resultados confirman que el Smart Rural IoT Lab constituye una plataforma estable y versátil para el estudio de la comunicación y la seguridad en entornos rurales.

Entre los hallazgos más relevantes destacan:

- Viabilidad técnica: los seis escenarios operaron de forma autónoma sin pérdidas significativas, demostrando que la infraestructura puede funcionar completamente sin servicios cloud.
- Diversidad de protocolos: la coexistencia de LoRaWAN, WiFi y protocolos ligeros permite reproducir distintos patrones de tráfico y evaluar su comportamiento bajo condiciones heterogéneas.
- Generación de datasets realistas: el tráfico capturado refleja condiciones propias del IoT rural, superando las limitaciones de los conjuntos de datos simulados disponibles.

Desde la perspectiva de ciberseguridad, el laboratorio proporciona un marco idóneo para estudiar la correlación entre eventos de red y comportamientos anómalos sin comprometer infraestructuras reales. Su modularidad facilita el análisis de ataques simulados, como spoofing, replay o DoS, que serán integrados en futuras fases experimentales y servirán para entrenar algoritmos de detección de intrusiones basados en inteligencia artificial..

En síntesis, la plataforma demuestra que es posible reproducir condiciones rurales complejas con recursos limitados, ofreciendo una herramienta replicable para universidades, centros de investigación y administraciones locales.

V. Conclusiones y Trabajo Futuro

Este trabajo presentó una versión condensada de los resultados del artículo original *"IoT Platform for Assessment and Research on Cybersecurity of Smart Rural Environments" [Future Internet, 2025] [1]*. Smart Rural IoT Lab constituye una infraestructura robusta, escalable y reproducible que permite analizar comunicaciones IoT en condiciones reales de despliegue rural, abarcando desde la capa física hasta la capa de aplicación. La validación de seis escenarios funcionales demuestra que es posible reproducir, con equipamiento asequible y software libre, todo el ciclo de vida de los datos IoT (captura, procesamiento, almacenamiento y análisis) sin recurrir a servicios externos en la nube.

Una de las principales aportaciones del trabajo es la integración práctica de tecnologías heterogéneas dentro de un mismo entorno experimental. La coexistencia de LoRaWAN y WiFi permite evaluar simultáneamente la eficiencia de protocolos de largo alcance y bajo consumo con soluciones de alta capacidad en redes locales.

Otra contribución relevante es la generación de un dataset genuino de tráfico IoT rural. La captura mediante Tcpdump y Zeek ofrece trazas reales de protocolos IoT, incluyendo metadatos de aplicación (topics MQTT, métodos CoAP, URIs HTTP) y registros de seguridad (weird.log, dns.log, ssl.log).

Desde el punto de vista científico, el Smart Rural IoT Lab amplía el ámbito de la investigación en ciberseguridad IoT hacia dominios rurales, tradicionalmente poco explorados. Por otro lado, la infraestructura demuestra el potencial de las arquitecturas autosuficientes y descentralizadas.

Como líneas de trabajo futuro, se prevé:

- La inyección controlada de tráfico maligno (spoofing, replay, DoS) etiquetando cada evento para construir datasets híbridos benigno/malicioso.
- El entorno se ampliará para ejecutar modelos ligeros de detección de intrusiones en dispositivos IoT, aplicando técnicas de inteligencia artificial para detección y respuesta temprana a incidentes.
- Implementar instrumentación para medir el consumo energético por nodo y protocolo, evaluando el equilibrio entre seguridad, frecuencia de transmisión y duración de la batería. Esto permitirá optimizar configuraciones para despliegues prolongados en campo.
- Se prevé incorporar nuevos sensores y módulos de comunicación adicionales para estudiar la interoperabilidad multiprotocolo.
- El laboratorio se plantea como recurso compartido. Se prevé publicar el dataset bajo licencia abierta, fomentando la replicación y comparación entre grupos de investigación.

En conjunto, el Smart Rural IoT Lab contribuye a reducir la brecha entre la investigación del IoT urbano/industrial y el IoT rural, ofreciendo una base sólida para el desarrollo de soluciones IoT seguras, sostenibles y orientadas al territorio.

Agradecimientos

Los autores desean agradecer el apoyo del Gobierno de España (a través del Instituto Nacional de Ciberseguridad de España, INCIBE), con el Proyecto de Investigación Estratégico "Análisis de aplicaciones móviles desde la perspectiva de la protección de datos: Ciberprotección y Ciberriesgos de la información ciudadana", así como la Cátedra Internacional "Smart Rural IoT y Entornos Seguros", en el contexto del Plan de Recuperación, Transformación y Resiliencia financiado por la Unión Europea (NextGenerationEU/PRTR).

Referencias

[1] D. Sernández-Iglesias, L. Tobarra, R. Pastor-Vargas, A. Robles-Gómez, P. Vidal-Balboa, and J. Sarraipa, "Internet of things platform for assessment and research on cybersecurity of smart rural environments," *Future Internet*, vol. 17, 8 2025.

[2] N. Cvar, J. Trilar, A. Kos, M. Volk, and E. S. Duh, "The use of iot technology in smart cities and smart villages: Similarities, differences, and future prospects," *Sensors (Basel, Switzerland)*, vol. 20, 2020.

[3] S. A. Alabdali, S. F. Pileggi, and D. Cetindamar, "Influential factors, enablers, and barriers to adopting smart technology in rural regions: A literature review," *Sustainability*, 2023.

[4] M. Gupta, M. Abdelsalam, S. Khorsandroo, and S. Mittal, "Security and privacy in smart farming: Challenges and opportunities," *IEEE Access*, vol. 8, pp. 34 564–34 584, 2020.

[5] S. Sontowski, M. Gupta, S. S. L. Chukkapalli, M. Abdelsalam, S. Mittal, A. Joshi, and R. Sandhu, "Cyber attacks on smart farming infrastructure," *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*, pp. 135–143, 2020.

[6] GNS3, "GNS3," https://docs.gns3.com/docs/, 2025, Último acceso: 12 de enero de 2026.

[7] A. Bounceur, O. Marc, M. Lounis, J. Soler, L. Clavier, P. Combeau, R. Vauzelle, L. Lagadec, R. Euler, M. Bezoui, and P. Manzoni, "CupCarbon-Lab: An IoT Emulator," in *2018 15th IEEE Annual Consumer Communications & Networking Conference : 12-15 January 2018, Las Vegas, NV, USA*. Institute of Electrical and Electronics Engineers, 2018, p. 172.

[8] M. Poisson, R. Carnier, and K. Fukuda, "GothX: A generator of customizable, legitimate and malicious IoT network traffic," in *ACM International Conference Proceeding Series*. Association for Computing Machinery, 8 2024, pp. 65–73.

[9] X. Saez-De-Camara, J. L. Flores, C. Arellano, A. Urbieta, and U. Zurutuza, "Gotham Testbed: A Reproducible IoT Testbed for Security Experiments and Dataset Generation," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, pp. 186–203, 1 2024.

[10] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *ICISSP 2018 - Proceedings of the 4th International Conference on Information Systems Security and Privacy*, vol. 2018-January. SciTePress, 2018, pp. 108–116.

[11] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data set for Network Intrusion Detection systems (UNSW-NB15 Network Data Set)," Tech. Rep., 2015.

[12] I. Tareq, B. M. Elbagoury, S. El-Regaily, and E. S. M. El-Horbaty, "Analysis of ToN-IoT, UNW-NB15, and Edge-IIoT Datasets Using DL in Cybersecurity for IoT," *Applied Sciences (Switzerland)*, vol. 12, 10 2022.

[13] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning," Tech. Rep., 2022.
[14] L. Alliance, "What is LoRaWAN®?" https://lora-alliance.org/about-lorawan/, 2025, Último acceso: 12 de enero de 2026.
[15] M. Jouhari, N. Saeed, M. S. Alouini, and E. M. Amhoud, "A Survey on Scalable LoRaWAN for Massive IoT: Recent Advances, Potentials, and Challenges," *IEEE Communications Surveys and Tutorials*, vol. 25, pp. 1841–1876, 2023.
[16] T. T. Network, "LoRaWAN Device Classes," https://www.thethingsnetwork.org/docs/lorawan/classes/, 2025, Último acceso: 12 de enero de 2026.
[17] E. Foundation, "Mosquitto MQTT Broker," https://mosquitto.org/, 2025, Último acceso: 12 de enero de 2026.
[18] T. Prokosch, "MQTT Protocol Guide: Everything You Need to Know," https://cedalo.com/blog/complete-mqtt-protocol-guide/, 2023, Último acceso: 12 de enero de 2026.
[19] E. Atilgan, I. Ozcelik, and E. N. Yolacan, "MQTT Security at a Glance," in *14th International Conference on Information Security and Cryptology, ISCTURKEY 2021 - Proceedings*. Institute of Electrical and Electronics Engineers Inc., 2021, pp. 138–142.
[20] C. B. Gemirter, Şagatay Çenturca, and Şebnem Baydere, "A Comparative Evaluation of AMQP, MQTT and HTTP Protocols Using Real-Time Public Smart City Data," in *Proceedings - 6th International Conference on Computer Science and Engineering, UBMK 2021*. Institute of Electrical and Electronics Engineers Inc., 2021, pp. 542–547.
[21] V. Upadhyay, D. A. Vaish, and D. J. Kokila, "Internet of Things(IoT) security: Approaches and existing solutions," Tech. Rep., 2023.
[22] O. Foundation and N.-R. Contributors, "Node-RED," https://nodered.org, 2025, Último acceso: 12 de enero de 2026.
[23] I. Inc., "InfluxDB," https://www.influxdata.com/, 2025, Último acceso: 12 de enero de 2026.
[24] M. Inc., "MongoDB," https://www.mongodb.com/, 2025, Último acceso: 12 de enero de 2026.
[25] G. Labs, "Grafana," https://grafana.com/, 2025, Último acceso: 12 de enero de 2026.
[26] T. Group, "Tcpdump," https://www.tcpdump.org/, 2025, Último acceso: 12 de enero de 2026.
[27] "Zeek," https://www.zeek.org, 2025, Último acceso: 12 de enero de 2026.
[28] D. Inc., "Docker," https://www.docker.com/, 2025, Último acceso: 12 de enero de 2026.

About Formal Security Models for the Post-Quantum ML-KEM

Inés Araujo Cañas
Department of Computer Science
EPFL, Lausanne, Switzerland
ines.araujocanas@epfl.ch

Luis Hernández Encinas
Institute of Physical and Information Technologies (ITEFI)
Spanish National Research Council (CSIC), Madrid, Spain
luis.h.encinas@csic.es

***Abstract*—This paper presents a study of the state of the art in machine-checked security proofs for post-quantum cryptography, with a focus on the formal verification of lattice-based schemes. Building on this analysis, we include a machine-checked security proof for the core public-key encryption primitive underlying the CRYSTALS-Kyber and Module-Lattice-Based Key Encapsulation Mechanism (ML-KEM) post-quantum cryptosystems. We use the EASYCRYPT proof assistant to formalize the Indistinguishability under Chosen-Plaintext Attack (IND-CPA) security of the Kyber/ML-KEM core key encapsulation and demonstrate reduction to the Module Learning With Errors (MLWE) problem, the foundational security assumption for these schemes. Our framework details each step in this reduction proof, explaining the chain of game-based transformations from real-world encryption security games down to the underlying hardness problem.**

***Index Terms*—EasyCrypt, IND-CPA security, Kyber, machine-checked security reduction, ML-KEM, MLWE**

I. INTRODUCTION

Current cryptography is facing a major turning point. The rapid development of quantum algorithms –with Shor's algorithm able to break widely used asymmetric encryption systems and Grover's algorithm able to weaken others– has brought new urgency to the field. Unlike older threats, attackers might now store large amounts of encrypted data and wait until quantum computers are available to break the encryption later on. This "store-now, decrypt-later" scenario means we need to update our cryptosystems quickly, well before quantum computers become cryptographically relevant.

To address these risks, the U.S. National Institute of Standards and Technology (NIST) began a public call to choose new cryptographic standards that are resistant to quantum attacks. These efforts have led to the selection of new algorithms, including ML-KEM, ML-DSA, SLH-DSA, FALCON, and HQC. This marks a fundamental shift away from older mathematical assumptions (like integer factoring or discrete logarithms) toward newer problems believed to be hard even for quantum computers, such as Module Learning With Errors (MLWE). However, this new cryptography, named Post-Quantum Cryptography (PQC), is less tested, with fewer mature and widely reviewed implementations, and so we must be vigilant in verifying both the theoretical security, as well as the implementation security of these algorithms.

This work examines the formal security models behind ML-KEM and discusses the latest verification tools and techniques. We highlight both the progress and the current limitations as we work toward reliable asymmetric PQC –always keeping in mind that real adversaries may have more capabilities than our models assume, especially in the quantum era.

The rest of the paper is organized as follows. In §II we present some of the background terms used in this work and in §III we include the state of the art about this subject. In §IV, we analyse the main formal security proofs which has been verified. Finally, some conclusions and possible future works are presented in §V.

II. BACKGROUND

A. The CRYSTALS-Kyber Cryptoscheme

The CRYSTALS-Kyber (or simply Kyber) key encapsulation mechanism (KEM) [1], for generating and securely transmitting a shared secret key, was submitted to NIST's post-quantum standardization effort in November 2017. Later, Kyber was published as the ML-KEM standard [2] by NIST in 2024, along with ML-DSA [3] and SLH-DSA [4].

1) Module Learning With Errors Security Game: The security of Kyber's encryption and encapsulation is based on the MLWE problem, a generalization of the Learning With Errors (LWE) [5] problem, which is assumed to be hard even for quantum computers.

The MLWE problem can be informally described as solving noisy linear equations over the quotient ring $R_q := R/qR$ (where R is a ring and q is a positive integer). These equations are constructed using a matrix A, a secret vector s, and an error vector e, sampled, respectively, from distributions D_{m}, D_{s}, and D_{e}, over R_q. A sample is obtained by computing $t = A \cdot s + e \pmod q$ and outputting the pair (A, t), or (sd, t), where sd is a seed from which we can derive matrix A. The decision variant of the MLWE problem (which is relevant for the cryptographic constructions) asks whether one can distinguish such an MLWE sample from a sample where A and t are chosen uniformly at random from the pairs of matrices and vectors over R_q.

The advantage of an adversary quantifies how well it can distinguish a valid MLWE sample from a uniformly random one. Formally, the MLWE challenger either samples a pair $(A, t = A \cdot s + e)$ –a structured sample– or a pair (A, t) where t is drawn uniformly at random. The adversary $\mathcal{A}$ outputs a bit b' guessing which distribution the pair came from. The advantage is defined as

$$\mathrm{Adv}^{\mathrm{MLWE}}(\mathcal{A}) = |\Pr[\mathcal{A}(\mathrm{A}, \mathrm{A} \cdot \mathrm{s} + \mathrm{e}) = 1] - \Pr[\mathcal{A}(\mathrm{A}, \mathrm{t}) = 1]| \,.$$

This game constitutes the base of the security guarantees of Kyber. If no efficient algorithm (classical or quantum)

exists that can solve the game, the underlying lattice system is considered secure.

2) The Chosen-Plaintext Attack Security Model: The *indistinguishability under chosen plaintext attack* (IND-CPA) security model constitutes a security game where an adversary can first see the encryptions of a number of plaintexts of their choice. Then, they choose two new plaintexts and receive their encryptions. The challenger flips a secret bit b, encrypts one of the two messages m_0, m_1 as $c = \text{Enc}(m_b)$, and gives c to the adversary. The adversary outputs a bit b', their guess on which encryption belongs to which plaintext. An encryption scheme is indistinguishable under chosen-plaintext attack if no adversary can distinguish between the encryptions of two plaintexts of their choice, other than by guessing.

The adversary's advantage, meaning how well an adversary can distinguish encryptions of two chosen plaintexts in the case of this game, is computed as

$$\text{Adv}_{\text{IND-CPA}}(\mathcal{A}) = \left|\Pr[b' = b] - \tfrac{1}{2}\right|.$$

This value quantifies how much better the adversary performs than random guessing.

3) Kyber's Base Public-Key Encryption Algorithms: Kyber's KEM is built from a standard IND-CPA lattice-based public-key encryption (PKE) scheme, called K-PKE. It has the following (simplified) core algorithms:

- Key Generation: compute matrix $A \in R_q^{k\times k}$, secret vector $s \leftarrow \chi_{\eta_1}^k$, and error vector $e \leftarrow \chi_{\eta_1}^k$. The public key is $pk = (sd, t)$, with $t = A \cdot s + e \pmod q$ and sd a seed from which we can derive A. The secret key is the secret vector $sk = s$, where $R_q = \mathbb{Z}_q[X]/(X^{256}+1)$, $q = 3329$, χ_η is the centered binomial distribution CBD_η (coefficients bounded $|e_i| \le \eta$), and parameters from [2, Table 2], i.e., ML-KEM-512 has $k = 2$, $\eta_1 = 3$, $\eta_2 = 2$.
- Encryption: Given the public key pk and message m, recover A from the public key. Then, sample ephemeral secret vector $y \leftarrow \chi_{\eta_1}^k$ and error vectors $e_1 \leftarrow \chi_{\eta_1}^k$, $e_2 \leftarrow \chi_{\eta_2}$. Output the ciphertext (u, v), where $u = A^\top \cdot y + e_1 \pmod q$ and $v = t^\top \cdot y + e_2 + \mu \pmod q$, where μ is an appropriate encoding of message m, where $\eta_2 = 2$ for ML-KEM-512/768/1024 except $\eta_1 = 2$ for 768/1024; aggregate decryption error $\|\tilde{n}\|_\infty < \lfloor q/4 \rfloor = 832$ with high probability $< 2^{-138}$ [2, Table 1].
- Decryption: Compute $v' = v - s^\top \cdot u \pmod q$. Then, recover the message m from v' by decoding the result to the nearest value.

The Encode, Decode, Compress, and Decompress functions affect the format, optimization, and security of each of the algorithms considered. These functions are defined as encoding or compression functions.

The KEM then wraps this PKE and uses a variant of the Fujisaki-Okamoto (FO) transform [6] for additional security guarantees.

B. Security Reduction

One way to validate the security of cryptographic schemes is to search for attacks: if no attack is found against a scheme, it would be declared "secure". However, this approach is fundamentally limited because the absence of discovered attacks does not prove that no attack exists.

There exists a more rigorous approach for ensuring the security of a cryptographic scheme, known as provable security. Its goal is to produce formal mathematical arguments demonstrating that the security of a cryptosystem relies on the hardness of a well-established computational problem (e.g., the MLWE problem in the case of Kyber). These proofs require us to model an adversary with specific capabilities and limitations, as well as a specific security game to formalize the way the adversary interacts with the cryptosystem. There are several common frameworks to do this, however, this work will focus only on game-based proofs.

1) Game-based proofs: The way in which an attacker interacts with the challenger is defined by a precise mathematical description, called a game. Game-base proofs are composed of sequences of games. The first game in the sequence represents the security goal (e.g. IND-CPA). We can prove other games representing security assumptions preserve or alter only slightly the overall security of this first game, represented as the adversary advantage. The end goal is to reduce the chosen security game to the chosen problem (e.g. MLWE).

Once we have proven these reductions, if we trust the underlying primitives (such as hash functions) of a cryptosystem to be secure, and the computational problems to be hard (meaning there currently is no efficient, polynomial-time algorithm to solve them), then we can conclude that there are no feasible attacks against the cryptographic scheme in question.

2) Oracle Models: In the Random Oracle Model (ROM) [7], a primitive H (typically a hash function) is modeled as a public black-box oracle, meaning that, on each new input x, outputs a value $H(x)$ chosen uniformly at random from the output domain. The oracle is required to respond consistently such that repeated queries to the same input x always yield the same output.

The Quantum Random Oracle Model (QROM) generalizes the ROM to quantum settings by allowing the adversary to query the oracle on quantum superpositions [7], [8]. In particular, we say the random oracle (RO) H is "quantum-accessible", meaning adversaries may submit a quantum superposition of classical queries and receive the corresponding superposition of outputs.

3) Hoare Logic and Relational Hoare Logic: Hoare logic is a formal system with a set of logical rules for reasoning rigorously about the correctness of computer programs. It was proposed in [9] (see also [10], [11]). The main feature of Hoare logic is the Hoare triple, where a triple describes how the execution of a piece of code changes the state of the computation, and it has the form $c\colon \Phi \Rightarrow \Psi$, where Φ and Ψ are assertions and c is a command. Every execution of a statement c starts with an initial state satisfying Φ (the relational pre-condition) and ends with a final state that satisfies Ψ (the relational post-condition). Hoare logic provides a proof system for reasoning about Hoare triples.

The relational properties play an important role in security proofs and they relate pairs of execution, which can be executions of the same program in different inputs, or different programs on the same, or different inputs. We can express this as Hoare quadruples of the form $c_1 \sim c_2\colon \Phi \Rightarrow \Psi$, which means that every pair of executions of statements c_1 and c_2

began with initial states satisfying the relational pre-condition Φ, and ended with final states satisfying the relational post-condition Ψ.

Two programs are equivalent iff they yield the same output when they are evaluated on the same input. Program equivalence is expressed as $c_1 \sim c_2 \colon \approx \Rightarrow \approx$, where $\approx$ denotes the states of variables from c_1 are equal to those of c_2. We can also write $c_1 \sim c_2 \colon \approx_{\mathcal{L}} \Rightarrow \approx_{\mathcal{L}}$ to denote non-interference, where $\approx_{\mathcal{L}}$ refers to pairs of states that coincide on their public variables.

C. Machine Checked Proofs

In [12] two main formal verification frameworks are examined: EASYCRYPT and Jasmin. We briefly describe each below.

1) EASYCRYPT*:* is an interactive framework for verifying the security of cryptographic constructions in the computational model. It uses the game-playing technique introduced in §II-B, reducing the security of a game that represents the security goal to the security of another game representing security assumptions. EASYCRYPT allows us to write code-based game-playing proofs. An EASYCRYPT extension specifically designed to formalize and verify post-quantum security proofs is EASYPQC [12].

The general (loose) proof sketch for proving security of a cryptographic scheme in EASYCRYPT is as follows:

1) *Defining the ROM*: The way the basic formalization of the ROM/QROM is encoded might depend on the specifics of the proof. For example, in [12] the QROM for modeling a hash function, that can be called in superposition, accepts values with the `message` input type, and `hash` output type.
2) *Cryptographic scheme*: Define an implementation for the cryptographic scheme in question. Note that, as long as the module requires the use of a hash function, it will need to be parameterized in the implementation of said hash function. In practice, the type of hash function will be the same as the QROM module described above.
   A quantum module type that describes the adversary of the game must also be defined and parameterized by all the oracles the adversary can query during the game. Lastly, the proof will depart from a module defining the security game for which we want to prove security of the cryptographic scheme in question.
3) *Security games*: The security proof is structured as a sequence of games, starting from the game defined in the previous step. The specific games and transformations will depend on the specifics of the proof. The last game in the sequence should represent the assumption so that the security of the scheme, established by the first security game, be reduced to the security of the last assumption in the sequence.

Refer to [13] for background and context, [13] and [14] to find examples of classical security proofs, and [12] for quantum ones.

2) Jasmin: is a framework for developing high-speed and high-assurance cryptographic software [15]. Its two main components are the Jasmin programming language and its associated compiler. Jasmin has a syntax similar to C and provides one-to-one instruction mapping to assembly, allowing for predictable and tightly controlled code. It is designed for enhancing portability of programs and for simplifying verification tasks. On the other hand, the compiler is designed to ensure predictability of the compiled code such that properties of programs written in Jasmin provably carry to the assembly code.

The set of Jasmin verification tools includes automated tools to prove the functional correctness, memory safety, and constant-time security of cryptographic implementations. The way in which verifying these implementations in Jasmin works is the following. Firstly, the developers write cryptographic algorithms in the Jasmin language. Then, the Jasmin compiler translates the source code into the architecture (normally a x64 assembly) performing optimizations while adhering to formal proofs. Finally, the integrated verification tools try to prove that the final assembly code is functionally correct and resistant to different types of attacks.

III. STATE OF THE ART

In this section we will present the results obtained in [16], which is, in our opinion, the main paper related to the use of formal verification models for Kyber and ML-KEM.

A. The Kyber Formally Verified Implementation and Security Proof

The Kyber KEM is the first post-quantum proposal to feature a formally verified implementation, introduced in [16]. Core codebases are written in Jasmin. An equivalent specification is written in EASYCRYPT, along with equivalence proofs relating it to the Jasmin implementation and IND-CCA security proofs under the classical model.

Verification proceeds in two primary stages:

1) Functional Correctness.
   The reference Jasmin implementation is proven to correctly implement the specification of Kyber described in the EASYCRYPT specification. On the other hand, optimized Jasmin implementation tailored for AVX2 processors is proven to be equivalent to the Jasmin reference implementation. This ensures low-level optimizations do not introduce errors.
2) Security Proofs.
   EASYCRYPT specification faithfully models Kyber version 3.02l [17]. The several steps taken to create this specification are detailed in [16, §2], starting from basic type definitions to reach the final, complete specification. The proof structure and the strategies taken to prove functional correctness and security of the implementation can be found in [16, §5]. Higher-level IND-CPA security and equivalence proofs under the classical model are based on low-level auxiliary lemmas proved at the finite-field, ring, and vector operations level.

The main security and correctness guarantees achieved by these proofs are:

- The Jasmin reference and AVX2-optimized implementations are functionally equivalent.
- The Jasmin reference and EASYCRYPT implementations are functionally equivalent.

- The EASYCRYPT implementation is IND-CCA secure in the classical model, and this security extends (via the equivalence proofs) to both Jasmin implementations.

In addition, security properties established in the Jasmin code are preserved after compilation to assembly, based on the guarantees provided by the Jasmin compiler.

Moreover, benchmarks of the formally verified implementations show the code achieves performance close to existing hand-optimized implementations in C and assembly [18].

Previous work to adapt the EASYCRYPT tool was necessary, as proof techniques commonly used for classic cryptography are not applicable in the quantum setting, such as rewinding [12].

B. The ML-KEM Formally Verified Implementation and Security Proof

In [18], the authors introduce a fully machine-checked, computer-verified proof that establishes the security of the ML-KEM against classical adversaries. These results were achieved before ML-KEM's formal standardization [2], and closely parallel prior efforts for formal verification of the Kyber scheme presented in [16].

The proof demonstrates IND-CCA security of ML-KEM, reducing its security to a specific, well-defined variant of the MLWE problem. It is important to note that this reduction and proof are *classical*; they do not guarantee quantum resistance, which is a limitation inherent to the current state of formal methods and tools.

Similarly to what was done with Kyber in [16], computer-verified proofs of IND-CCA security of the ML-KEM standard [2] are provided, reducing the security of ML-KEM to the MLWE [5] assumption. Moreover, equivalence with two efficient, cryptographic constant time Jasmin implementations of the scheme are provided.

The security proof is divided into the following steps.

1) Modeling the Kyber PKE: Formalize the correctness and IND-CPA security of the Kyber public-key encryption scheme as the core primitive.
2) Formalizing the Fujisaki–Okamoto Transform: Create a formalization of the relevant variant of the FO transform in the ROM.
3) Proving ML-KEM IND-CCA Security: Proving that ML-KEM achieves IND-CCA security and correctness as a KEM based on the previous results.
4) Implementation verification: These results are then complemented by two formally verified, constant-time implementations in Jasmin that are functionally equivalent to the specification, and inherit concrete, modular security guarantees down to (a variant of) MLWE.

IV. IND-CPA SECURITY PROOF OF KYBER/ML-KEM PKE

This section presents an outline of the formalization of the IND-CPA security proof for the ML-KEM base PKE scheme in the ROM, using EASYCRYPT. While the IND-CPA security of K-PKE is established in [18], this work formalizes an adapted extract of those proofs in EASYCRYPT as a reference example. The strategy and sequence of game transformations follow those established in [18, §4] and [19, §5, §7], relying on the modules `MLWE_PKE_Basic.ec` and `MLWE_PKE.ec` with the former serving as a foundation for more complex formalizations.

Notably, while `MLWE_PKE_Basic.ec` doesn't make use of an Oracle, and `MLWE_PKE.ec` utilizes a Sampler module as an interface between the adversary and the ROM, our formalization interacts directly with the ROM, making use of the `POracle` interface from the repository associated to [18].

This change and does not alter the semantics of the proof: the distribution of sampled values remains identical, and all oracle queries are handled in a comparable manner. However, it simplifies the game transitions by reducing the number of modules involved and clarifying the dependency between the PKE construction and the ROM.

A. Cryptographic scheme and security game

Our aim is to construct and relate a chain of security games so that at each step, we prove (by a game jump lemma) that the adversary cannot distinguish adjacent games, except with negligible probability under the MLWE assumption:

1) Game_1: The original PKE and its IND-CPA game, where we replaced t, member of the public key, with a random value.
2) Game_2: The challenge ciphertext is replaced with independent uniform randomness.
3) Final Game: An environment in which the adversary's advantage in the IND-CPA game should be exactly zero.

For space reasons, this paper focuses exclusively on the first game (Game_1) in the sequence and its associated lemmas. We will also provide a concise overview of the design and implementation concept for the second and final games and their lemmas.

Listing 1 is an abstracted EASYCRYPT module for K-PKE, following `MLWE_PKE_Basic.ec`.

Listing 1. Cryptographic Scheme

```
module MLWE_PKE_ROM(O: POracle) (*: Scheme *) =
  {
  proc kg() : pkey * skey = {
    var sd, s,e,_A,t;
    sd <$ dseed;
    s  <$ dshort;
    e  <$ dshort;
    _A <@ O.get(sd);
    t  <- _A *^ s + e;
    return ((t, sd),s);
  }

  proc enc(pk : pkey, m : plaintext) :
      ciphertext = {
    var sd,t,r,e1,e2,_A,u,v;
    (t, sd) <- pk;
    r  <$ dshort;
    e1 <$ dshort;
    e2 <$ dshort_R;
    _A <@ O.get(sd);
    u  <- m_transpose _A *^ r + e1;
    v  <- (t `<*>` r) &+ e2 &+ (m_encode m);
    return(u,v);
  }

  proc dec(sk : skey, c : ciphertext) :
      plaintext option = {
    var u,v;
    (u,v) <- c;
    return Some (m_decode (v &- (sk `<*>` u)));
  }
}.
```

B. *Game*$_1$

Listing 2 presents the PKE scheme for the security Game$_1$. In this game, the challenge MLWE sample is generated similarly to the real scheme, except that the target value t is sampled uniformly at random –as in the decisional MLWE variant– instead of being derived from the secret key.

Listing 2. PKE for Security Game$_1$

```
module MLWE_PKE_ROM1(O: POracle) = {
  proc kg() : pkey * skey = {
    var sd,s,_A,t;
    sd <$ dseed;
    s  <$ dshort;
    t  <$ duni;
    _A <@ O.get(sd);
    return ((t, sd),s);
  }

  include MLWE_PKE_ROM(O) [-kg]

}.
```

To prove the lemmas associated with Game$_1$, we construct the reduction module `B1ROM` shown in Listing 3. This reduction transforms an adversary $\mathcal{A}$ that breaks IND-CPA security of the scheme into an algorithm that solves the MLWE problem. Its sole procedure, `guess`, models a CPA game where the public key is provided as input rather than generated internally by the scheme's key generation function.

Listing 3. Game$_1$ Reduction

```
module (B1ROM(A: Adversary) : ROAdv_T) (H :
    POracle) = {
  proc guess(sd : seed, t : vector, uv : vector
      * R) : bool = {
    var pk, m0, m1, c, b, b';
    pk <- (uv.`1, sd);
    (m0, m1) <@ A(H).choose(pk);
    b <$ {0,1};
    c <@ MLWE_PKE_ROM(H).enc(pk, if b then m1
        else m0);
    b' <@ A(H).guess(c);
    return b' = b;
  }
}.
```

Listings 4 and 5 present the two lemmas associated with the first game hop of our proof. We now explain the meaning and implications of these lemmas, both individually and in relation to each other. In EasyCrypt, an expression such as `Pr[M.f() @ &m:res]` denotes the probability that executing procedure `f` of module `M`, starting from memory state `&m`, produces the output **res**, where `&m` represents the initial state of all program variables.

Similarly, a statement of the form `Pr[M1.f() @ &m:res] = Pr[M2.g() @ &m:res]` asserts that, for every possible output **res**, both programs produce **res** with identical probability when started from the same initial state. This equality expresses that the adversary's success probability is the same in both games. The first five lemmas, corresponding to the initial two game hops, follow this format.

The `CPA` module models the IND-CPA game described in §II-A2, while the `MLWE_RO` module –defined in `MLWE.ec` from the artifacts of [18]– implements the decisional variant of the MLWE game. Its behavior depends on a Boolean flag: when set to *false*, the module provides the adversary (here, `B1ROM`) a pair (u, v) with $u = A \cdot s + e$ (the structured public-key element t) and $v = t \cdot s + e$; when *true*, both u and v are sampled uniformly at random. Thus, when the adversary interprets u as the public-key component t, `MLWE_RO` faithfully simulates the decisional MLWE game.

The first lemma then states that executing the real IND-CPA game `CPA(O, MLWE_PKE_ROM, A)` yields the same success probability as the reduction game `MLWE_RO(B1ROM(A), O)` in the structured-sample case.

Listing 4. Game$_1$ Left Lemma

```
lemma hop1_left_ro &m:
  Pr[CPA(O,MLWE_PKE_ROM,A).main() @ &m : res] =
  Pr[MLWE_RO(B1ROM(A),O).main(false,false) @ &m
      : res].
```

Likewise, the second lemma asserts that running `CPA(O, MLWE_PKE_ROM1, A)` succeeds with the same probability as running `MLWE_RO(B1ROM(A), O)` in the case where the MLWE sample is sampled at random.

Listing 5. Game$_1$ Right Lemma

```
lemma hop1_right_ro &m:
  Pr[CPA(O,MLWE_PKE_ROM1,A).main() @ &m : res] =
  Pr[MLWE_RO(B1ROM(A),O).main(false,true) @ &m :
      res].
```

The proof proceeds with a second hop, where the adversary must distinguish a structured ciphertext from a random one. For this, we define a variant of our scheme that uses the same key generation and decryption as `MLWE_PKE_ROM1`, but an encryption algorithm that outputs ciphertexts (u, v) with both components sampled uniformly at random, rather than structured ML-KEM ciphertexts.

The `B2ROM` game simulates an adversary playing the IND-CPA game, differing from `B1ROM` in that the ciphertext is constructed from the input pair (u, v) instead of by calling `MLWE_PKE_ROM2`'s `encrypt` function. Specifically, the ciphertext is $(u, v + m)$, where m is the adversary's chosen plaintext.

Two lemmas relate the CPA games instantiated with `MLWE_PKE_ROM1` and `MLWE_PKE_ROM2` to the decisional MLWE game. For both lemmas, we instantiate the adversary in `MLWE_RO` with the first flag set to `true`, shifting from the previous decisional MLWE variant to one in which the adversary must distinguish a structured ciphertext (u, v), where $u = m_transpose\ A \cdot r + e$ and $v = t \cdot r + e$ and $r = s$ (the secret key), from a random pair. The second flag continues to determine whether the ciphertext is random or well-formed.

The final game hop, `Game2RO`, simulates the IND-CPA game with both public key and ciphertext uniformly random rather than structured. We prove that an adversary playing `CPA(O, MLWE_PKE_ROM2, A).main()` and one playing `Game2RO(A).main()` have identical success probabilities when started with the same memory.

Two additional lemmas conclude the proof. The first one states that, assuming all oracles and adversaries terminate (are lossless), the main procedure of `Game2RO` with adversary $\mathcal{A}$ outputs success with probability exactly $\frac{1}{2}$, yielding zero adversarial advantage. The main lemma then relates all prior results, stating that under these assumptions, the adversary's IND-CPA advantage in the ML-KEM scheme is characterized entirely by its ability to distinguish structured from random public keys and ciphertexts in the MLWE problem. Listing 6 presents the core statement of the main lemma.

Listing 6. Main Lemma

```
Pr[CPA(O,MLWE_PKE_ROM,A).main() @ &m : res] -
    1%r / 2%r =
  Pr[MLWE_RO(B1ROM(A),O).main(false,false) @ &
      m : res] -
     Pr[MLWE_RO(B1ROM(A),O).main(false,true) @
         &m : res] +
  Pr[MLWE_RO(B2ROM(A),O).main(true,false) @ &m
      : res] -
     Pr[MLWE_RO(B2ROM(A),O).main(true,true) @
         &m : res].
```

V. Conclusions and Future Work

Formal security proofs play a crucial role in establishing the trustworthiness of post-quantum cryptographic schemes such as the ML-KEM based on the CRYSTALS-Kyber framework. This paper has reviewed state-of-the-art machine-checked security proofs, focusing on the formalization of IND-CPA security for the base public-key encryption scheme underlying ML-KEM using the EasyCrypt toolset. By detailing the sequence of game-based reductions from real-world security games to the foundational hardness of the MLWE problem, we have demonstrated the rigor and precision achievable through mechanized verification.

There are some important open lines that we pretend to study in the future. The first one is related to the fact that security proof is classical, so it is important to consider how to extend this proof to post-quantum algorithms. In fact, on verification of ML-KEM in the QROM would require a specialised library to be developed in addition to EasyCrypt. Note that the current version of EasyPQC does not allow to prove Kyber's security in the QROM. Currently, the guarantees only apply to the code produced by the Jasmin compiler. In the future, it would be interesting to extend both the security and functional correctness guarantees to other implementations through equivalence checking.

Acknowledgements

The completion of this work owes much to the opportunity provided by I.A.C.'s summer research internship with the Cryptology and Information Security Research Group at ITEFI-CSIC, generously sponsored by the Max Mazin Foundation. I.A.C. is grateful for the support and resources provided during this period, which greatly contributed to the development and realization of this work.

References

[1] J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehle, "CRYSTALS-Kyber: A CCA-secure module-lattice-based KEM," in *Proc. 2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, 2018, pp. 353–367. [Online]. Available: https://doi.org/10.1109/EuroSP.2018.00032

[2] NIST, "Module-lattice-based key-encapsulation mechanism standard," National Institute of Standards and Technology, Tech. Rep. FIPS 203, 2024. [Online]. Available: https://csrc.nist.gov/pubs/fips/203/final

[3] ——, "Module-lattice-based digital signature standard," National Institute of Standards and Technology, Tech. Rep. FIPS 204, 2024. [Online]. Available: https://csrc.nist.gov/pubs/fips/204/final

[4] ——, "Stateless hash-based digital signature standard," National Institute of Standards and Technology, Tech. Rep. FIPS 205, 2024. [Online]. Available: https://csrc.nist.gov/pubs/fips/205/final

[5] O. Regev, "The learning with errors problem," *Proc. Japan Conference on Discrete and Computational Geometry*, 2010. [Online]. Available: https://cims.nyu.edu/~regev/papers/lwesurvey.pdf

[6] D. Hofheinz, K. Hövelmanns, and E. Kiltz, "A modular analysis of the Fujisaki-Okamoto transformation," Cryptology ePrint Archive, Paper 2017/604, 2021. [Online]. Available: https://eprint.iacr.org/2017/604

[7] T. Yamakawa and M. Zhandry, "Classical vs quantum random oracles," Cryptology ePrint Archive, Paper 2020/1270, 2020. [Online]. Available: https://eprint.iacr.org/2020/1270

[8] D. Boneh, Ö. Dagdelen, M. Fischlin, A. Lehmann, C. Schaffner, and M. Zhandry, "Random oracles in a quantum world," in *Proc. Advances in Cryptology – ASIACRYPT 2011, Lecture Notes Comput. Sci.*, D. H. Lee and X. Wang, Eds., vol. 7073. Berlin, Heidelberg: "Springer Berlin Heidelberg, 2011, pp. 41–69. [Online]. Available: https://doi.org/10.1007/978-3-642-25385-0_3

[9] C. A. R. Hoare, "An axiomatic basis for computer programming," *Communications of the ACM*, vol. 12, no. 10, pp. 576–583, 1969. [Online]. Available: https://doi.org/10.1145/363235.363259

[10] N. Benton, "Simple relational correctness proofs for static analyses and program transformations," Microsoft Research, Cambridge, UK, Tech. Rep. MSR-TR-2005-26, 2005. [Online]. Available: https://nickbenton.name/correctnessfull.pdf

[11] G. Barthe, "An introduction to relational program verification," 2020. [Online]. Available: https://software.imdea.org/~gbarthe/__introrelver.pdf

[12] M. Barbosa, G. Barthe, X. Fan, B. Grégoire, S.-H. Hung, J. Katz, P.-Y. Strub, X. Wu, and L. Zhou, "EasyPQC: Verifying post-quantum cryptography," in *Proc. The ACM Conference on Computer and Communications Security (CCS'21)*, 2021, pp. 2564–2586. [Online]. Available: https://dl.acm.org/doi/pdf/10.1145/3460120.3484567

[13] M. Barbosa, G. Barthe, K. Bhargavan, B. Blanchet, C. Cremers, K. Liao, and B. Parno, "SoK: Computer-aided cryptography," Cryptology ePrint Archive, Paper 2019/1393, 2019. [Online]. Available: https://eprint.iacr.org/2019/1393

[14] G. Barthe, B. Grégoire, S. Heraud, and S. Z. Béguelin, "Computer-aided security proofs for the working cryptographer," in *Proc. Advances in Cryptology – CRYPTO 2011, Lecture Notes Comput. Sci.*, P. Rogaway, Ed., vol. 6841. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 71–90. [Online]. Available: https://doi.org/10.1007/978-3-642-22792-9_5

[15] J. Bacelar Almeida, M. Barbosa, G. Barthe, A. Blot, B. Grégoire, V. Laporte, T. Oliveira, H. Pacheco, B. Schmidt, and P.-Y. Strub, "Jasmin: High-assurance and high-speed cryptography," in *Proc. 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 1807–1823. [Online]. Available: https://doi.org/10.1145/3133956.3134078

[16] J. Bacelar Almeida, M. Barbosa, G. Barthe, B. Grégoire, V. Laporte, J.-C. Léchenet, T. Oliveira, H. Pacheco, M. Quaresma, P. Schwabe, A. Séré, and P.-Y. Strub, "Formally verifying kyber. Episode IV: Implementation correctness," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 3, 2023. [Online]. Available: https://doi.org/10.46586/tches.v2023.i3.164-1

[17] R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé, "CRYSTALS-Kyber: Algorithm specifications and supporting documentation (version 3.02)," PQCrypto, NIST Post-Quantum Cryptography Project, Tech. Rep., August 2021. [Online]. Available: https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf

[18] J. Bacelar Almeida, S. Arranz Olmos, M. Barbosa, G. Barthe, F. Dupressoir, B. Grégoire, V. Laporte, J.-C. Léchenet, C. Low, T. Oliveira, H. Pacheco, M. Quaresma, P. Schwabe, and P.-Y. Strub, "Formally verifying Kyber. Episode V: Machine-checked IND-CCA security and correctness of ML-KEM in EasyCrypt," vol. 14921, 2024, pp. 384–421. [Online]. Available: https://doi.org/10.1007/978-3-031-68379-4_12

[19] K. Kreuzer, "Verification of correctness and security properties for CRYSTALS-KYBER," Cryptology ePrint Archive, Paper 2023/087, 2023. [Online]. Available: https://eprint.iacr.org/2023/087

Building Blocks Descentralizados del Pasaporte Digital de Producto para Mejorar la Soberanía de la Cadena de Suministro y las Prácticas de Economía Circular

Santiago Iván de Diego de Diego
FUNDACIÓN TECNALIA RESEARCH & INNOVATION
Astondo Bidea, Edificio 700, 48160, Derio, Spain
santiago.dediego@tecnalia.com

Iván Gutierrez-Aguero
FUNDACIÓN TECNALIA RESEARCH & INNOVATION
Astondo Bidea, Edificio 700, 48160, Derio, Spain
ivan.gutierrez@tecnalia.com

***Resumen*—Este es un resumen extendido de la publicación: [1]. Este artículo de investigación explora la descentralización de los componentes clave del Pasaporte Digital de Producto (PDP) en el contexto de los *Asset Administration Shells* (AAS), con el objetivo de mejorar su gestión y trazabilidad. Se propone un sistema PDP que integra tres elementos críticos: gestión de identidad, trazabilidad y permisos de acceso, diseñado para garantizar una cadena de suministro más resiliente. Se presenta una arquitectura integral que combina estos elementos, asegurando la seguridad y la transparencia en el manejo de los datos del PDP. El estudio también aborda la cuestión de la custodia de los datos del PDP, analizando los roles y responsabilidades de los distintos actores involucrados en el mantenimiento y protección del sistema. Además, se ha experimentado con tecnologías existentes para conceptualizar un sistema PDP descentralizado aplicado a las AAS, demostrando su viabilidad y efectividad en escenarios reales. Este trabajo arroja luz sobre el potencial de los sistemas descentralizados en la gestión de productos digitales, destacando tanto los beneficios como los desafíos asociados con su implementación.**

***Index Terms*—Pasaporte Digital de Producto, Asset Administration Shell, Gestion de Identidades, Descentralización, Soberanía del dato, Integridad de los datos, Sostenibilidad**

I. Introducción

El Pasaporte Digital de Producto (PDP) es un registro digital que recopila y comparte toda la información de un bien a lo largo de su vida útil [2], con el objetivo de apoyar la economía circular (Reutilizar, Reparar, Rehabilitar y Reciclar) y cumplir los Objetivos de Desarrollo Sostenible y las directrices europeas. Los PDP permiten el seguimiento integral de materiales, facilitan el reciclaje y dotan a todos los actores (fabricantes, recicladores, reguladores, consumidores, etc) con datos verificados para decisiones sostenibles. Aunque ya está adoptado en el Reino Unido, la UE y Alemania [3], su implementación se ve limitada por la fragmentación de datos, la falta de interoperabilidad y los problemas de custodia y derechos de acceso.

El artículo propone una arquitectura descentralizada basada en el Asset Administration Shell (AAS) de Industria 4.0, integrada con Identidad Auto-soberana (SSI) mediante Identificadores Descentralizados (DIDs) y credenciales verificables (VCs), y respaldada por una cadena de bloques para garantizar la inmutabilidad y la confianza sin depender de una autoridad central. El diseño modular propuesto para el PDP incluye gestión de identidades, un registro descentralizado y políticas de permisos que permiten la divulgación selectiva de atributos y certificaciones, ofreciendo así un PDP fiable, escalable y seguro que facilita la trazabilidad.

II. Estado del arte

Basado en una revisión bibliográfica realizada el 15 de mayo de 2025 en la base de datos Scopus utilizando el término de búsqueda 'TITLE-ABS-KEY ("Digital Product Passport")', se obtuvieron 198 resultados, de los cuales 158 corresponden al período entre 2023 y 2024. Esto demuestra que el concepto de Pasaporte Digital de Producto (PDP) está generando un interés significativo en el ámbito académico, además del interés ya observado en el sector empresarial y gubernamental [3].

Algunos autores [4] presentan una revisión detallada de la literatura disponible, abordando la sostenibilidad desde una perspectiva de Industria 5.0. En la literatura, la economía circular es el caso de uso principal en los estudios y soluciones que involucran PDP ([5], [2]). Sin embargo, este debate aún carece de trabajos de investigación que presenten una arquitectura completa para habilitar PDP en AAS y de implementaciones reales [6], así como artículos que analicen enfoques descentralizados para la custodia de los datos. Al mismo tiempo, la falta de estándares y regulaciones establecidos puede llevar a una adopción fragmentada, resultando en inconsistencias y limitando los beneficios potenciales de los PDP [7].

En cuanto al nivel de madurez de los PDP para AAS, algunos autores [8] introdujeron el Digital Lifecycle Passport (DLCP) para AAS, definido como una estructura de datos agnóstica alojada en una plataforma en la nube y accesible por todos los actores del ciclo de vida del producto. Por otro lado, [6] proponen una implementación propia basada en un hubs PDP donde se almacenan todos los PDP. De manera similar, [9] proponen un enfoque basado en la nube para un sistema PDP para AAS. Aunque estas definiciones representan avances en la implementación de PDP mediante submodelos de AAS, aún sufren de centralización, lo que las hace propensas a fugas de datos.

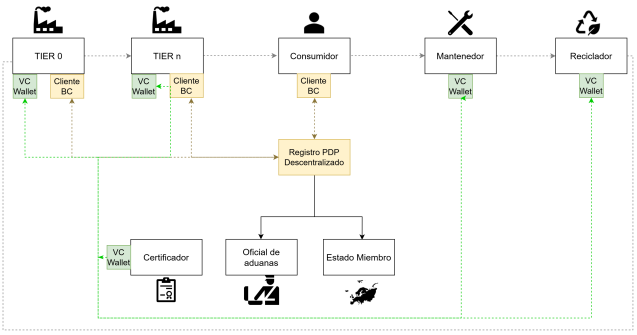


Figura 1. Actores y componentes de nuestro sistema DPP genérico

II-A. *Arquitectura propuesta*

El PDP es una herramienta para seguir los cambios en el ciclo de vida de una AAS, donde la información asociada evoluciona desde la creación hasta el desecho del producto. Esta información puede ser pública o restringida, y algunas deben mostrarse en tiempo real. Es aquí donde la *Self-Sovereign-Identity* (SSI) emerge como una herramienta poderosa para demostrar atributos a un verificador confiable [10]. SSI resulta útil para verificar atributos de manera confiable. Como consecuencia, las tres partes que componen la arquitectura propuesta se describen a continuación:

1. **Identidad descentralizada:** Cada AAS debe tener una identidad digital con atributos únicos y compartidos, asignados por distintas entidades a lo largo de su ciclo de vida. Ejemplos de estos atributos incluyen información ambiental, tasa de recuperación energética o certificados de fabricación, que se pueden almacenar en VCs. Un verificador puede solicitar al AAS una demostración digital de estos atributos, garantizando así que la información proviene de fuentes confiables.
2. **Registro descentralizado:** El PDP puede incluir información que no forma parte de la identidad del AAS, como datos de trazabilidad, almacenados en un registro descentralizado, como una blockchain. Esto permite que las entidades consulten dicha información sin necesidad de conocer al propietario, y con acceso autorizado en cualquier momento. La cadena de bloques y la descentralización garantizan la integridad y disponiblidad de datos críticos, como la ficha técnica, la fecha de registro, el propietario o el historial de mantenimiento.
3. **Política de permisos:** El acceso a la información del AAS debe ser estrictamente gestionado. Esto implica que los permisos deben asignarse a las diferentes partes interesadas según su derecho a escribir información. Por lo tanto, se requiere un mecanismo de control de acceso.

Es necesario definir qué información se almacena en una *wallet* como VCs y qué se registra en un sistema descentralizado, ya que cada opción tiene ventajas y desventajas. Ambos garantizan integridad: los registros descentralizados (como blockchains) lo hacen mediante la enlazabilidad de bloques, y las VCs mediante firmas digitales de emisores confiables. La información en registros descentralizados es más accesible y permite consultas sin involucrar al titular, pero no se puede eliminar ni ocultar parcialmente. En cambio, las VCs permiten el descubrimiento selectivo y la eliminación, y su autenticidad es fácil de verificar. Por ello, las *wallets* son preferibles cuando se requiere privacidad y flexibilidad en la gestión de datos.

La Figura 1 ilustra las partes de un sistema PDP genérico, cuyos actores y roles pueden variar según el caso de uso. El producto comienza en TIER 0 y atraviesa su ciclo de vida hasta TIER n, con la participación de actores como el Consumidor, el Mantenedor y el Reciclador. Los cambios se registran en el PDP descentralizado, con acceso a lectura y escritura restringido según los permisos y el rol de cada actor, mediante un cliente blockchain. Los órganos nacionales e inspectores aduaneros deberán acceso a información específica según normas de la EC [11]. Además, cada actor puede poseer una *wallet* que almacena VCs, emitidas por entidades confiables como Certificadores, que contienen atributos del PDP. Estas VCs pueden usarse para generar Presentaciones Verificables (VP) cuando se soliciten por un verificador.

II-B. *Custodia del PDP*

Aunque el registro descentralizado almacena información accesible públicamente, no es sencillo determinar quién debe custodiar las VCs asociadas a una AAS. Aunque el producto podría tener su propia *wallet*, como se muestra en la Figura 2, este enfoque tiene limitaciones: no todos los productos son capaces de almacenar información de forma estandarizada, no actúan de forma proactiva, ni siempre garantizan seguridad. Además, en caso de robo del PDP, el producto no puede reaccionar, y su fabricante tendría que adaptar el dispositivo para soportar estas funcionalidades. Por ello, el PDP debería ser custodiado por una entidad legal, con un punto central de acceso a todas las VCs asociadas al producto. Además, las VCs deben ser exportables (según la W3C [12]), permitiendo reunir toda la información en una única *wallet* central. Quién asume este rol dependerá del tipo de producto, como por ejemplo el vendedor o el ensamblador en el caso de vehículos.

De acuerdo con las recomendaciones[1] de CEN/CLC/JTC 24 sobre “Digital Product Passport - Framework and System”, existen seis áreas técnicas armonizadas que un sistema PDP debe cubrir (CEN-CENELEC, 2024): i) Almacenamiento de datos, ii) Identificador del portador de datos, iii) Confianza/Seguridad/Soberanía/Acceso, iv) Intercambio de datos, v) Servicios informáticos/APIs, vi) Trabajos y procesamiento de datos. Nuestra propuesta de bloques arquitectónicos para un sistema PDP descentralizado las cubre todas, ya que proporciona un sistema de almacenamiento de datos (i), donde se conserva el identificador del portador de datos (ii), preservando la soberanía de la información y garantizando la confianza mediante el control de acceso (iii), permitiendo el intercambio de datos (iv) y su procesamiento (vi) mediante mecanismos estándar de interoperabilidad como APIs (v).

III. Ciclo de vida del PDP

Esta sección describe el ciclo de vida de un PDP desde su emisión hasta su revocación. Inicialmente, es necesario diferenciar entre la información que se almacena en una credencial verificable (VC), denotada como un conjunto de atributos, y la información que se almacena en el registro

[1] A pesar de la ausencia de estándares formales de JTC 24 en el momento de redactar este documento, las reuniones preliminares muestran interés por cubrir áreas técnicas.

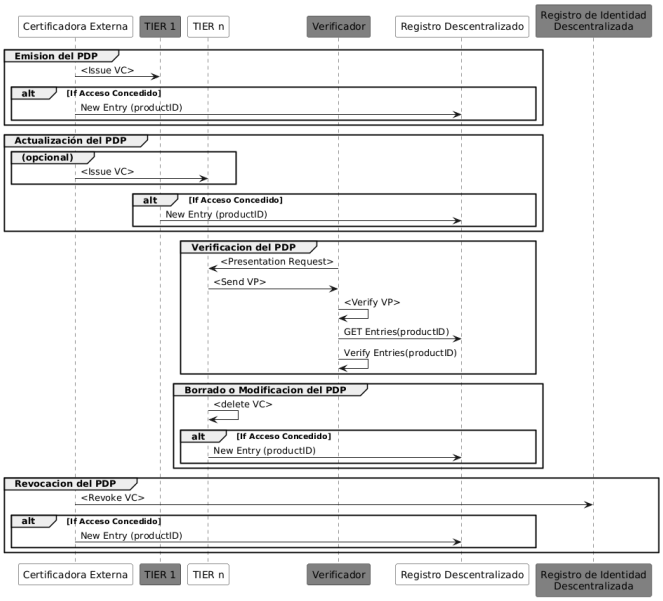


Figura 2. Ciclo de vida del PDP

descentralizado. Este último puede ser de los siguientes tipos: información sobre el producto que puede ser consultada por cualquier persona (pública) o por usuarios autorizados (privada); o registro de cambios en el producto, que se representará como una entrada en el registro con un identificador común del producto, que identifica al producto que se está rastreando. Por lo tanto, un PDP se representará como un conjunto de VCs y entradas en el registro descentralizado. Además, cada nueva pieza de información escrita en el registro descentralizado requiere verificar si el actor tiene los permisos adecuados. La figura 2 representa el ciclo completo del PDP, que se detalla a continuación.

III-A. Emisión del PDP

El ciclo de vida del PDP comienza con su emisión. La emisión del PDP se compone de dos procesos diferentes:

1. Emisión de VC: Se emite una VC al primer actor de la cadena de suministro (TIER 0), quien custodia esta información en su *wallet*. Un ejemplo de esta VC podría ser un certificado del proceso de fabricación. Es importante señalar que estas VCs representan al producto, no al TIER, es decir, el sujeto de la VC es el producto, no el TIER.
2. Nueva entrada en el registro: Se registra un nuevo conjunto de entradas en el registro descentralizado, que contiene información pública asociada al producto y un registro de cambios. Las operaciones de escritura en el registro descentralizado están sujetas a permisos. El contenido de estas entradas dependerá totalmente del caso de uso.

III-B. Actualización del PDP

El PDP se actualiza a lo largo de la cadena de suministro. Cada actualización implica necesariamente una entrada en el registro que refleje que el producto ha pasado a otra etapa en la cadena de suministro. Puede implicar también la emisión de una VC en el caso de que alguna entidad certificadora emita una VC al actor dentro de la cadena de suministro, y una entrada en el registro si esta persona o entidad registra información en el registro descentralizado referente a este producto.

III-C. Verificación del PDP

La verificación del PDP implica dos procesos. En primer lugar, se realiza una verificación SSI estándar, bajo el estándar W3C [12] para VCs. Esto implica que el titular de la VC genera una VP (Presentación Verificable) y la presenta al verificador. El segundo proceso es la verificación de la información registrada en el registro descentralizado, que se realiza haciendo una consulta a dicho registro.

III-D. Borrado o modificación del PDP

Aunque las VCs pueden eliminarse fácilmente de la *wallet*, la información registrada en el registro descentralizado no puede ser borrada ni modificada. Para evitar que no se puedan realizar modificaciones en los datos del registro descentralizado, se puede generar una nueva transacción que contenga la nueva información aplicable al PDP, o hacer una declaración que indique que la información registrada ya no es válida en el caso de un borrado.

III-E. Revocación del PDP

La revocación del PDP aplica únicamente al módulo de identidad descentralizada, ya que la revocación de la información en el registro descentralizado funciona de manera similar al borrado. Las VCs pueden ser revocadas por su emisor, indicando al verificador que no debe confiar en la información contenida en esa VC. En este sentido, la revocación se propone siguiendo el estándar W3C [12] para VCs. Además de la revocación de la VC, se necesita añadir una nueva entrada al registro para indicar que una VC asociada al PDP ha sido revocada.

III-F. Políticas de permisos

Los métodos que implican escribir o actualizar en el registro descentralizado están sujetos a permisos, por lo que es necesaria una autoridad que los asigne a los actores del ciclo de vida del PDP. La primera opción implica una autoridad centralizada, como la EC, que gestione la gobernanza del registro y asigne los permisos, pero no cumple con el requisito 3 de la EC, que establece que la información del PDP debe ser gestionada por la empresa creadora o por entidades autorizadas por ella. La segunda opción, y más adecuada, es que la empresa creadora del PDP (TIER 0) sea quien defina los permisos para escribir o actualizar en el registro. Esto sí cumple con el requisito 3, aunque exige que la red mantenga una lista blanca de entidades autorizadas, gestionada por una autoridad encargada de la gobernanza.

III-G. Esquema propuesto de VC para AAS

De acuerdo con la especificación IDTA 02006-2-0 Digital Nameplate for Industrial Equipment [13], se ha seleccionado un conjunto de campos que se incluirán en una VC, y por lo tanto formarán parte del PDP. Un nameplate contiene información identificativa, descriptiva e indicativa sobre un activo. Según el documento, la directiva de la UE 2006/42/EC establece que el nameplate de cualquier máquina debe proporcionar, al menos, la siguiente información:

(I) Nombre comercial y dirección completa del fabricante y su representante, si aplica (II) Designación de la maquinaria

(nombre común de la categoría a la que pertenece) (III) Marca CE (IV) Designación de la serie o tipo (V) Número de serie, si lo tiene (VI) Año de construcción.

Las imágenes no pueden incluirse directamente en una VC. Por lo tanto, proponemos incluir un campo de texto que contenga una imagen codificada en base64. Otra opción es incluir un enlace que refiera a la marca, pero solo se garantiza la integridad del contenido incluido en la VC (el enlace). Otras VCs emitidas por actores pueden reclamar información adicional, como certificados de calidad, certificados de seguridad, etc.

IV. Implementación

Con el objetivo de validar la arquitectura genérica propuesta, implementamos una prueba de concepto que demuestra el funcionamiento de los componentes del PDP aplicados a la información de las AAS. La implementación se basa en tecnologías Hyperledger, que en el momento de redactar este documento están incluidas bajo la iniciativa Decentralized Trust de la Linux Foundation, seleccionadas por el hecho positivo de que son de código abierto. Para el sistema de identidad descentralizada se ha elegido Aries, y para el registro descentralizado se ha utilizado Besu.

Cada actor que realiza acciones sobre la identidad de un producto en el ciclo de vida del PDP debe implementar dos componentes, cada uno conectado a su red correspondiente: el cliente Aries, para la identidad descentralizada almacenada en Indy, y el cliente Besu, para el registro descentralizado y las políticas de permisos almacenados en una red privada de Ethereum. Nuestra propuesta de implementación es una arquitectura en la que dos redes blockchain están separadas y funcionan como microservicios, lo que ofrece varias ventajas sobre los enfoques tradicionales, como una mayor escalabilidad, una mantenibilidad y actualización más sencillas, y la flexibilidad de utilizar diferentes tecnologías para distintos servicios. Se han implementado 9 operaciones, que forman parte del ciclo de vida de un PDP presentado en la Figura 2. Las que están incluidas entre los símbolos <,> son métodos estándar de SSI: (I) <Issue VC>, (II) <Presentation Request>, (III) <Send VP>, (IV) <Verify VP>, (V) <Delete VC>, (VI) <Revoke VC>, (VII) GET Entries (productID), (VIII) New Entry (productID), (IX) Verify Entries (productID),

Para fines experimentales, se han implementado dos redes blockchain: una red Indy con 4 nodos que actúa como registro de datos verificables para la parte de SSI, y una red Ethereum con 4 nodos Besu. Cada actor del ciclo de vida debe tener un agente Aries y un cliente web3j para interactuar con el sistema. Los DIDs públicos de los actores se han creado y registrado previamente en la red Indy, lo que les permite emitir VCs, las cuales se almacenan en una *wallet* en la nube implementada con Aries, simulando un custodio del PDP. Aunque cualquier DID autorizado puede emitir VCs, escribir en el nodo Besu requiere permisos. En la configuración básica, los actores de cada TIER pueden registrar cambios en el producto mediante transacciones del tipo *New Entry (productID)* en formato JSON, que documentan la transferencia del producto entre entidades y pueden incluir información pública. Asimismo, la revocación de una VC también se registra mediante una transacción en el registro descentralizado.

V. Conclusiones y trabajo futuro

Este artículo propone una arquitectura descentralizada y basada en permisos para los PDPs, abordando vacíos en la literatura al introducir un modelo de gobernanza que gestiona la custodia y autorización de datos mediante blockchain. La solución garantiza integridad, trazabilidad y cumplimiento con normativas como el Plan de Acción de la Economía Circular de la UE. El enfoque permite compartir información de forma segura y selectiva entre actores autorizados de la cadena de suministro, reduciendo fraudes y apoyando los principios de la Industria 4.0 y la economía circular. Sin embargo, se reconocen limitaciones en escalabilidad y interoperabilidad, sugiriéndose como líneas futuras el uso de soluciones de capa-2, criptografía avanzada (como ZKP y ABE), y adaptaciones sectoriales y regulatorias para diferentes productos y regiones.

Referencias

[1] S. de Diego and I. Gutierrez, "Decentralized digital product passport building blocks for enhancing supply chain sovereignty and circular economy practices," *IEEE Access*, vol. 13, pp. 137 973–137 985, 2025.

[2] K. Voulgaridis, T. Lagkas, C. M. Angelopoulos, A. A. A. Boulogeorgos, V. Argyriou, and P. Sarigiannidis, "Digital product passports as enablers of digital circular economy: a framework based on technological perspective," *Telecommunication Systems*, vol. 85, pp. 699–715, 4 2024.

[3] M. R. King, P. D. Timms, and S. Mountney, "A proposed universal definition of a digital product passport ecosystem (dppe): Worldviews, discrete capabilities, stakeholder requirements and concerns," *Journal of Cleaner Production*, vol. 384, 2023.

[4] L. Panza, G. Bruno, and F. Lombardi, "Integrating absolute sustainability and social sustainability in the digital product passport to promote industry 5.0," *Sustainability (Switzerland)*, vol. 15, 2023.

[5] T. Adisorn, L. Tholen, and T. Götz, "Towards a digital product passport fit for contributing to a circular economy," *Energies*, vol. 14, 2021.

[6] S. Ajdinović, M. Strljic, A. Lechler, and O. Riedel, "Interoperable digital product passports: An event-based approach to aggregate production data to improve sustainability and transparency in the manufacturing industry," in *2024 IEEE/SICE International Symposium on System Integration, SII 2024*. Institute of Electrical and Electronics Engineers Inc., 2024, pp. 729–734.

[7] T. Götz, "Digital product passport: the ticket to achieving a climate neutral and circular european economy?" University of Cambridge Institute for Sustainability Leadership, Cambridge, Tech. Rep., 2022. [Online]. Available: http://nbn-resolving.de/urn:nbn:de:bsz:wup4-opus-80497

[8] C. Plociennik, M. Pourjafarian, A. Nazeri, W. Windholz, S. Knetsch, J. Rickert, A. Ciroth, A. D. C. P. Lopes, T. Hagedorn, M. Vogelgesang, W. Benner, A. Gassmann, S. Bergweiler, M. Ruskowski, L. Schebek, and A. Weidenkaff, "Towards a digital lifecycle passport for the circular economy," in *Procedia CIRP*, vol. 105. Elsevier B.V., 2022, pp. 122–127.

[9] M. Pourjafarian, Christiane, M. H. Rimaz, P. Stein, M. Vogelgesang, C. Li, S. Knetsch, S. Bergweiler, and M. Ruskowski, "A multi-stakeholder digital product passport based on the asset administration shell," in *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*, vol. 2023-September. Institute of Electrical and Electronics Engineers Inc., 2023.

[10] A. Tobin and D. Reed, "The inevitable rise of self-sovereign identity: A white paper from the sovrin foundation [white paper]," 2017. [Online]. Available: https://sovrin.org/wp-content/uploads/2018/03/The-Inevitable-Rise-of-Self-Sovereign-Identity.pdf

[11] R. Kebede, A. Moscati, H. Tan, and P. Johansson, "A modular ontology modeling approach to developing digital product passports to promote circular economy in the built environment," *Sustainable Production and Consumption*, vol. 48, pp. 248–268, 7 2024. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S2352550924001362

[12] W. W. W. Consortium, "Verifiable credentials data model 2.0: W3c candidate recommendation draft," 12 2024. [Online]. Available: https://www.w3.org/TR/vc-data-model-2.0/

[13] IDTA, "Idta 02006-2-0 digital nameplate for industrial equipment. submodel template of the asset administration shell," 10 2022. [Online]. Available: https://industrialdigitaltwin.org/wp-content/uploads/2022/10/IDTA-02006-2-0_Submodel_Digital-Nameplate.pdf

Detección de Malware utilizando Lógica difusa y Machine Learning

Alfonso Gotor Burguete[1], Bruce W. Percílio Azevedo[1,2], Robson de Oliveira Albuquerque[1,3], Luis Javier García Villalba[1]

[1]Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid, España
Email: {alfonbur, robsonde, javierv}@ucm.es

[2]Universidad Politécnica de Madrid (UPM), Madrid, España
Email: bruce.percilio@alumnos.upm.es

[3]Universidad de Brasilia (UnB), Brasil
Email: robson@redes.unb.br

***Resumen*—La detección de malware basada en similitud y umbralización identifica software malicioso midiendo la semejanza de una muestra desconocida frente a un repositorio de muestras conocidas, y asignándola a una familia cuando la similitud supera un umbral. En este trabajo proponemos un pipeline reproducible que combina: (i) preprocesado de volcados binarios para eliminar ruido (p. ej., posiciones dominadas por bytes nulos `0x00`), (ii) una métrica de similitud a nivel byte (definida formalmente en la Sección V-B) y (iii) una capa de inferencia difusa que transforma la similitud "crisp" en una evaluación gradual (baja/media/alta) para producir un nivel de riesgo interpretable. La evaluación cuantitativa del criterio de similitud y umbral se realiza sobre el banco Juliet/ROMEO, utilizado como conjunto etiquetado para analizar separación intra/inter-grupo y calibrar el umbral de decisión. En particular, se adopta $\tau = 0{,}50$ como punto operativo, tras descartar umbrales permisivos (0.20–0.30) por exceso de falsos positivos y umbrales estrictos (0.80–0.90) por aumento de falsos negativos. Como demostración cualitativa del flujo de evidencias, incluimos un caso de estudio de buffer overflow (programa vulnerable vs. corregido) que muestra cómo la capa difusa eleva o reduce el riesgo al incorporar señales dinámicas (p. ej., fallo reproducible bajo entrada límite) junto con rasgos estáticos (APIs inseguras y mitigaciones de compilación).**

***Index Terms*—Detección de malware, similitud binaria, umbralización, *fuzzy hashing*, lógica difusa, aprendizaje automático, análisis dinámico en contenedor.**

I. Introducción

La proliferación de malware ha alcanzado niveles alarmantes en los últimos años. Según estudios recientes, el número de programas maliciosos conocidos superó los mil millones en 2023, con cientos de miles de nuevas muestras apareciendo cada día [1]. Esta abundancia de variantes, muchas de las cuales son modificaciones menores de familias existentes, desafía los métodos tradicionales de detección basados en firmas exactas. Los atacantes emplean técnicas de ofuscación y metamorfismo para generar malware nuevo que evite las firmas conocidas, haciendo que los enfoques clásicos resulten insuficientes para detectar amenazas de día cero.

Para afrontar este problema, la comunidad de seguridad ha explorado métodos de detección de malware basados en análisis de similitud. En lugar de buscar una coincidencia exacta con una firma, estos métodos miden qué tan similar es un archivo desconocido a otros archivos maliciosos conocidos [3], [4]. La premisa es que variantes de la misma familia de malware compartirán características de código o comportamiento similares, aun si difieren en detalles superficiales. Por ejemplo, técnicas pioneras emplearon distancias de edición entre secuencias de instrucciones ensambladas o comparaciones de gráficas de llamadas a funciones para agrupar malware en familias [5]. Otros trabajos calcularon vectores de características (frecuencias de n-gramas de bytes o instrucciones) y usaron medidas de distancia (p. ej., coseno o Jaccard) para determinar similitud entre binarios [6]. Incluso se han aplicado métodos de hashing difuso (fuzzy hashing) como ssdeep [7] o sdhash para obtener digests que permiten comparar archivos con tolerancia a cambios menores.

Estos enfoques basados en similitud han demostrado ser eficaces para identificar malware polimórfico y variantes evolucionadas [8]. Al clasificar automáticamente muestras nuevas en clústeres según su parecido con malware conocido, es posible detectar amenazas emergentes antes de que existan firmas específicas para ellas. Sin embargo, un componente crítico en este proceso es la elección de un umbral de similitud adecuado. Si el umbral es muy bajo (permisivo), se agruparán juntos archivos que quizás no sean de la misma familia (aumentando falsos positivos). Por el contrario, un umbral demasiado alto (estricto) podría dejar fuera variantes genuinamente relacionadas que presentan diferencias significativas (aumentando falsos negativos) [13]. Por tanto, seleccionar el umbral óptimo es fundamental para equilibrar la sensibilidad y la precisión de la detección.

En este trabajo presentamos un sistema de detección de malware basado en análisis de similitud y umbralización. El sistema extrae características de las muestras (estáticas de código) y calcula una métrica de similitud par a par. Si la similitud entre una muestra desconocida y algún malware conocido supera cierto umbral, la muestra se considera malware (posiblemente de la misma familia); si no, se considera benigna o desconocida. Para entrenar y validar nuestro método, empleamos un conjunto de datos extenso

obtenido del Juliet Test Suite y de su derivado compilado ROMEO [10], [11]. En las siguientes secciones detallamos la metodología: construcción del banco de pruebas y repositorios de referencia (Sección V-A), definición de la métrica de similitud y calibración del umbral, y la capa de inferencia difusa para fusión de evidencias. La evaluación cuantitativa se centra en el comportamiento del criterio de similitud y umbral (capacidad de separación y punto de operación), mientras que la parte cualitativa utiliza un buffer overflow como caso de estudio controlado para mostrar la integración de evidencias estáticas y dinámicas dentro de un flujo aplicable a malware en general.

II. Funciones de membresía y evaluación difusa de la similitud

En esta sección introducimos lógica difusa como mecanismo para transformar una medida de similitud "crisp" en una valoración gradual que facilite la toma de decisiones. En lugar de decidir de forma binaria si dos binarios son "similares" o "no similares", el sistema asigna grados de pertenencia a categorías lingüísticas (`low`, `medium`, `high`) sobre el porcentaje de similitud.

II-A. Similitud crisp como entrada

Dadas dos secuencias de bytes $x = (x_1, \ldots, x_L)$ e $y = (y_1, \ldots, y_L)$ (alineadas a la misma longitud), definimos el conjunto de índices informativos

$$I = \{\, i \in \{1, \ldots, L\} \mid \neg(x_i = \texttt{0x00} \ \wedge \ y_i = \texttt{0x00}) \,\}, \tag{1}$$

y la similitud a nivel byte como:

$$Sim(x, y) = \frac{1}{|I|} \sum_{i \in I} \mathbb{1}[x_i = y_i], \qquad Sim(x, y) \in [0, 1]. \tag{2}$$

La variable difusa opera sobre $s = 100 \cdot Sim(x, y) \in [0, 100]$.

II-B. Funciones de membresía (parametrización explícita)

Definimos la variable lingüística `similarity_level` con tres conjuntos difusos en el dominio $s \in [0, 100]$:

$$\mu_{low}(s) = \begin{cases} 1 - \frac{s}{50}, & 0 \le s \le 50, \\ 0, & 50 < s \le 100, \end{cases} \tag{3}$$

$$\mu_{medium}(s) = \begin{cases} 0, & s < 25 \text{ o } s > 75, \\ \frac{s-25}{25}, & 25 \le s \le 50, \\ \frac{75-s}{25}, & 50 < s \le 75, \end{cases} \tag{4}$$

$$\mu_{high}(s) = \begin{cases} 0, & 0 \le s \le 50, \\ \frac{s-50}{50}, & 50 < s \le 100. \end{cases} \tag{5}$$

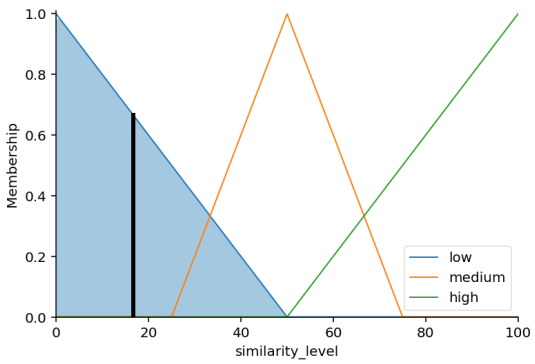


Figura 1. Funciones de membresía para `similarity_level` con puntos de quiebre 25/50/75 (`low`/`medium`/`high`).

II-C. Fuzzificación (ejemplo numérico)

En el caso de estudio (Sección III), se analizaron 18 volcados asociados a buffer overflow. Tras eliminar posiciones dominadas por `0x00`, la similitud byte-a-byte observada fue $s = 0{,}89$ (i.e., $Sim = 0{,}0089$). Con las funciones anteriores se obtiene:

$$\mu_{low}(0{,}89) = 1 - \frac{0{,}89}{50} = 0{,}9822, \quad \mu_{medium}(0{,}89) = 0, \quad \mu_{high}(0{,}89) = 0.$$

Para defuzzificación se puede emplear centroide; en ese caso, cuando la salida activa prácticamente solo `low`, el valor defuzzificado tiende al centroide geométrico del conjunto `low` (en torno a 16,7). En cualquier caso, el defuzzificado se interpreta como indicador agregado del sistema difuso, no como una "nueva" similitud byte-a-byte.

II-D. Racional del filtrado de `0x00`

Eliminar posiciones dominadas por bytes nulos evita que regiones vacías o de relleno sesguen $Sim(x, y)$, y fuerza a comparar contenido con carga informativa.

III. Caso de estudio: Buffer Overflow como ejemplo de evidencia explotable en binarios

Un *buffer overflow* es una condición de corrupción de memoria que aparece cuando un programa escribe más datos de los que un búfer puede almacenar, pudiendo provocar fallo de ejecución, corrupción silenciosa o incluso desviación del flujo de control. Aunque el objetivo del artículo es la detección de malware en general, utilizamos este caso como ejemplo controlado por tres razones: (i) permite generar binarios reproducibles (vulnerable vs. corregido), (ii) produce señales estáticas interpretables (uso de APIs inseguras, mitigaciones) y (iii) facilita evidencias dinámicas simples (inestabilidad bajo entradas límite) dentro de un contenedor. El caso de estudio no pretende representar la totalidad del fenómeno malware (p. ej., persistencia, C2 o movimiento lateral), sino ilustrar cómo el sistema propuesto fusiona evidencias heterogéneas. En un escenario de malware real, la evidencia dinámica equivalente se obtiene mediante ejecución en *sandbox* (trazas de llamadas, acceso a red, modificaciones de sistema, etc.) y se integra de la misma forma en la capa difusa.

III-A. Problemas y Consecuencias

Las consecuencias de un buffer overflow son variadas y, en muchos casos, críticas:

- **Ejecución Arbitraria de Código:** Un atacante puede inyectar código malicioso en la memoria, redirigiendo la ejecución del programa hacia instrucciones perjudiciales.
- **Elevación de Privilegios:** Al explotar la vulnerabilidad, es posible obtener derechos administrativos o de sistema, comprometiendo la seguridad general del dispositivo.
- **Denegación de Servicio (DoS):** La corrupción de la memoria puede llevar a la interrupción del funcionamiento normal del sistema, provocando bloqueos o reinicios.
- **Corrupción de Datos:** La manipulación de la memoria puede alterar datos críticos, generando errores en la integridad de la información.

Históricamente, los buffer overflows han sido la puerta de entrada a muchos de los ataques informáticos más notorios, y a pesar de la evolución de técnicas de programación segura, su detección sigue siendo un desafío importante.

IV. Trabajos Similares y Previos en la Aplicación de Lógica Difusa para la Detección de Malware

La detección de malware en entornos con alta incertidumbre y variabilidad en las características de las muestras ha impulsado el uso de técnicas basadas en lógica difusa. Diversos estudios han explorado cómo esta metodología puede complementar los enfoques tradicionales, ofreciendo una mayor flexibilidad al modelar la imprecisión en los indicadores de comportamiento malicioso. A continuación, se destacan algunas contribuciones relevantes:

- **Atacak et al. (2023)**: En [18] se propone un enfoque para la detección de malware en dispositivos Android que integra lógica difusa con clasificadores basados en machine learning. El método utiliza reglas difusas para evaluar múltiples indicadores de seguridad, permitiendo que el sistema maneje de manera más efectiva la incertidumbre y las variaciones en el comportamiento de las aplicaciones. Los resultados experimentales demuestran que la incorporación de lógica difusa mejora significativamente la precisión del detector, sobre todo en casos donde la división entre muestras benignas y maliciosas no es nítida.
- **Almutairi et al. (2018)**: Aunque centrado en la detección de ataques de Cross-Site Scripting en aplicaciones web [19], este trabajo evidencia la eficacia de la lógica difusa para capturar comportamientos anómalos y gestionar la incertidumbre inherente en la identificación de amenazas. La metodología basada en reglas difusas presentada en este estudio puede ser adaptada para otros dominios, incluido el análisis de malware, permitiendo la evaluación flexible de múltiples indicadores de riesgo.

Estos trabajos resaltan la versatilidad de la lógica difusa al abordar problemas de clasificación en entornos de seguridad, donde la ambigüedad de las características y la presencia de muestras límite hacen necesaria una aproximación adaptable. En el presente trabajo, la integración de lógica difusa con análisis de similitud y técnicas de machine learning se fundamenta en estas contribuciones, buscando mejorar la asignación de umbrales y la detección precisa de variantes de malware.

V. Metodología

En esta sección se describe el enfoque propuesto para la detección de malware mediante análisis de similitud y umbral. Primero, explicamos la construcción y características del conjunto de datos empleado, enfatizando cómo se obtuvieron los datos de Juliet y ROMEO, su estructura y calidad. Luego detallamos el método de análisis de similitud aplicado a las muestras de malware y la selección del umbral óptimo para la clasificación.

V-A. Obtención del dataset

Para evaluar nuestra propuesta utilizamos un dataset derivado del Juliet Test Suite y de su correspondiente versión compilada conocida como ROMEO. Juliet es una colección amplia de casos de prueba de código fuente en C/C++ y Java que contienen ejemplos tanto vulnerables como seguros para una variedad de debilidades de software [10]. Originalmente desarrollado para probar herramientas de análisis estático, Juliet abarca 91 categorías de vulnerabilidades de la lista CWE (Common Weakness Enumeration), proporcionando más de 100 000 funciones etiquetadas como *good* (benignas) o *bad* (con una vulnerabilidad explícita). Cada caso de prueba de Juliet consiste típicamente en una función que ilustra una vulnerabilidad (por ejemplo, un buffer overflow) y una función alternativa segura que corrige dicho problema. Esta dualidad de funciones buenas/malas ofrece un terreno controlado para experimentación, con un etiquetado de alta calidad garantizado por su naturaleza sintética y revisada.

El dataset ROMEO, introducido por Brust et al. [11], se construye compilando las pruebas de Juliet para obtener sus binarios correspondientes y luego extrayendo de ellos representaciones ensambladas. En esencia, ROMEO es un banco de datos de vulnerabilidades a nivel binario, derivado directamente de Juliet, que preserva la etiqueta (tipo de vulnerabilidad) para cada fragmento de código compilado. El proceso de generación de ROMEO implica desensamblar los ejecutables obtenidos de Juliet y extraer las secciones de código (`.text`) relevantes, asegurando que no se filtre información del label durante la compilación (por ejemplo, eliminando funciones benignas duplicadas que pudieran delatar la etiqueta) [11]. El resultado es un conjunto reproducible y público de 168 000 muestras etiquetadas en 91 categorías CWE. Además de representaciones ensambladas, ROMEO proporciona imágenes binarias precomputadas de memoria (segmentos `.text`, `.data` y heap) junto con metadatos y etiquetas, lo que permite trabajar de forma directa con secuencias de bytes normalizadas sin depender de instrumentación de volcado en tiempo de ejecución.

En nuestro trabajo utilizamos ROMEO como banco de pruebas controlado para estudiar separación por similitud, calibración de umbrales y comportamiento intra/inter-grupo en binarios etiquetados. Aunque ROMEO está orientado a debilidades (CWE) y no a familias de malware reales, ofrece ventajas experimentales (etiquetas fiables y gran volumen) que permiten validar el criterio de similitud y la integración de evidencias. En despliegues operativos, el mismo pipeline se aplica sobre repositorios de malware reales, manteniendo la misma definición de similitud y la misma fusión difusa, pero reemplazando las etiquetas del banco por familias/tipos de malware y/o por políticas de riesgo.

Antes de proceder al análisis, preprocesamos los datos de ROMEO para adecuarlos a nuestra metodología de similitud de malware. Esto incluyó la normalización de las representaciones de ensamblador (por ejemplo, removiendo símbolos volátiles como direcciones absolutas o nombres de variables temporales, que pueden diferir entre compilaciones) y la extracción de características relevantes de cada muestra. Entre las características evaluadas se encontraron:

- **Secuencias de instrucciones**: listas o n-gramas de opcodes extraídos de cada función ensamblada.
- **Hashes difusos**: digest de similitud calculados con técnicas como *ssdeep* [7] o TLSH para cada binario, que permiten comparaciones tolerantes a cambios menores.
- **Gráficos de flujo de control**: estructuras grafo de cada función (nodos corresponden a bloques básicos y aristas a saltos), que capturan la estructura lógica del código.
- **Características estadísticas**: vectores con métricas como distribución de bytes, entropía, frecuencias de llamadas a API específicas, etc.

Adicionalmente, para reducir el impacto de regiones de relleno o baja información, filtramos padding frecuente (p. ej., secuencias largas de `0x00`) y normalizamos el desensamblado eliminando elementos volátiles (direcciones absolutas, etiquetas temporales). Este preprocesado se aplica antes de: (i) calcular fuzzy hashes sobre la región de código y (ii) extraer *n*-gramas de opcodes para la similitud de Jaccard. Como comprobación opcional (no usada en la decisión), puede estimarse una coincidencia byte-a-byte sobre las secuencias filtradas para detectar sesgos por relleno.

Estas distintas representaciones nos proporcionan múltiples formas de medir similitud. En nuestros experimentos, los mejores resultados se obtuvieron combinando dos enfoques complementarios: hashing difuso de la sección de código y comparativa de secuencias de opcodes. La combinación permite aprovechar tanto una visión de alto nivel (hash difuso resume globalmente el contenido binario [7]) como un nivel más fino de detalle secuencial.

V-B. Análisis de similitud y definición de umbral

Con el dataset de referencia preparado, nuestro sistema calcula una medida de similitud entre pares de muestras. Formalmente, definimos $Sim(x, y) \in [0, 1]$ como una combinación ponderada de (i) fuzzy hashing sobre la región de código y (ii) similitud de Jaccard sobre *n*-gramas de opcodes extraídos del desensamblado:

- **Jaccard sobre conjuntos de características**: por ejemplo, comparar el conjunto de *k*-gramas de instrucciones presentes en x y y usando la similitud de Jaccard $Sim_J = \frac{|K_x \cap K_y|}{|K_x \cup K_y|}$. Esta métrica es útil para capturar coincidencias de fragmentos de código.
- **Distancia de edición**: medir la mínima cantidad de operaciones para transformar la secuencia de opcodes de x en la de y (por ejemplo, utilizando distancia de Levenshtein normalizada). Una distancia menor implica mayor similitud.
- **Hashing difuso**: calcular el puntaje de coincidencia producido por funciones de hashing como *ssdeep* o *sdhash* al comparar x con y. Estas funciones producen un porcentaje de similitud estimado en base a patrones de bytes comunes.
- **Similitud de gráficas**: si representamos cada binario por su grafo de flujo de control, podemos definir $Sim(x, y)$ en términos del subgrafo común más grande o mediante algoritmos de alineamiento de grafos [3], [5]. Este enfoque estructural puede ser robusto contra cambios menores en código que no alteran la lógica general.

En nuestra implementación, utilizamos principalmente la métrica de hashing difuso combinada con Jaccard de *n*-gramas de opcodes, ya que ofrecieron buen equilibrio entre precisión y costo computacional.

$$Sim(x, y) = w_{FH} \cdot Sim_{FH}(x, y) + (1 - w_{FH}) \cdot Sim_J(x, y), \quad (6)$$

donde $Sim_{FH}(x, y) \in [0, 1]$ es el *score* normalizado de coincidencia entre *fuzzy hashes* (p. ej., ssdeep/TLSH) y $Sim_J(x, y)$ es la similitud de Jaccard entre conjuntos de *n*-gramas de opcodes: $Sim_J = \frac{|K_x \cap K_y|}{|K_x \cup K_y|}$. El parámetro $w_{FH} \in [0, 1]$ controla la contribución relativa de cada componente.

En los experimentos, se emplearon *n*-gramas de opcodes con $n =$ N y un peso $w_{FH} =$ W (con $w_{FH} + (1 - w_{FH}) = 1$). Ambos parámetros se mantuvieron fijos en todas las evaluaciones reportadas. Aunque existen alternativas (distancia de edición, similitud estructural de CFG, etc.), en este trabajo la evaluación cuantitativa y la umbralización se realizan exclusivamente con la definición de Sim de la Ecuación 6.

Una vez que disponemos de una función de similitud confiable, el siguiente paso es aplicar un umbral τ para tomar decisiones de clasificación. El criterio es sencillo: dada una muestra desconocida m, calculamos $Sim(m, s)$ respecto a cada muestra s del conjunto de malware conocido (nuestro dataset ROMEO, que hace las veces de base de datos de malware). Si existe al menos un s tal que $Sim(m, s) \geq \tau$, clasificamos m como malware (y potencialmente asignado a la familia/tipo de s más similar). En caso contrario ($Sim(m, s) < \tau$ para todo s conocido), consideramos m como benigno o, cuanto menos, sin similitud suficiente con ningún malware registrado.

La elección del umbral τ es crítica. Un valor de τ demasiado bajo provocaría muchas detecciones (alta sensibilidad) pero también puede conllevar falsos positivos al agrupar artefactos que no son realmente malware similar. Por otro lado, un τ demasiado alto reduciría falsos positivos pero a costa de no detectar variantes que sí están relacionadas (disminuyendo la tasa de verdaderos positivos). Para determinar un τ adecuado, procedimos de la siguiente manera:

1. **Distribución de similitudes**: analizamos la distribución de valores $Sim(x, y)$ para pares de muestras de la misma familia (ejemplos dentro de la misma categoría CWE en nuestro dataset) y para pares de familias distintas. Encontramos que las similitudes intrafamilia tienden a ser significativamente más altas que las interfamilia. En la Figura 1 (discutida en la sección de Resultados) se ilustra este comportamiento: por ejemplo, la similitud promedio entre dos muestras cualesquiera de la misma categoría CWE fue de 0.75, mientras que entre categorías distintas estuvo alrededor de 0.20. Existe una zona de solapamiento moderada, por lo que ningún umbral separa perfectamente todos los casos; sin embargo, se observa un "hueco" en la distribución alrededor de $\sim 0{,}5$.
2. **Curva ROC y punto óptimo**: realizamos un barrido del umbral τ entre 0 y 1 y, para cada valor, evaluamos la tasa de verdaderos positivos (TPR, proporción de malware correctamente identificado) y la tasa de falsos positivos (FPR, proporción de benignos incorrectamente etiquetados como malware) usando nuestro dataset. Esto permite trazar una curva ROC. El punto de máxima eficacia se determinó aproximadamente en $\tau = 0{,}50$ donde se logra un buen equilibrio: se detectan la mayoría de los malware conocidos sin un incremento drástico de falsas alarmas.
3. **Validación cruzada**: dividimos el conjunto de muestras conocidas en subconjuntos y verificamos que el umbral elegido generalizaba bien. Es decir, que no estaba ajustado únicamente a un caso particular. En todas las particiones, τ alrededor de 0.5–0.6 ofreció resultados cercanos al óptimo.

Cabe destacar que este resultado concuerda con hallazgos reportados en estudios independientes. Botacin et al. [9],

al analizar funciones de hashing de similitud en escenarios reales, encontraron que un umbral de coincidencia del 50 % representaba el mejor equilibrio entre verdaderos positivos y falsos negativos al asistir la detección de malware con estos métodos. Del mismo modo, otros autores han señalado que umbrales demasiado laxos pueden agrupar malware que sólo comparten similitudes superficiales, mientras que umbrales muy estrictos pierden la capacidad de generalizar frente a variantes ofuscadas [13].

En síntesis, la metodología se resume así: contamos con un repertorio de muestras maliciosas con sus características (ROMEO), definimos una métrica Sim que cuantifica parecido entre binarios, y determinamos un umbral τ adecuado mediante análisis estadístico. El sistema resultante puede, dado un nuevo binario, asignarlo a una familia existente (si su similitud con esa familia supera τ) o marcarlo como potencialmente benigno o novedoso en caso contrario. En la siguiente sección presentaremos los resultados de aplicar este método, mostrando cuantitativamente su desempeño y cualitativamente cómo logra reagrupar diferentes tipos de malware.

V-C. Módulo de Machine Learning y fusión con inferencia difusa

Además del emparejamiento por similitud, incorporamos un módulo de machine learning para aprender patrones discriminativos a partir de representaciones numéricas extraídas de muestras binarias (p. ej., distribuciones de bytes, histogramas de opcodes y métricas de flujo de control). El pipeline se compone de: (i) ingesta y vectorización de muestras, (ii) clasificación supervisada con modelos base (árboles de decisión, SVM) y modelos profundos (autoencoders y arquitecturas generativas), y (iii) fusión de evidencias mediante un sistema difuso tipo Mamdani.

Sea $p_{ML}(m) \in [0,1]$ la puntuación devuelta por el clasificador para una muestra m (probabilidad o score de pertenencia a la clase maliciosa/vulnerable). La decisión final no depende de una sola señal: se integra la similitud $Sim(m,s)$, el score $p_{ML}(m)$ y evidencias auxiliares (p. ej., presencia de APIs inseguras o inestabilidad observada en ejecución contenida). Un sistema difuso con funciones de pertenencia triangulares/trapezoidales y reglas lingüísticas (p. ej., bajo/medio/alto) realiza la agregación y defuzzificación para producir un nivel de riesgo interpretable (Bajo/Medio/Alto) y/o un valor continuo.

Este diseño permite que la similitud capture relaciones tipo "familia/variantes" y que el clasificador ML capture patrones globales no necesariamente reflejados por coincidencias por pares, reduciendo fragilidad ante cambios superficiales.

VI. Resultados

A continuación, presentamos los resultados de los experimentos realizados para evaluar la eficacia de la detección de malware por similitud y umbral. Nos enfocamos en la relevancia de las métricas obtenidas y en entender cómo se relacionan los distintos tipos de malware entre sí en términos de similitud, así como el impacto del umbral utilizado en dichas métricas.

VI-A. Distribución de similitudes intra e interfamilia

Se analizaron puntuaciones Sim distinguiendo pares intragrupo (misma categoría CWE) y pares intergrupo (categorías distintas). Se observa:

- Los pares intragrupo presentan una distribución unimodal con modo en $Sim = 0{,}8$.
- Los pares intergrupo presentan una distribución unimodal con modo en $Sim = 0{,}2$ y masa principal en $Sim < 0{,}4$.
- El solapamiento entre ambas distribuciones se concentra en el intervalo $Sim \in [0{,}40, 0{,}60]$.

En consecuencia, se fija $\tau = 0{,}50$ como umbral operativo, al situarse en la región de separación entre distribuciones y maximizar el compromiso entre TPR y FPR.

Estos hallazgos confirman la viabilidad del criterio de similitud para distinguir tipos de malware: en general, existe un buen margen para separar dos distribuciones (intra vs. inter) con un umbral apropiado. Tomando en cuenta estas distribuciones, seleccionamos $\tau = 0{,}50$ como umbral de decisión principal para las siguientes evaluaciones, puesto que cae aproximadamente en la mitad del valle que separa ambas curvas, alineado con la recomendación de [9].

VI-B. Rendimiento de detección según el umbral

Para cuantificar el desempeño del detector, medimos varias métricas clásicas bajo distintos valores de umbral. La Tabla I resume los resultados clave para umbrales desde 0.4 hasta 0.7. Se reportan la tasa de verdaderos positivos (TPR), tasa de falsos positivos (FPR), precisión (porcentaje de muestras clasificadas como malware que realmente lo eran) y la métrica F_1 (armonía entre precisión y TPR).

Tabla I
DESEMPEÑO DEL DETECTOR PARA DIFERENTES VALORES DE UMBRAL τ DE SIMILITUD.

Umbral τ	TPR (%)	FPR (%)	Precisión (%)	F_1 (%)
0.40	98.5	15.2	86.6	91.3
0.50	95.1	5.4	95.8	95.4
0.60	88.7	2.0	98.1	93.1
0.70	80.4	0.8	99.0	88.9

Como era de esperarse, un umbral más bajo conduce a mayor sensibilidad (TPR) a costa de más falsos positivos, mientras que umbrales altos hacen la detección más conservadora:

- A $\tau = 0{,}40$, la tasa de detección prácticamente alcanza el máximo (98.5 % de los malware conocidos fueron reconocidos), pero a costa de un 15 % de falsos positivos, lo cual sería inaceptable en un entorno real (muchos archivos benignos podrían ser marcados erróneamente). Este umbral tan permisivo agrupa incluso similitudes débiles, causando sobreclasificación.
- En el rango medio ($\tau = 0{,}50$ a $0{,}60$) se observa el mejor equilibrio. Con $\tau = 0{,}50$, obtenemos TPR de 95 % con sólo 5.4 % FPR; la precisión resultante es superior al 95 %, lo que implica que la mayoría de las alertas corresponden efectivamente a malware. El puntaje F_1 alcanza 95.4 %, el más alto de la tabla, indicando el punto de mejor compromiso entre detectar la mayor cantidad de malware y minimizar falsos positivos. Este

resultado apoya la selección de $\tau = 0{,}5$ como óptima para nuestro sistema.

- Al subir a $\tau = 0{,}60$, la TPR cae a 89 %, indicando que nos perdemos alrededor de un 11 % de las muestras maliciosas (falsos negativos) que eran más atípicas en su familia. No obstante, la precisión sube ligeramente (98 %) y la FPR baja al 2 %. Dependiendo de las necesidades (p.ej., en un entorno donde se priorice muy pocos falsos positivos), $\tau = 0{,}60$ podría considerarse aceptable. Sin embargo, en nuestro caso preferimos maximizar la detección, por lo que un ligero incremento de falsos positivos al usar 0.50 fue tolerable a cambio de 6 % más TPR.
- Con $\tau = 0{,}70$, el sistema se vuelve muy estricto: aún captura el 80 % del malware, pero prácticamente elimina falsos positivos (¡1 %). La precisión llega a 99 %, pero el F_1 desciende a 88.9 %, evidenciando la pérdida de detección.

Estos resultados empíricos validan que el umbral $\tau \approx 0{,}5$ es una buena elección, coherente con la intuición extraída de las distribuciones de similitud. Además, concuerdan con la literatura en cuanto al efecto del umbral: valores bajos maximizan la cobertura (útil quizás en un escenario de priorización de análisis, como sugieren [9]), mientras que valores más altos sirven para confirmación muy confiable (por ejemplo, herramientas de *incident response* podrían usar τ alto para identificar sólo los casos más claramente vinculados a una amenaza conocida).

VI-C. Robustez del detector y evaluación adversaria

Se evaluó la generación de muestras adversarias frente a un detector basado en aprendizaje automático (implementación tipo CNN sobre representaciones de bytes/características normalizadas) mediante dos estrategias complementarias: (i) generación con modelos generativos (GAN) y (ii) perturbación guiada por gradiente con feature injection (inyección de secuencias extraídas de ficheros benignos en regiones no ejecutables o huecos del fichero).

En la estrategia de inyección guiada, la tasa de evasión fue un 30–60 % superior a una línea base de inserción aleatoria de bytes. Frente a un ataque iterativo puramente por gradiente (modificación byte a byte), la inyección guiada consiguió entre 6 y 10 puntos porcentuales adicionales de evasión contra el detector CNN. En coste, tras entrenar el modelo (10 horas), la generación de cada muestra adversaria requirió 1 minuto, frente a 100 minutos por muestra en la estrategia iterativa byte a byte.

Estos resultados cuantifican la sensibilidad del componente ML ante evasiones y justifican contramedidas como entrenamiento adversario y ensembles.

VI-D. Similitud entre distintos tipos de malware

Un aspecto relevante es cómo nuestro método refleja la similitud relativa entre distintos tipos de malware, más allá de simplemente distinguir si son iguales o no. Para explorar esto, generamos una matriz de similitud promedio entre categorías seleccionadas de nuestro dataset (Tabla II). Escogimos cuatro categorías de vulnerabilidad de Juliet (a modo de analogía, cuatro familias de malware) y calculamos el puntaje Sim promedio entre cada par de categorías (calculado sobre múltiples combinaciones de muestras de ambas).

Tabla II
MATRIZ DE SIMILITUD PROMEDIO ENTRE CUATRO CATEGORÍAS DE MUESTRAS (VALORES EN %). LAS CELDAS EN LA DIAGONAL (EN NEGRITA) REPRESENTAN LA SIMILITUD INTRAFAMILIA PROMEDIO, MIENTRAS QUE LAS CELDAS FUERA DE LA DIAGONAL SON INTERFAMILIA.

Familia	A	B	C	D
A (Overflow)	**78**	22	18	20
B (SQLi)	24	**74**	19	21
C (XSS)	17	15	**80**	16
D (Crypto)	21	23	18	**72**

En este ejemplo, A–D representan clases del banco de pruebas (p. ej., categorías etiquetadas) utilizadas para analizar separación intra/inter-grupo mediante similitud. En un escenario de malware real, estas clases se corresponderían con familias (p. ej., troyanos bancarios, downloaders, botnets, ransomware), y el mismo análisis permite estimar cohesión y solapamientos entre familias.

- Las celdas en la diagonal (en negrita) son altas, rondando 72–80 %. Esto reafirma que nuestras medidas de similitud capturan bien la cohesión dentro de la misma familia: p.ej., dos muestras de la familia A (Overflow) comparten en promedio un 78 % de similitud.
- Las celdas fuera de la diagonal son mucho menores (15–24 %). Familias muy distintas, como C (XSS) y B (SQLi), muestran apenas 19 % de similitud en promedio, lo cual es cercano al azar, indicando que prácticamente no comparten patrones de código. En algunos casos, notamos ligeramente más similitud entre ciertas parejas (por ejemplo, B y D alrededor de 23–24 %), posiblemente porque comparten alguna funcionalidad común (imaginemos que ambas vulnerabilidades tienen código de manejo de strings similar, etc.). Aún así, esos porcentajes están muy por debajo de los intrafamilia.

Este análisis de la matriz de similitud nos permite entender que nuestro sistema no sólo decide binariamente malware/benigno, sino que también puede ayudar a clusterizar o relacionar familias. En un escenario práctico de análisis de malware real, esto significaría que, dado un nuevo espécimen malicioso, podríamos identificar a cuál familia conocida se asemeja más (por ejemplo, ."este ejecutable desconocido tiene un 76 % de similitud con la familia Zeus, pero sólo 30 % con Mirai, por lo que probablemente sea una variante de *Zeus*"). Herramientas como ésta pueden asistir a analistas a entender la procedencia de nuevas muestras y a mapear linajes de malware.

VI-E. Visualización de agrupamiento

Para ilustrar gráficamente la agrupación por similitud, la Figura 2 muestra un diagrama de dispersión (proyección en 2D mediante PCA o *t-SNE*) de muestras representativas, coloreadas según su familia verdadera. Se puede observar que las muestras naturalmente forman clústeres densos por color (familia), confirmando que la similitud capturada es coherente con la clasificación real. Los pocos puntos mal ubicados (colores mezclados) corresponden a aquellos casos de traslapo discutidos; por ejemplo, un punto de la familia A cercano al

grupo de familia B, que resultó ser un caso atípico dentro de A con similitud inusualmente baja hacia sus congéneres. No obstante, en general los clústeres están bien definidos.

En cuanto a rendimiento computacional, nuestros experimentos con este dataset (decenas de miles de muestras) indican que la estrategia es escalable. Usando hashing difuso y comparación de vectores optimizada, la comparación de una muestra nueva contra la base de datos de referencia toma del orden de segundos en un PC estándar. Métodos más avanzados, como indexado de características o técnicas de approximate nearest neighbors [2], podrían acelerar aún más la búsqueda de similitudes en bases de millones de muestras, lo que sugiere viabilidad para entornos operativos reales donde los repositorios de malware son masivos.

En resumen, los resultados muestran que:

- El método logra altas tasas de detección (95 %+ TPR) con bajos falsos positivos, siempre que se elija un umbral adecuado (en nuestro caso 0.5).
- Las diferentes familias de malware están claramente separadas en el espacio de similitud definido, validando la capacidad del análisis para reconocer patrones de código compartido.
- Un umbral intermedio permite capturar la mayoría de las variantes sin sobrepasar con falsos positivos. Este hallazgo coincide con trabajos previos que recomendaron calibrar cuidadosamente la sensibilidad de los hashes de similitud para usarlos efectivamente en detección [9].
- La presentación de resultados mediante tablas y figuras nos permitió evidenciar de forma tangible la efectividad del enfoque: las tablas con métricas cuantifican la eficacia global, mientras que las figuras comparativas (distribuciones de similitud, diagramas de clústeres) ilustran la separación entre tipos de malware que fundamenta dicha eficacia.

VI-F. Implementación con Docker y Postman

Toda la lógica anterior se implementó dentro del contenedor Docker mencionado, utilizando lenguajes de scripting y herramientas disponibles en Linux. Por ejemplo, para el análisis estático se emplearon utilidades como `objdump` o `radare2` para obtener cadenas de texto y llamadas a funciones desde el binario. Para el sistema difuso, se utilizó una biblioteca de lógica difusa (como scikit-fuzzy en Python) que procesase las métricas calculadas.

El flujo completo de uso es el siguiente: un analista de seguridad, a través de Postman, envía una petición POST al servicio REST del contenedor, adjuntando el archivo .o sospechoso. El servicio recibe el archivo, ejecuta automáticamente el análisis estático y dinámico, y luego aplica el sistema difuso a los resultados obtenidos. Finalmente, responde a la petición con un informe en formato JSON que contiene el nivel de riesgo evaluado y un desglose de las condiciones encontradas (por ejemplo: "RiesgoMalware: Alto; Funciones inseguras detectadas: 3; Ejecución inestable en 2/5 pruebas").

La carpeta de pruebas dentro del Docker contiene los casos de test mencionados (entradas preparadas). Este conjunto de tests puede ampliarse fácilmente para adaptarse al contexto del programa analizado. Por ejemplo, si se sospecha que el programa procesa imágenes o archivos binarios, los casos de prueba podrían incluir archivos malformados de gran tamaño para intentar causar un desbordamiento. La comparación de la salida del programa con resultados esperados es útil: si el programa tiene una salida determinística esperada para una entrada válida, cualquier desviación o interrupción indica un problema. El servicio puede comparar la salida de cada prueba con un resultado esperado almacenado en la carpeta de test; una discordancia o excepción sumará evidencia de fallo.

Gracias a Docker, este proceso es reproducible y portátil. Se puede desplegar el contenedor en distintas máquinas asegurando que el entorno (sistema operativo, herramientas, versiones de librerías) permanece constante, evitando falsos resultados por diferencias de entorno. Además, el aislamiento previene que, si el archivo analizado resulta ser malware ejecutable, cause daño al host real: cualquier actividad maliciosa queda contenida en el Docker, que se puede detener y restablecer fácilmente.

Postman, por su parte, se utilizó durante el desarrollo para iterar rápidamente en las pruebas, ajustando las peticiones y observando las respuestas del sistema de detección. En un escenario de producción, el mismo endpoint podría integrarse en una herramienta de seguridad automatizada o pipeline de CI/CD, donde los archivos compilados de un proyecto sean analizados en busca de vulnerabilidades antes de ser desplegados.

VI-G. Ejemplo de detección y validación cualitativa

Esta sección valida el enfoque en dos planos complementarios: (i) una evaluación cuantitativa del rendimiento del criterio de similitud con umbral y (ii) una validación cualitativa, a nivel de binario ejecutable, mediante una batería de pruebas dinámicas en contenedor.

VI-H. Evaluación cuantitativa del umbral de similitud

El sistema calcula una similitud $Sim(x, y) \in [0, 1]$ entre pares de muestras y aplica un umbral τ para decidir si una muestra desconocida se asocia a una familia conocida. Para cuantificar el impacto de τ, se evaluaron métricas estándar bajo cuatro valores (0.40, 0.50, 0.60 y 0.70): tasa de verdaderos positivos (TPR), tasa de falsos positivos (FPR), precisión y F_1.

Los resultados se recogen en la Tabla I. El mejor compromiso se obtiene con $\tau = 0{,}50$: TPR=95.1 %, FPR=5.4 %, precisión=95.8 % y F_1=95.4 %. Umbrales más permisivos (p. ej., 0.40) incrementan la sensibilidad (TPR=98.5 %) a costa de falsos positivos (FPR=15.2 %), mientras que umbrales más estrictos (p. ej., 0.70) reducen FPR (0.8 %) pero penalizan la detección (TPR=80.4 %). Estos datos justifican $\tau = 0{,}50$ como umbral operativo por defecto.

VI-I. Validación cualitativa mediante análisis estático y pruebas dinámicas

Además del criterio de similitud, el sistema emite un nivel de riesgo mediante inferencia difusa a partir de evidencias (estáticas y dinámicas). La fase dinámica se implementa como una batería de ejecuciones controladas ("fuzzing" de tipo caja negra a nivel de entradas), orientada a provocar condiciones límite: entradas sobredimensionadas, valores extremos y cargas que estresan rutas de copia/parseo. El objetivo no es generar un *proof-of-concept* de explotación, sino observar inestabilidad de ejecución (p. ej., fallos de segmentación) como indicador operativo.

VI-I1. Caso vulnerable: Se preparó un programa en C intencionalmente vulnerable con un búfer de 64 bytes y lectura con `gets()`, función intrínsecamente insegura por no acotar el tamaño de entrada. Tras compilar el programa a binario, se analizó con el sistema. En la fase estática, se detectó la llamada a `gets()` (indicador de API insegura) y se observó ausencia de canarios de pila (protecciones bajas). En la fase dinámica, al ejecutar el binario dentro de Docker con una entrada de 100 caracteres, el proceso terminó en un fallo de segmentación, confirmando comportamiento inestable. La combinación de evidencias (API insegura + ausencia de mitigaciones + crash) activó las reglas difusas hacia un RiesgoMalware alto, devolviendo un informe JSON con alerta de posible buffer overflow y el desglose de evidencias.

VI-I2. Caso corregido: Como contrapartida, se analizó un programa equivalente pero corregido, que utiliza `fgets()` con un límite de 63 caracteres para el mismo tamaño de búfer. En este segundo caso, el análisis estático no encontró funciones inseguras (categoría "ninguna"), y las ejecuciones de prueba no produjeron fallos ni comportamientos anómalos (ejecución estable). Aunque el binario continuaba sin canarios, el sistema difuso asignó *RiesgoMalware* bajo al no existir señales convergentes de vulnerabilidad.

VI-I3. Interpretación: Estos ensayos muestran dos propiedades relevantes: (i) la inferencia difusa prioriza la correlación de evidencias (no se apoya en una única señal) y (ii) la fase dinámica actúa como mecanismo de confirmación: cuando existe un *crash* reproducible bajo entradas límite, el riesgo se incrementa de forma robusta. En escenarios con información parcial (p. ej., sin ejecución dinámica), el sistema sigue proporcionando una estimación basada en reglas, pero con menor contundencia (riesgo medio/bajo dependiendo del resto de señales).

VI-J. Limitaciones experimentales

El rendimiento observado depende (a) de la cobertura de la batería de pruebas dinámicas y (b) de la calidad/representatividad de las reglas y funciones de pertenencia. En particular, los buffer overflows que no se manifiestan como fallo de ejecución bajo entradas simples (p. ej., corrupción silenciosa o sobreescrituras no inmediatas) requieren ampliar el conjunto de pruebas y/o enriquecer las evidencias con instrumentación adicional (p. ej., sanitizers, trazas, o análisis dinámico con mayor observabilidad).

VI-K. Discusión

Los resultados cuantitativos confirman que el umbral de similitud es un parámetro crítico: $\tau = 0{,}50$ maximiza el equilibrio entre TPR (95.1 %) y FPR (5.4 %), con precisión y F_1 superiores al 95 % (Tabla I). En paralelo, la validación cualitativa muestra que la decisión final es más fiable cuando la similitud se combina con evidencias ortogonales (APIs inseguras, ausencia de mitigaciones y comportamiento en ejecución).

Desde el punto de vista metodológico, la lógica difusa se utiliza para transformar señales continuas y parcialmente ambiguas (similitud, presencia de indicadores, estabilidad en ejecución) en una evaluación gradual de riesgo, evitando fronteras rígidas en casos límite. La fase dinámica descrita en la Subsección de validación cualitativa (pruebas dinámicas en contenedor) se integra en la experimentación como una batería de pruebas tipo fuzzing orientada a condiciones límite: su salida (crash/no crash, y estabilidad observada) alimenta directamente las reglas difusas y explica la escalada a riesgo alto en el caso vulnerable.

Respecto al machine learning, el sistema integra un módulo de clasificación que complementa el criterio de similitud con umbral e inferencia difusa. En particular, se emplean clasificadores supervisados de referencia (p. ej., decision trees y SVM) y modelos profundos (p. ej., autoencoders y arquitecturas generativas tipo GAN) para aprender patrones discriminativos sobre representaciones numéricas extraídas de los volcados de memoria/binarios. La capa difusa (tipo Mamdani) fusiona (i) la similitud por pares, (ii) la salida del clasificador (probabilidad/puntuación de malignidad) y (iii) evidencias estáticas/dinámicas, produciendo un nivel de riesgo interpretable. Adicionalmente, se realiza evaluación adversaria para medir robustez del detector ante muestras perturbadas, lo que permite identificar debilidades y guiar refuerzos como *adversarial training* o esquemas en ensemble.

VII. Pruebas en Otros Tipos de Malware

Para evaluar la generalidad del enfoque, se aplicó el mismo procedimiento a binarios asociados a otras clases de vulnerabilidad (interpretadas como familias/tipos). La Tabla III recoge los umbrales y el número de fallos observados en cada caso:

Tabla III
Resultados en otros tipos (familias) evaluados.

Tipo CWE	Umbral	Fallos
CWE369 (Divide by Zero)	0.09	0
CWE124 (Buffer Underwrite)	0.13	1
CWE252 (Unchecked Return Value)	0.00	0
CWE134 (Uncontrolled Format String)	0.14	0
CWE194 (Unexpected Sign Extension)	0.14	0

El único fallo se observó en CWE124. Es importante matizar que estos "fallos" no deben interpretarse automáticamente como errores reales de detección: al comparar similitud entre binarios de tipos distintos, pueden existir rasgos estructurales compartidos (p. ej., rutinas de manejo de cadenas o plantillas de código) que induzcan asociación espuria, especialmente con umbrales bajos. En consecuencia, estos resultados respaldan dos conclusiones prácticas: (i) el método es aplicable a múltiples tipos sin degradación general (0 fallos en 4 de 5 categorías) y (ii) conviene complementar la similitud con evidencias adicionales (estáticas/dinámicas) para desambiguar casos limítrofes.

VIII. Conclusiones

En este trabajo se ha evaluado un enfoque de detección basado en (1) análisis de similitud entre muestras y (2) decisión por umbral, complementado por (3) inferencia difusa para producir una estimación gradual del riesgo e integrar evidencias heterogéneas. Los resultados cuantitativos muestran que el umbral $\tau = 0{,}50$ ofrece el mejor equilibrio operativo, con TPR=95.1 %, FPR=5.4 %, precisión=95.8 % y F_1=95.4 % (Tabla I). Estos valores evidencian que la similitud constituye un criterio discriminativo eficaz cuando se calibra adecuadamente. La validación cualitativa mediante binarios de ejemplo confirma el valor de combinar análisis estático con pruebas dinámicas en contenedor. En particular, la

detección de APIs inseguras, la ausencia de mitigaciones y la observación de inestabilidad en ejecución bajo entradas límite se refuerzan mutuamente: la inferencia difusa prioriza la convergencia de señales, reduciendo decisiones frágiles basadas en un único indicador. Asimismo, las pruebas en otras familias/tipos (Tabla III) sugieren que el enfoque es transferible a distintas clases de vulnerabilidad, aunque aparecen casos aislados de confusión (CWE124), lo que refuerza la necesidad de enriquecer evidencias o elevar criterios cuando el solapamiento estructural entre tipos sea significativo. Entre líneas de mejora inmediatas se incluyen: ampliar la batería dinámica (fuzzing con mayor cobertura), instrumentar la ejecución para detectar corrupción silenciosa y explorar estrategias de umbral adaptativo dependientes del contexto y la densidad de similitud observada.

APÉNDICE

Este anexo resume (i) las variables empleadas en la inferencia difusa, (ii) un conjunto de reglas tipo y (iii) el alcance práctico del sistema para *buffer overflow*.

A. Variables de entrada/salida

Entradas (ejemplos):

- **Similitud** (Sim): valor normalizado en $[0, 1]$ (o en %).
- **Uso de APIs inseguras**: indicador categórico (Ninguna / Algunas / Muchas) a partir de coincidencias con funciones como `gets`, `strcpy`, `strcat`, `sprintf`, etc.
- **Mitigaciones de compilación**: indicador categórico (Bajas / Medias / Altas), p. ej., presencia/ausencia de canarios de pila.
- **Estabilidad en ejecución**: indicador binario o ternario (Estable / Inestable / No evaluado) según resultado de la batería de pruebas dinámicas.

Salida:

- **RiesgoMalware**: etiqueta (Bajo / Medio / Alto) o valor defuzzificado equivalente.

B. Reglas difusas tipo (plantillas)

A modo de referencia, reglas representativas (no exhaustivas) incluyen:

- **R1:** Si (Sim es *alto*) entonces (RiesgoMalware es *alto*).
- **R2:** Si (APIs inseguras son *muchas*) y (Mitigaciones son *bajas*) entonces (RiesgoMalware es *alto*).
- **R3:** Si (Ejecución es *inestable*) y (APIs inseguras no son *ninguna*) entonces (RiesgoMalware es *alto*).
- **R4:** Si (Sim es *medio*) y (APIs inseguras son *algunas*) entonces (RiesgoMalware es *medio*).
- **R5:** Si (Sim es *bajo*) y (APIs inseguras son *ninguna*) y (Ejecución es *estable*) entonces (RiesgoMalware es *bajo*).

C. Alcance real de detección para buffer overflow

El sistema es especialmente eficaz para:

- **Overflows inducidos por APIs inseguras** (copias/lecturas sin acotación) cuando están presentes en el binario y/o se manifiestan bajo entradas límite.
- **Casos con inestabilidad reproducible** (p. ej., fallo de segmentación) bajo la batería dinámica.

Limitaciones típicas (que motivan trabajo futuro):

- **Corrupción silenciosa** sin fallo observable con entradas simples.
- **Dependencia de rutas específicas** que requieren fuzzing más profundo o instrumentación (p. ej., sanitizers, cobertura).
- **Solapamiento estructural entre tipos**, donde la similitud por sí sola puede inducir asociaciones espurias sin evidencias adicionales.

AGRADECIMIENTOS

Este trabajo se ha realizado en el marco de los fondos del Plan de Recuperación, Transformacionales y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra "Ciberseguridad para la Innovación y la Protección y la Protección Digital" (Cátedra INCIBE-UCM). Asimismo, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, Proyecto CIRMA-CM (TEC-2024/COM-404).

REFERENCIAS

[1] AV-ATLAS, "About Malware and PUA," Online: https://portal.av-atlas.org/malware, accessed 5 April 2023.
[2] I. U. Haq and J. Caballero, "A Survey of Binary Code Similarity," *ACM Computing Surveys*, vol. 54, no. 2, pp. 1–38, 2021.
[3] E. Carrera and G. Erdélyi, "Digital genome mapping: advanced binary malware analysis," in *Proc. 14th Virus Bulletin Conf.*, 2004, pp. 187–197.
[4] M. Gheorghescu, "An automated virus classification system," in *Proc. Virus Bulletin Conf.*, 2005, pp. 294–300.
[5] X. Hu, T. Chiueh, and K. G. Shin, "Large-scale malware indexing using function-call graphs," in *Proc. ACM CCS*, 2009, pp. 611–620.
[6] S. M. Tabish, M. Z. Shafiq, and M. Farooq, "Malware detection using statistical analysis of byte-level file content," in *Proc. ACM SIGKDD Workshop on Cyber Security and Intelligence Informatics*, 2009, pp. 23–31.
[7] J. Kornblum, "Identifying almost identical files using context triggered piecewise hashing," *Digital Investigation*, vol. 3, pp. 91–97, 2006.
[8] J. Jang, A. Avgerinos, and D. Brumley, "BitShred: feature hashing malware for scalable triage and semantic analysis," in *Proc. 18th ACM CCS*, 2011, pp. 109–120.
[9] M. Botacin, V. H. G. Moia, F. Ceschin, M. A. A. Henriques, and A. Grégio, "Understanding uses and misuses of similarity hashing functions for malware detection and family clustering," preprint, 2023.
[10] "Juliet Test Suite v1.2 for C/C++ - User Guide," Department of Homeland Security, 2012.
[11] C. A. Brust, T. Sonnekalb, and B. Gruner, "ROMEO: A binary vulnerability detection dataset for exploring Juliet through the lens of assembly language," *Computers & Security*, vol. 128, p. 103165, 2023.
[12] C. Tamás, D. Papp, and L. Buttyán, "SIMBIoTA: Similarity-based malware detection on IoT devices," in *Proc. 6th Int. Conf. on Internet of Things, Big Data and Security (IoTBDS)*, 2021, pp. 58–69.
[13] K. Wang *et al.*, "ModDiff: Modularity similarity-based malware homologation detection," *Electronics*, vol. 12, no. 10, p. 2258, 2023.
[14] B. A. Kuperman, C. E. Brodley, H. Ozdoganoglu, T. N. Vijaykumar, and A. Jalote, "Detection and Prevention of Stack Buffer Overflow Attacks," *Communications of the ACM*, vol. 48, no. 11, pp. 51–56, Nov. 2005.
[15] M. A. Butt, Z. Ajmal, Z. I. Khan, and Y. Javed, "An In-Depth Survey of Bypassing Buffer Overflow Mitigation Techniques," 2022.
[16] H. Shahriar and H. M. Haddad, "Rule-Based Source Level Patching of Buffer Overflow Vulnerabilities," in *Proc. 10th Int. Conf. Information Technology: New Generations (ITNG)*, 2013, pp. 386–391.
[17] J. Duraes and H. Madeira, "A Methodology for Automated Identification of Buffer Overflow Vulnerabilities in Executable Software," in *Proc. 2005 DSN Workshops (Lecture Notes in Computer Science vol. 3746)*, 2005, pp. 297–300.
[18] İ. Atacak, "An Ensemble Approach Based on Fuzzy Logic Using Machine Learning Classifiers for Android Malware Detection," *Applied Sciences*, vol. 13, no. 3, p. 1484, 2023.
[19] A. Almutairi, M. A. Hossain, and S. Alshomrani, "Detecting Cross-Site Scripting in Web Applications Using Fuzzy Logic," *Security and Communication Networks*, vol. 2018, Article ID 8159548, 12 pages, 2018.
[20] L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, no. 3, pp. 338–353, 1965.

[21] X. Xu, S. Chen, and M. Xie, “A fuzzy logic-based approach for intrusion detection in wireless sensor networks,” IEEE Wireless Communications, vol. 17, no. 4, pp. 30–37, 2010.
[22] A. Singh and R. Kumar, “A hybrid fuzzy logic approach for malware detection,” in Proc. IEEE International Conference on Cyber Security and Cloud Computing, 2012, pp. 78–83.
[23] Y. Chen, M. Farooq, and L. Liu, “Cross-layer analysis for malware detection using fuzzy clustering,” IEEE Transactions on Information Forensics and Security, vol. 9, no. 2, pp. 250–261, 2014.
[24] M. Alazab, M. Awad, and E. M. H. M. Elkareem, “A fuzzy clustering-based approach for detection of botnets and malware traffic,” in Proc. IEEE International Conference on Communications, 2015, pp. 1398–1403.
[25] K. Wang and X. Gu, “Integrating fuzzy logic with deep learning for malware detection,” IEEE Access, vol. 4, pp. 2783–2791, 2016.
[26] J. Park, H. Lee, and S. Park, “A similarity-based approach for detecting malware families using fuzzy clustering,” IEEE Transactions on Dependable and Secure Computing, vol. 14, no. 1, pp. 12–24, 2017.
[27] C. Bishop and D. Dasgupta, “A survey of fuzzy logic applications in cybersecurity,” ACM Computing Surveys, vol. 50, no. 3, Article 45, 2018.
[28] A. Rana, P. Verma, and S. Sharma, “Cross-reference based malware detection using fuzzy rule-based systems,” in Proc. IEEE Conference on Dependable Systems and Networks, 2019, pp. 102–109.
[29] L. Martinez and F. Gomez, “Fuzzy machine learning for anomaly detection in malware: A case study,” IEEE Access, vol. 8, pp. 112345–112356, 2020.

Generación mejorada de ruido para oráculos de descifrado de la cifra Kyber

Francesc Sebé
Departament de Matemàtica. Universitat de Lleida
C. Jaume II, 69,
E-25001 Lleida (Spain)
francesc.sebe@udl.cat

***Resumen*—La cifra Kyber es un estándar NIST de cifrado de clave pública cuya seguridad depende de la supuesta intratabilidad del problema M-LWE (module learning with errors). En este artículo estudiamos el uso de Kyber en un entorno donde el poseedor de la clave privada actúa como un oráculo que colabora en el descifrado de textos cifrados que le proporcionan otras partes con el requisito de que el oráculo no debe obtener información sobre los textos en claro resultantes. Veremos que, en un escenario como el descrito, existe un riesgo de que se revele información que permite la reconstrucción de la clave privada. Esto obliga a limitar el número de veces que el oráculo puede utilizar una determinada clave. El artículo propone una solución para incrementar el número máximo de usos de cada clave.**

***Index Terms*—cifra Kyber, computador cuántico, oráculo de descifrado**

I. Introducción

La posibilidad de que en un futuro se consiga construir un computador cuántico criptográficamente relevante obliga a tomar medidas para reducir las consecuencias en lo que se refiere a la seguridad de los sistemas de cifrado. Existen fuentes [1] que afirman que existe un riesgo significativo de que dicho computador cuántico pueda construirse en un plazo de entre 10 y 16 años.

En criptografía de clave compartida y funciones hash, la ventaja para un atacante aportada por un computador cuántico se puede compensar doblando la longitud de las claves y de los resúmenes, respectivamente [2]. En cuanto a la criptografía de clave pública, la situación es totalmente distinta. Los algoritmos de Shor [3] permiten resolver, en un computador cuántico, tanto el problema de logaritmo discreto como el de la factorización entera en tiempo polinomial. Esto implica que cifras y sistemas de firma digital como ElGamal [4], RSA [5] y DSA [6] dejarán de ser seguros. La comunidad investigadora lleva años estudiando la construcción de criptosistemas de clave pública cuya seguridad depende de problemas matemáticos que se cree que son intratables incluso en un entorno de computación cuántica.

Los retículos [7] han demostrado ser una fuente de problemas intratables, incluso en entornos de computación cuántica, que sirven de base para la construcción de criptosistemas de clave pública tanto para cifrado como para firma digital. Por ejemplo, una variante del problema SIS [8] (small integer solution) es la base del sistema de firma digital Falcon [9] mientras que una variante del problema LWE [10] (learning with errors) es la base del sistema de cifrado Kyber [11] y del sistema de firma digital Dilithium [12].

II. La cifra Kyber

Kyber [11] es un KEM (key encapsulation mechanism) diseñado para ser resistente a ataques cuánticos. Su seguridad se basa en la supuesta intratabilidad del problema M-LWE (module-LWE).

II-A. Fundamentos y notación

Sea R_q el anillo de polinomios truncados definido como $\mathbb{Z}_q[X]/(X^n+1)$ cuyos elementos son de la forma

$$v = \sum_{i=0}^{n-1} v_i X^i, \quad v_i \in \mathbb{Z}_q. \tag{1}$$

En la especificación de Kyber, se ha tomado $q = 3329$ y $n = 256$. Estos parámetros cumplen que n es potencia de dos y, además, n divide a $q-1$, de tal modo que es posible aplicar técnicas basadas en la NTT [13] (number theoretic transform) para acelerar los cálculos.

En Kyber se utiliza el concepto de elementos *pequeños* de R_q cuyos coeficientes v_i (véase *Ec. (1)*) han sido tomados al azar considerando una distribución binomial centrada CBD_η. Es posible generar números enteros aleatorios distribuidos acorde a una distribución CBD_η, con espacio mostral $\{-\eta, \ldots, \eta\}$, mediante el *Algoritmo 1*. En la especificación de Kyber, se toma $\eta = 2$ o 3 según el caso.

Algorithm 1 Generación aleatoria de elementos distribuidos acorde a una distribución CBD_η

Entrada: $\eta \geq 1$ ▷ Parámetro de la distribución
Salida: $r \approx \mathrm{CBD}_\eta$
$r \leftarrow 0$
for $j = 0, \ldots, \eta - 1$ **do**
  $b_j \leftarrow_R \{0,1\}$
  $b'_j \leftarrow_R \{0,1\}$
  $r \leftarrow r + b_j - b_j$
end for
return r

En este artículo utilizaremos χ_η para referirnos a aquellos elementos de R_q cuyos coeficientes han sido generados al azar a partir de una distribución CBD_η.

Finalmente, denotaremos como $\mathbf{A}$ a una matriz aleatoria $\mathbf{A} \in R_q^{k \times k}$. En la especificación de Kyber se ha tomado $k = 2$ en la parametrización `Kyber512`, $k = 3$ en `Kyber768`, y $k = 4$ en `Kyber1024`.

II-B. Generación de claves

En Kyber, una clave privada se genera escogiendo, al azar, un vector (columna) $\mathbf{s} \in \chi_{\eta_1}^k$ ($\eta_1 = 3$ en `Kyber512`; $\eta_1 = 2$ para `Kyber768` y `Kyber1024`). Para generar la correspondiente clave pública, tomando una matriz conocida $\mathbf{A} \in R_q^{k \times k}$, se genera $\mathbf{e_s} \in \chi_{\eta_1}^k$ y se calcula el vector $\mathbf{t} \in R_q^k$, mediante el siguiente cálculo,

$$\mathbf{t} = \mathbf{A}\mathbf{s} + \mathbf{e_s}. \tag{2}$$

La clave privada es $\mathbf{s}$ mientras que la clave pública es el par $(\mathbf{A}, \mathbf{t})$. No es necesario almacenar $\mathbf{e_s}$.

Dada una clave pública $(\mathbf{A}, \mathbf{t})$, hallar la clave privada $\mathbf{s}$ implica resolver el sistema mostrado en *Ec. (2)* donde los valores $\mathbf{s}, \mathbf{e_s} \in \chi_{\eta_1}^k$ son desconocidos. Esto implica la resolución de un instancia del problema M-LWE que, bajo una correcta elección de los parámetros, es intratable.

II-C. Proceso de cifrado

En Kyber, un texto en claro es un vector $(m[0], \ldots, m[n-1]) \in \{0,1\}^n$. Durante el proceso de cifrado, los bits del texto en claro se utilizan para generar un elemento $m \in R_q$ definido como,

$$m = \sum_{i=0}^{n-1} m[i] \left\lfloor \frac{q}{2} \right\rceil X^i.$$

De este modo, en m, el coeficiente de X^i es 0 si $m[i] = 0$ y es $\left\lfloor \frac{q}{2} \right\rceil$ cuando $m[i] = 1$.

El texto cifrado bajo una clave pública $(\mathbf{A}, \mathbf{t})$ consiste en una tupla $(\mathbf{u}, v) \in R_q^k \times R_q$ que se construye tomando al azar $\mathbf{r} \in \chi_{\eta_1}^k$, $\mathbf{e_1} \in \chi_{\eta_2}^k$ ($\eta_2 = 2$ en todas las parametrizaciones), $e_2 \in \chi_{\eta_2}$, y luego calculando,

$$\begin{cases} \mathbf{u}^T &= \mathbf{r}^T\mathbf{A} + \mathbf{e_1}^T, \\ v &= \mathbf{r}^T\mathbf{t} + e_2 + m. \end{cases}$$

II-D. Proceso de descifrado

Dado un texto cifrado $(\mathbf{u}, v)$, el poseedor de la clave privada $\mathbf{s}$ puede descifrarlo calculando primero,

$$\tilde{m} = v - \mathbf{u}^T\mathbf{s}. \tag{3}$$

Luego, dado,

$$\tilde{m} = \sum_{i=0}^{n-1} \tilde{m}_i X^i,$$

se define,

$$\hat{m}[i] = \begin{cases} 1 & \text{si} \quad q/4 < \tilde{m}_i < 3q/4, \\ 0 & \text{en caso contrario.} \end{cases} \tag{4}$$

El texto en claro resultante es el vector $(\hat{m}[0], \ldots, \hat{m}[n-1])$.

Dado que el valor de $\tilde{m}$ calculado en la *Ec. (3)* es

$$\begin{aligned} \tilde{m} = v - \mathbf{u}^T\mathbf{s} = \mathbf{r}^T\mathbf{t} + e_2 + m - (\mathbf{r}^T\mathbf{A} + \mathbf{e_1}^T)\mathbf{s} = \\ = \mathbf{r}^T(\mathbf{A}\mathbf{s} + \mathbf{e_s}) + e_2 + m - \mathbf{r}^T\mathbf{A}\mathbf{s} - \mathbf{e_1}^T\mathbf{s} = \\ = m + \mathbf{r}^T\mathbf{e_s} + e_2 - \mathbf{e_1}^T\mathbf{s}, \end{aligned}$$

el texto descifrado coincidirá con el texto en claro original siempre que los coeficientes de

$$(\mathbf{r}^T\mathbf{e_s} + e_2 - \mathbf{e_1}^T\mathbf{s}) \in R_q \tag{5}$$

sean todos lo suficientemente pequeños.

III. Oráculo ciego de descifrado

Consideremos un escenario con dos participantes. Uno de ellos es un oráculo, $\mathcal{O}$, que posee una clave privada $\mathbf{s}$ del criptosistema Kyber. El otro participante, $\mathcal{P}$, posee un texto cifrado $(\mathbf{u}, v)$.

En este entorno, $\mathcal{P}$ desea obtener el texto en claro de $(\mathbf{u}, v)$ sin que el oráculo obtenga ninguna información (se mantenga ciego) sobre dicho texto en claro. Además, $\mathcal{O}$ quiere que su clave privada se mantenga secreta.

Dicha situación se daría en un entorno de voto electrónico donde $\mathcal{P}$ representa a la plataforma que se encarga de la recepción de los votos cifrados en Kyber. Una vez se han desvinculado los criptogramas recibidos de la identidad de los votantes, $\mathcal{P}$ solicita a la autoridad que custodia las claves, en este caso $\mathcal{O}$, que participe en el proceso descifrado. Si el contenido de la votación es confidencial, es preciso que $\mathcal{O}$ no pueda enterarse del resultado.

El requisito de que $\mathcal{O}$ se mantenga ciego elimina la solución trivial en la que $\mathcal{P}$ manda el texto cifrado a $\mathcal{O}$ y este devuelve el resultado de descifrarlo.

Bajo estas condiciones, la opción que estudiaremos es la siguiente:

1. $\mathcal{P}$ toma el texto cifrado $(\mathbf{u}, v)$ y manda $\mathbf{u}$ a $\mathcal{O}$.
2. $\mathcal{O}$ genera un valor aleatorio $e_d \in \chi_\eta$ y calcula $d = \mathbf{u}^T\mathbf{s} + e_d$.
3. $\mathcal{O}$ manda d a $\mathcal{P}$.
4. $\mathcal{P}$ calcula $\tilde{m} = v - d$ y emplea la función de colapso (*Ec. (4)*) para obtener el texto en claro.

La adición, por parte de $\mathcal{O}$, del ruido e_d al cálculo de d es necesaria ya que en caso contrario información sobre la clave privada $\mathbf{s}$ podría derivarse, por parte de $\mathcal{P}$, directamente de la respuesta sin ruido $d = \mathbf{u}^T\mathbf{s}$.

Por ejemplo, considerando que la clave privada es el vector $\mathbf{s} = (s[0], \ldots, s[k-1])^T$, $\mathcal{P}$ podría escoger $\mathbf{u} = (1, 0, \ldots, 0)^T$ como reto de tal modo que la respuesta obtenida sería $d = \mathbf{u}^T\mathbf{s} = s[0]$. Repitiendo el mismo proceso modificando $\mathbf{u}$ se podría obtener el resto de elementos del vector $\mathbf{s}$. Con la presencia del ruido e_d, obtener $\mathbf{s}$ a partir de varios d's obtenidos en distintintas ejecuciones implica la resolución del problema M-LWE.

III-A. Ataques de promedio

El proceso de descifrado que hemos descrito anteriormente protege la clave privada del oráculo a través de la adición del ruido e_d a cada respuesta d. A pesar de ello, la posibilidad de que $\mathcal{P}$ pueda obtener la respuesta d a cualquier reto $\mathbf{u}$ es fuente de riesgos.

En el caso de que cada ruido e_d se genere de manera independiente, $\mathcal{P}$ podría solicitar ℓ ejecuciones del proceso de descifrado proporcionando siempre el mismo valor de $\mathbf{u}$ como reto. Supongamos que $\mathcal{P}$ escoge $\mathbf{u} = (1, 0, \ldots, 0)^T$ como reto y sean $\{d_0, \ldots, d_{\ell-1}\}$ las respuestas recibidas ($d_j = s[0] + e_{d,j}$). Calculando el promedio de todas estas respuestas, después de interpretar los valores de $\mathbb{Z}_q$ como enteros en el rango $\left\{-\frac{(q-1)}{2}, \ldots, \frac{(q-1)}{2}\right\}$, se obtiene,

$$\frac{1}{\ell}\sum_{i=0}^{\ell-1} d_j = s[0] + e, \tag{6}$$

donde e es el promedio de $\{e_{d,0}, \ldots, e_{d,\ell-1}\}$ cuyos coeficientes (reales) se ajustan a una distribución gausiana de media cero y cuya desviación tiende a cero a medida que ℓ se incrementa. Por tanto, ℓ no puede tomar valores arbitrariamente grandes. Esto causa que el número de veces que el oráculo puede participar en el protocolo de descifrado ciego con una misma clave privada $\mathbf{s}$ esté limitado. A continuación estudiaremos el valor máximo que puede tomar ℓ una vez se ha fijado η.

Partiendo de que la media y la desviación de una variable aleatoria CBD_η son 0 y $\sqrt{\frac{\eta}{2}}$, respectivamente, el teorema central del límite nos dice que el promedio de ℓ muestras de dicha variable tiende a una distribución gaussiana $N\left(0, \sqrt{\frac{\eta}{2\ell}}\right)$.

Por otro lado, el estándar Kyber, a partir de su parametrización `Kyber768`, considera que el ruido proporcionado por una distribución CBD_η es de magnitud suficiente siempre que el parámetro η sea como mínimo 2. Por tanto, dicho ruido debe tener una desviación no inferior a 1 ($\sqrt{\frac{\eta}{2}}$ para $\eta = 2$).

Concluimos que la magnitud del ruido promedio e es suficiente siempre que se cumpla que $\sqrt{\frac{\eta}{2\ell}} \geq 1$, lo cual ocurre siempre que $\frac{\eta}{\ell} \geq 2$, o lo que es lo mismo, $\ell \leq \frac{\eta}{2}$. En la *Tabla I* se muestran los valores máximos permitidos para ℓ correspondientes a algunos valores de η.

Tabla I
NÚMERO MÁXIMO, ℓ, DE USOS DE UNA CLAVE PRIVADA CORRESPONDIENTES A DIFERENTES ELECCIONES DEL PARÁMETRO η, CUANDO EL RUIDO SE GENERA DE MANERA INDEPENDIENTE.

η	ℓ máx.
2	1
20	10
50	25

Dado que aumentar η incrementa la probabilidad de que el proceso de descifrado de Kyber falle (estamos añadiendo un término $-e_d \in \chi_\eta$ a la *Ec. (5)*), η no puede tomar valores arbitrariamente grandes con lo cual, el número de veces, ℓ, que se puede ejecutar el protocolo de descifrado ciego con una misma clave es muy reducido. Esto es un inconveniente. Es posible mantener una probabilidad pequeña de error en el descifrado después de aumentar η aumentando también el valor de q. De todos modos, la restricción $\ell \leq \frac{\eta}{2}$ es muy restrictiva. En la siguiente sección proponemos una manera alternativa de generar el ruido e_d que permite aumentar el número de usos de una clave.

Es preciso mencionar que, en caso de aumentar el valor de q, el valor tomado es deseable que cumpla la condición de que $q - 1$ sea divisible por n (véase la Sec. II-A) con el objetivo de se pueda seguir utilizando técnicas basadas en NTT en los cómputos.

IV. GENERACIÓN MEJORADA DE RUIDO

A continuación se propone un método alternativo para generar el ruido e_d que, dada una magnitud de ruido, aumenta significativamente el número máximo que puede tomar ℓ respecto al caso estudiado en la Sec. III-A.

En esta propuesta, el oráculo $\mathcal{O}$ escoge un entero $z \geq 1$, a partir del cual se establece el número máximo de usos de una clave privada en 2^z. También, fijado z, se cumple que $e_d \in \chi_{4z}$.

El método propuesto para la generación del ruido se describe en el *Algoritmo 2*. Se usan z variables estáticas $r_0, \ldots, r_{z-1} \in \chi_4$ cuyos valores se conservan entre las distintas llamadas al algoritmo. Cada una de estas variables almacena un elemento del anillo R_q con coeficientes pequeños generados a partir de una distribución aleatoria χ_4.

Algorithm 2 Generación mejorada de ruido

Salida: $r \approx \chi_{4z}$
static: $i \leftarrow 0$
static: $r_0, \ldots, r_{z-1} \in \chi_4$
if $i \geq 2^z$ **then**
  return $\perp$
end if
for $j = 0, \ldots, z-1$ **do**
  if $i \pmod{2^j} = 0$ **then**
    $r_j \leftarrow_R \chi_4$
  end if
end for
$i \leftarrow i + 1$
$r \leftarrow r_0 + \ldots + r_{z-1}$
return r

El algoritmo mantiene un entero i que indica que es la $(i+1)$-ésima vez que se le solicita un elemento de ruido al algoritmo. En la primera llamada, cuando $i = 0$, todos los elementos $r_0, \ldots, r_{z-1}$ son muestreados. A partir de entonces, r_0 será muestreado en cada llamada, r_1 será muestreado cada dos llamadas, r_2 cada cuatro, y así sucesivamente. Después de muestrear estas variables, el algoritmo retorna la suma $r = r_0 + \ldots + r_{z-1}$ que sigue una distribución χ_{4z}. El hecho de que el algoritmo almacene variables estáticas causa que los valores retornados no sean independientes entre sí.

IV-A. Desviación estándar de variables aleatorias CBD_η

Sean dos variables aleatorias independientes $X, Y \approx \mathrm{CBD}_4$. Sabemos que tienen una desviación estándar de $\sqrt{2}$ ($\sqrt{\frac{\eta}{2}}$ para $\eta = 4$). Si calculamos un promedio ponderado de ellas,

$$Z = pX + (1-p)Y \quad \text{con} \quad 0 \leq p \leq 1,$$

la variable resultante Z tiene una desviación de $\sqrt{2}\sqrt{p^2 + (1-p)^2}$. Dicha desviación toma el valor mínimo cuando $p = \frac{1}{2}$ siendo 1 este valor mínimo. En este caso, se obtiene exactamente la misma desviación que tiene una variable aleatoria CBD_2 y que se considera de magnitud suficiente para proporcionar seguridad a la cifra Kyber en sus parametrizaciones `Kyber768` y `Kyber1024`.

IV-B. Ataques de promedio al sistema mejorado

Supongamos que $\mathcal{P}$ quiere obtener información sobre la clave privada de $\mathcal{O}$. Para ello, manda de manera repetida el reto $\mathbf{u} = (1, 0, \ldots, 0)^T$ y almacena las ℓ respuestas recibidas $\{d_0, \ldots, d_{\ell-1}\}$, con $d_j = s[0] + e_{d,j}$, habiéndose generado cada elemento de ruido $e_{d,j}$ mediante el *Algoritmo 2*.

Vamos a denotar como $e^{(t)}$ el promedio de la suma parcial de las t primeras variables estáticas del *Algoritmo 2* ($r_0 + \ldots + r_{t-1}$) de $\{e_{d,0}, \ldots, e_{d,\ell-1}\}$.

Promediando las respuestas obtenidas, $\mathcal{P}$ obtiene una versión ruidosa de $s[0]$,

$$\frac{1}{\ell}\sum_{i=0}^{\ell-1} d_i = s[0] + e^{(z)}.$$

Dado que $\mathcal{O}$ no responderá a más de 2^z peticiones, el *Algoritmo 2* habrá generado, como máximo, dos valores distintos para la variable estática r_{z-1}. Llamemos $r_{z-1}^{(I)}$ y $r_{z-1}^{(II)}$ a estos valores. Entonces,

$$\frac{1}{\ell}\sum_{i=0}^{\ell-1} d_i = s[0] + e^{(z-1)} + p \cdot r_{z-1}^{(I)} + (1-p) \cdot r_{z-1}^{(II)},$$

con $0 \leq p \leq 1$. Dado que cada coeficiente de $r_{z-1}^{(I)}$ y de $r_{z-1}^{(II)}$ se ha generado aleatoriamente a partir de una distribución CBD_4, cada coeficiente de su promedio ponderado $p \cdot r_{z-1}^{(I)} + (1-p) \cdot r_{z-1}^{(II)}$ tiene una desviación de, como mínimo, la desviación de una variable CBD_2 (véase la Sec. IV-A). Por tanto, la versión ruidosa de $s[0]$ que obtiene $\mathcal{P}$ garantiza una complejidad del problema M-LWE como mínimo tan alta como la que se fija en el estándar de Kyber.

Hemos visto que $\mathcal{P}$ no puede obtener una versión *limpia* de $s[0]$. A pesar de ello, $\mathcal{P}$ podría intentar obtener dos valores *limpios* $u_1[0]s[0] + r_{z-1}^{(I)}$ y $u_2[0]s[0] + r_{z-1}^{(I)}$ y obtener $s[0]$ mediante la resolución de un sistema de ecuaciones 2×2. Para ello debería asegurarse de mandar sus peticiones dentro de las primeras 2^{z-1} peticiones atendidas por parte de $\mathcal{O}$ ya que en ellas, la variable estática r_{z-1} se sabe que toma un valor constante (aunque desconocido) que denotamos $r_{z-1}^{(I)}$.

Para obtener una versión limpia de $u_1[0]s[0] + r_{z-1}^{(I)}$, $\mathcal{P}$ mandaría ℓ peticiones a $\mathcal{O}$ cuyo promedio daría como resultado,

$$\frac{1}{\ell}\sum_{i=0}^{\ell-1} d_i = u_1[0]s[0] + r_{z-1}^{(I)} + e^{(z-1)}.$$

Dado que estas peticiones se han mandado dentro de las primeras 2^{z-1} peticiones recibidas por $\mathcal{O}$, el *Algoritmo 2* habrá generado, como máximo, dos valores distintos para la variable estática r_{z-2}. Siguiendo el mismo razonamiento anterior, el valor obtenido de $u_1[0]s[0] + r_{z-1}^{(I)}$ irá acompañado de un ruido cuyos coeficientes tienen una magnitud, como mínimo, a la correspondiente a una variable CBD_2.

Aplicando este razonamiento de manera recursiva, concluimos que ningún valor de la forma $u_1[0]s[0] + r_{z-1}^{(I)} + r_{z-2}^{(I)} + \ldots$ puede obtenerse sin la presencia de un ruido con coeficientes con una desviación inferior al de una variable CBD_2.

IV-C. Comparación

En la *Tabla II* se muestra, para distintos valores del parámetro z, el parámetro $\eta = 4z$ que define la magnitud del ruido generado por el *Algoritmo 2* en el sentido que los elementos de R_q que retorna tienen sus coeficientes distribuidos acorde a un modelo CBD_η. Dado este valor η, luego se muestra el número máximo de veces que se podría utilizar una misma clave si se utiliza el mecanismo trivial que retorna valores independientes entre sí y luego el número máximo de usos si se utiliza el mecanismo mejorado que se propone en este trabajo. La diferencia entre las dos opciones es muy clara. Dado η, la propuesta trivial permite un número proporcional de usos, mientras que la propuesta mejorada admite un número exponencial.

Tabla II
COMPARACIÓN ENTRE LA GENERACIÓN DE RUIDO ALEATORIA Y MEJORADA.

z	$\eta = 4z$	ℓ máx. (indep.)	ℓ máx. (mej.)
1	4	2	2
2	8	4	4
4	16	8	16
8	32	16	256
10	40	20	1024

Es preciso mencionar que el uso de la propuesta mejorada puede implicar algunas dificultades en un entorno con múltiples servidores actuando en la nube como un mismo oráculo. Dado que se debe mantener un estado, en un entorno distribuido es necesario que todos los servidores se mantengan sincronizados entre ellos. Esta problemática no se da en la propuesta trivial donde los valores de ruido se generan de manera aleatoria e independiente.

V. CONCLUSIONES

En este trabajo se ha estudiado el uso de la cifra Kyber en un entorno donde existe un oráculo que participa en el descifrado de textos proporcionados por otras partes. Hemos considerado un protocolo en el que el oráculo no tiene acceso a los textos en claro pero que implica un riesgo para la clave privada que limita el número de usos de esta.

Se ha estudiado el máximo número de veces que se puede utilizar una clave privada cuando la información dada por el oráculo se protege mediante ruido aleatorio independiente. Luego se ha propuesto y analizado una forma alternativa de generar este ruido que permite un número exponencial de usos de una misma clave.

AGRADECIMIENTOS

Este trabajo es parte del proyecto R&D+i PID2021-124613OB-I00 financiado por MICIU/ AEI/10.13039/501100011033 y FEDER, EU.

Esta investigación también resulta del Proyecto Estratégico 'Avances en criptografía post-cuántica aplicados al desarrollo de un sistema de cupones' (C039/24), fruto del convenio de colaboración suscrito entre el Instituto Nacional de Ciberseguridad (INCIBE) y la Universidad de Lleida. Esta iniciativa se realiza en el marco de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation).

REFERENCIAS

[1] EU PQC Workstream, "A coordinated implementation roadmap for the transition to post-quantum cryptography," European Commission, Tech. Rep., 2025. [Online]. Available: https://ec.europa.eu/newsroom/dae/redirection/document/117507

[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, ser. STOC '96. New York, NY, USA: Association for Computing Machinery, 1996, p. 212–219.

[3] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134.

[4] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," in *Advances in Cryptology*, G. R. Blakley and D. Chaum, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1985, pp. 10–18.

[5] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, p. 120–126, Feb. 1978.
[6] FIPS 186-5, "Digital signature standard (dss)," National Institute of Standards and Technology, Tech. Rep., 2023. [Online]. Available: https://doi.org/10.6028/NIST.FIPS.186-5
[7] S. D. Galbraith, *Mathematics of Public Key Cryptography*. Cambridge University Press, 2012.
[8] M. Ajtai, "Generating hard instances of lattice problems (extended abstract)," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, ser. STOC '96. New York, NY, USA: Association for Computing Machinery, 1996, p. 99–108.
[9] P.-A. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Prest, T. Ricosset, G. Seiler, W. Whyte, and Z. Zhang, "FALCON: Fast-fourier lattice-based compact signatures over NTRU," 2020. [Online]. Available: https://falcon-sign.info/
[10] V. Lyubashevsky, C. Peikert, and O. Regev, "On ideal lattices and learning with errors over rings," *J. ACM*, vol. 60, no. 6, Nov. 2013.
[11] R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé, "CRYSTALS-Kyber. algorithm specifications and supporting documentation (version 3.02)," 2021. [Online]. Available: https://pq-crystals.org/kyber/
[12] S. Bai, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé, "CRYSTALS-dilithium. algorithm specifications and supporting documentation (version 3.1)," 2021. [Online]. Available: https://pq-crystals.org/dilithium/
[13] H. J. Nussbaumer, *Number Theoretic Transforms*. Springer Berlin Heidelberg, 1982, pp. 211–240.

Metodología de Auditoría de Juguetes

Diego Narciandi-Rodríguez
Grupo de Invest. SECOMUCI
Universidad de León
dnarr@unileon.es

Jose Aveleira-Mata
Grupo de Invest. SECOMUCI
Universidad de León
jose.aveleira@unileon.es

Pablo de la Hera Martínez
Inst. Nacional de Ciberseguridad de España
Av. José Aguado, 41, León
pablo.hera@incibe.es

Iván Gómez Muñoz
Inst. de Invest. RIASC
Universidad de León
igomm@unileon.es

Jonatan García González
Inst. de Invest. RIASC
Universidad de León
jgago@unileon.es

Miguel Martín Redondo
Inst. Nacional de Ciberseguridad de España
Av. José Aguado, 41, León
miguel.martin@incibe.es

***Resumen*—El creciente uso de dispositivos IoT en entornos domésticos y educativos evidencia la necesidad de una metodología estandarizada de auditoría que permita evaluar su ciberseguridad siguiendo las directrices europeas ETSI EN 303 645 y ETSI TS 103 701. En este trabajo se propone una metodología basada en ocho vectores de análisis, que abarcan desde el estudio del firmware y los mecanismos de actualización hasta la aplicación móvil, el portal de configuración y las comunicaciones del dispositivo. La propuesta se aplica a un caso práctico de un juguete conectado anonimizado, donde se identifican diversas deficiencias representativas de los riesgos actuales en este tipo de productos. Este proyecto fue publicado oficialmente por INCIBE en 2024 dentro del informe divulgativo "Estudio de la ciberseguridad en juguetes conectados", elaborado durante la campaña de Navidad de 2024, con el objetivo de concienciar y establecer criterios homogéneos de evaluación para fabricantes y usuarios.**

***Index Terms*—IoT, Ciberseguridad, Auditoría, Privacidad, Juguetes**

I. Introducción

En los últimos años, la proliferación de dispositivos del Internet de las Cosas (IoT) ha transformado la forma en que interactuamos con el entorno digital y físico. Estos dispositivos, diseñados para operar con recursos computacionales limitados y conectividad permanente, manejan con frecuencia información sensible, como datos biométricos, geolocalización o imágenes en tiempo real, lo que los convierte en objetivos especialmente atractivos para los ciberdelincuentes. La creciente dependencia de estas tecnologías en toda clase de sectores (industrial, juguetero, ocio, domótica, etc.) ha incrementado la superficie de ataque y, en consecuencia, la frecuencia e impacto de los incidentes de ciberseguridad asociados a ellos [1], [2]. Buena parte de las vulnerabilidades observadas en dispositivos IoT no deriva de fallos complejos de software, sino de deficiencias básicas en su configuración o mantenimiento. La ausencia de actualizaciones de seguridad, la reutilización de credenciales por defecto, la apertura innecesaria de puertos o la omisión de configuraciones esenciales son factores que degradan la seguridad de estos sistemas [3], [4], [5]. Este escenario pone de manifiesto la necesidad de aplicar mecanismos de gestión y verificación estandarizados de seguridad efectivos a lo largo de todo el ciclo de vida del dispositivo [6]. Para dar respuesta a estos desafíos, el European Telecommunications Standards Institute (ETSI) desarrolló la Norma Europea EN 303 645, que establece los requisitos fundamentales de ciberseguridad para productos IoT de consumo dentro del mercado europeo [7]. Esta norma define trece categorías de requisitos y sesenta y ocho disposiciones, orientadas a garantizar la protección del usuario, la privacidad y la integridad del producto. Como complemento, el Documento Técnico ETSI TS 103 701 describe una metodología de evaluación que detalla los casos de prueba y criterios de conformidad asociados a la norma. Asimismo, otros marcos de referencia, como el OWASP IoT Security Verification Standard [8] o el IoT Security Compliance Framework de la IoT Security Foundation [9], han contribuido a consolidar un conjunto de buenas prácticas globales que buscan armonizar los procesos de auditoría y certificación en materia de seguridad IoT [10]. Dentro de este ecosistema, los juguetes conectados constituyen una categoría particularmente crítica. Se trata de dispositivos dirigidos al público infantil que, mediante tecnologías de comunicación inalámbrica, como Bluetooth, WiFi u otras, permiten la interacción remota, el control mediante aplicaciones móviles o el intercambio de datos personales del menor [11].

Estudios recientes han demostrado que un número significativo de juguetes IoT comercializados en España transmiten información sin aplicar mecanismos básicos de protección, incluso en modelos conectados a aplicaciones móviles populares [11]. Estas deficiencias, unidas al perfil de su público objetivo, convierten a este tipo de dispositivos en un ámbito de especial sensibilidad y relevancia para la ciberseguridad doméstica.

En este contexto, el presente trabajo propone una metodología de auditoría para juguetes IoT que adapta y estandariza, a nivel nacional, las directrices de la ETSI EN 303 645 y la ETSI TS 103 701. El objetivo es proporcionar un marco común de evaluación que permita analizar los servicios, procesos y mecanismos de seguridad implementados en este tipo de dispositivos de consumo, contribuyendo a su mejora continua y a la protección de los usuarios más vulnerables. La metodología propuesta se centra en las amenazas más frecuentes y realistas del entorno IoT, excluyendo del alcance los ataques altamente dirigidos o los análisis invasivos de hardware y firmware, con el fin de ofrecer una guía práctica y reproducible para auditores, fabricantes y organismos de certificación.

II. Trabajo relacionado

La estandarización de la ciberseguridad en IoT de consumo ha avanzado desde guías de buenas prácticas hacia requisitos verificables y esquemas de evaluación. En este contexto, ETSI EN 303 645 define un conjunto de requisitos base (credenciales, actualizaciones seguras y divulgación de vulnerabilidades) orientados a dispositivos IoT de consumo y sus servicios asociados. Para operativizar dichos requisitos en un proceso de auditoría, ETSI TS 103 701 propone una metodología de *conformance assessment* con criterios y casos de prueba, pensada para laboratorios, organismos certificadores o autoevaluaciones del fabricante [12]. En paralelo, otras referencias han definido capacidades y controles mínimos comparables, como el *core baseline* de NIST para dispositivos IoT [13] o el *Code of Practice* del Reino Unido, que ha servido de antecedente para iniciativas posteriores [14]. Además, marcos como el *IoT Security Assurance Framework* de IoTSF (con mapeos explícitos a ETSI y NIST) [9] y el *OWASP IoT Security Verification Standard (ISVS)* [15] proporcionan catálogos de requisitos y evidencias que facilitan la trazabilidad, aunque suelen ser genéricos y no siempre detallan una puesta en práctica reproducible para categorías específicas como juguetes conectados.

En el ámbito concreto de *smart toys*, gran parte de la literatura se ha articulado como estudios de caso o análisis de mercado que evidencian riesgos de seguridad, privacidad y cumplimiento, pero con enfoques, herramientas y alcances heterogéneos. Trabajos tempranos sistematizan problemáticas y amenazas en juguetes conectados, resaltando la necesidad de controles mínimos y de transparencia hacia familias y usuarios [16]. Otros estudios realizan análisis técnicos sobre juguetes infantiles IoT, identificando debilidades en comunicaciones, autenticación, exposición de datos y dependencias con servicios en la nube [17]. También se han propuesto requisitos específicos para *smart toys* como base para pruebas y verificación [18]. Más recientemente, evaluaciones centradas en el mercado europeo han reabierto la discusión sobre transparencia y tratamiento de datos en juguetes actuales [19].

En conjunto, el estado del arte muestra dos carencias: (i) la falta de procedimientos homogéneos que conecten de forma trazable requisitos normativos (p. ej., ETSI/NIST/OWASP/IoTSF) con evidencias técnicas obtenibles en laboratorio; y (ii) la limitada comparabilidad entre juguetes cuando los trabajos se apoyan en estudios de caso aislados o alcances dispares. En esta línea, surge la metodología propuesta en esta contribución para estandarizar la auditoría de este tipo de dispositivos con herramientas y pruebas reales y adaptadas a un alcance definido.

III. Objetivo de la metodología

El objeto de esta metodología es evaluar los servicios y procesos de los dispositivos IoT de consumo según la Norma Europea EN 303 645 del Instituto Europeo de Normas de Telecomunicaciones [7]. Esto aborda los 13 requerimientos y las 68 disposiciones, tanto obligatorias como recomendadas y condicionales, en las que se dividen mencionados requerimientos, definiendo casos de prueba y criterios de evaluación para cada disposición a partir de la especificación técnica ETSI TS 103 701 [12]. La presente metodología pretende contribuir a la protección de los productos IoT de consumo contra las amenazas de ciberseguridad más comunes, por lo que los ataques altamente dirigidos y/o sofisticados y, por lo tanto, el análisis invasivo de módulos de hardware y software, están fuera del alcance de la misma.

Los objetivos principales que se persiguen con la aplicación de esta metodología son:

- Identificación, recopilación de información técnica y estudio de los componentes, mecanismos de comunicación, servicios, puertos, mecanismos criptográficos, tratamiento de datos y telemetría, etc., más comunes y aplicables a dispositivos IoT de consumidor, a partir de la documentación pública y de aquella proporcionada por el fabricante.
- Identificación del método de evaluación propuesto por parte de ETSI mediante los documentos ETSI EN 303 645 / TS 103 701, alineando nuestra metodología de análisis de ciberseguridad con los requisitos y procesos definidos por estos estándares.
- Definición y configuración de escenarios (arquitecturas, componentes y herramientas) que permitan la aplicación de la presente metodología de evaluación, asegurando que dichos escenarios simulan condiciones y entornos de operación reales o lo más cercano posible a ello.
- Comprobación de las evidencias y suficiente y correcta información dada por el fabricante durante el previo proceso de consultoría, sobre el relleno de las tablas de identificación del POE, de la tabla ICS de disposiciones aplicables y las tablas IXIT de información sobre implementación y características del POE.
- Verificación de la conformidad con los requisitos de seguridad definidos en la especificación ETSI TS 103 701.
- Aplicación de procesos de divulgación de vulnerabilidades que los fabricantes han de llevar a cabo.
- evaluación continua y actualización de la metodología basada en nuevos descubrimientos y estándares emergentes; así como en la actualización propia de las normas y especificaciones en las que se basa la metodología.

IV. Alcance

Esta metodología da cumplimiento a los requisitos de la norma ISO/IEC 17025 [20]. Concretamente, aplicado a:

- Todo el personal involucrado en las operaciones del laboratorio, incluyendo tanto al personal técnico como al de gestión administrativa, asegurando que todos comprendan y cumplan con los procedimientos y políticas establecidos.
- El conjunto de los ensayos bajo el esquema ETSI EN 303 645 y ETSI TS 103 701 en lo relacionado a la comprobación de la seguridad de dispositivos IoT de consumidor.
- Las instalaciones del laboratorio, teniendo en cuenta la posibilidad de que los ensayos se tengan que realizar parcial o totalmente en las instalaciones del cliente o terceras partes, si bien se diera el caso.
- La gestión completa de equipos y materiales de ensayo, enfocándose en garantizar su correcta calibración, mantenimiento y trazabilidad para asegurar la precisión y validez de los resultados obtenidos.

- La gestión de toda la documentación y los registros necesarios para demostrar la conformidad con los requisitos de la norma ISO/IEC 17025 y de los esquemas aplicables. Esto incluye la creación, revisión, almacenamiento y acceso controlado a dichos documentos.

V. Métodos y Materiales

Para la correcta realización de las diferentes pruebas es necesario estandarizar la topología de la red del laboratorio, la cual se muestra en la Figura 1, así como sus diferentes componentes con el objetivo de poder proveer a otros investigadores e instituciones la capacidad de reproducir las pruebas y obtener los mismos resultados.

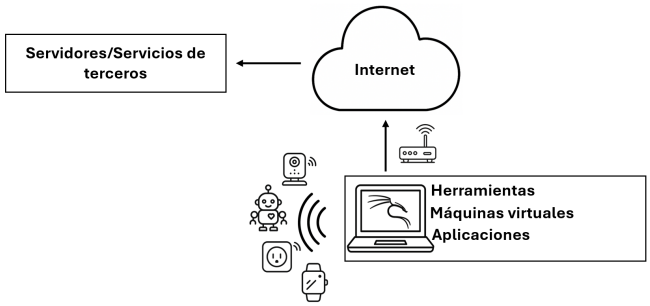


Figura 1. Arquitectura del laboratorio

Dentro de las diferentes herramientas a utilizar encontramos tanto soluciones comerciales como de código abierto, abarcando un amplio espectro de utilidades orientadas a la evaluación de la seguridad en distintos niveles. Entre las herramientas comerciales, las cuales se muestran en la Tabla I, destacan Tenable Nessus y Burp Suite Professional, centradas en el escaneo automatizado de vulnerabilidades en sistemas y aplicaciones web, respectivamente. En cuanto a las herramientas de código abierto presentes en la Tabla II, se incluyen alternativas para análisis y depuración de dispositivos móviles, auditoría de redes inalámbricas y radio, escaneo y descubrimiento de red, captura y análisis de tráfico, fuzzing y explotación, ataques o recuperación de contraseñas y análisis de firmware, contenedores o sistemas embebidos.

Tabla I
Herramientas comerciales

Herramienta	Uso principal
Tenable Nessus	Escáner automático de vulnerabilidades
Burp Suite Professional	Herramientas para pruebas de seguridad web y análisis de tráfico HTTP/HTTPS.

Finalmente, las utilidades y comandos UNIX (apt, ip, grep, ping, sudo, entre otros) proporcionan la base operativa necesaria para ejecutar, automatizar y supervisar las pruebas. En conjunto, esta selección ofrece un abanico de herramientas que permite ajustar cada prueba a los recursos y objetivos del laboratorio.

VI. Procedimiento de evaluación

Con el objetivo de analizar el nivel de ciberseguridad y privacidad de un dispositivo conectado dirigido a público infantil, se desarrolló una metodología centrada en la identificación de vulnerabilidades técnicas y en la verificación de su alineamiento con los requisitos regulatorios vigentes. La evaluación se organizó en vectores de prueba (V1–V8), cada uno de ellos con casos verificables y criterios de resultado clasificados como Favorable, Desfavorable o Inconcluyente. Esta estructura modular permitió una valoración exhaustiva del comportamiento del dispositivo desde distintas perspectivas: red, firmware, aplicación móvil, autenticación, interfaces físicas y aéreas, entre otros.

En una primera fase, se procedió a la identificación y

Tabla II
Herramientas open-source

Herramienta	Uso principal
ADB	Interacción con dispositivos Android.
Aircrack-ng	Auditoría y descifrado de redes Wi-Fi.
Bettercap	Análisis y ataques MiTM en redes.
Binwalk	Análisis y extracción de firmware.
Bluetoothctl	Gestión de dispositivos Bluetooth.
BlueZ	Pila y gestión de Bluetooth en Linux.
Bully	Recuperación de contraseñas WPS.
Clair	Escaneo de vulnerabilidades en imágenes Docker.
Docker Bench for Security	evaluación de seguridad en entornos Docker.
Ettercap	Ataques MiTM y sniffing de red.
Ffuz	Fuzzing de aplicaciones web.
Flashrom	Lectura y escritura de firmware flash.
FOCA	Análisis de metadatos en documentos.
Frida	Instrumentación dinámica y análisis de apps.
Genymotion	Emulación de Android.
GnuPG	Cifrado y gestión de claves OpenPGP.
Gobuster	Escaneo de directorios y subdominios.
GQRX	Análisis de señales de radio SDR.
Hashcat	Recuperación de contraseñas por GPU.
Hash-identifier	Identificación de algoritmos hash.
homePwn	evaluación de seguridad IoT.
Hping3	Generación y prueba de paquetes TCP/IP.
Hydra	Ataques de fuerza bruta a servicios.
inSSIDer	Análisis de señal y cobertura Wi-Fi.
JADX	Descompilación de APKs Android.
John the Ripper	Recuperación de contraseñas.
KillerBee	Análisis y ataques en redes ZigBee.
Masscan	Escaneo masivo de puertos.
Medusa	Fuerza bruta para múltiples protocolos.
Metasploit	Ejecución y desarrollo de exploits.
MobaXterm	Terminal remoto con utilidades de red.
MobSF	Análisis estático y dinámico de apps móviles.
Netdiscover	Detección de dispositivos en red.
NetSpot	Auditoría y planificación Wi-Fi.
Nmap	Escaneo de puertos y servicios.
Nmap NSE	Automatización de escaneos con scripts.
OpenOCD	Depuración de microcontroladores.
OpenSSH	Conexiones seguras SSH y transferencia.
OpenVAS	Escaneo de vulnerabilidades en red.
Peach	Fuzzing de software.
PixieWPS	Ataque offline a PINs WPS.
Postman	Pruebas y desarrollo de APIs.
PuTTY	Cliente SSH/Telnet.
Rainbowcrack	Recuperación de contraseñas por tablas arcoíris.
Reaver	Ataque WPS para obtener contraseñas WPA/WPA2.
SQLMap	Explotación de inyecciones SQL.
sslscan	Análisis de configuraciones SSL/TLS.
SSLStrip	Intercepción de tráfico HTTPS.
testssl.sh	Verificación de seguridad TLS/SSL.
USBDeview	Gestión de dispositivos USB.
VirtualBox	Virtualización de sistemas operativos.
Wfuz	Fuzzing de aplicaciones web.
Whispers	Detección de credenciales en código fuente.
WiGLE Wi-Fi	Mapeo y localización de redes inalámbricas.
Wireshark	Captura y análisis de tráfico de red.
Zenmap	Interfaz gráfica para Nmap.
ssh-audit	Auditoría de configuraciones SSH.
lynis	Auditoría y hardening de sistemas UNIX.

documentación del dispositivo, registrando datos como el modelo, número de serie, tipo de producto, edad recomendada, idiomas soportados y cualquier información adicional relevante. Asimismo, se realizaron fotografías del contenido del paquete y del propio dispositivo para documentar el análisis físico y visual.

Posteriormente, se recopilaron los datos técnicos y operativos relativos a la conectividad, interacción con el usuario, sensores integrados, puertos físicos, fuente de energía, versión del firmware y software asociado. Esta información permitió establecer el contexto funcional del dispositivo y determinar su superficie de exposición potencial.

De igual forma, se revisaron aspectos vinculados al tratamiento de datos personales y la privacidad, incluyendo el tipo de transmisión (local o en la nube), el almacenamiento de datos, la existencia de funciones de geolocalización, la presencia de controles parentales y de publicidad integrada, así como las certificaciones de conformidad y seguridad aplicables (por ejemplo, FCC Part 15 y marcado CE).

El dispositivo objeto de estudio cuenta con conectividad dual Wi-Fi (2.4/5 GHz) y Bluetooth 4.2, una serie de sensores ambientales y de movimiento (IMU de 6 DoF, cámara, micrófono, sensor ToF y de luz) y múltiples modalidades de interacción, entre ellas control por voz mediante asistentes virtuales y una aplicación móvil compatible con Android e iOS. Se ejecuta sobre una base Debian Linux 9 con ROS (melodic), y dispone de actualización automática de firmware por red, lo que amplía su funcionalidad pero también introduce potenciales vectores de riesgo si no se aplican medidas de control adecuadas.

Por último, se constató la ausencia de funciones de localización y de publicidad en la plataforma, así como la falta de especificación formal de mecanismos de control parental. El almacenamiento se realiza tanto en la nube como de forma local, lo que refuerza la necesidad de evaluar las políticas de gestión de datos y la seguridad en las comunicaciones con los servidores del fabricante.

Comenzando con el primer vector (**V1. Análisis de vulnerabilidades**), el objetivo es identificar y documentar todos los puertos accesibles del dispositivo en un entorno controlado. Para ello, se realiza un escaneo completo de puertos con el fin de determinar aquellos que se encontraban abiertos, escuchando conexiones entrantes o ya establecidas, así como los que permanecían cerrados. A partir de estos resultados, se procede al análisis de los servicios asociados, verificando sus versiones e identificando posibles vulnerabilidades o exploits conocidos relacionados con dichos servicios.

En la auditoría del dispositivo de prueba con el primer vector, durante el escaneo en red se detectaron un total de 41 puertos abiertos, entre los cuales destaca la presencia del servicio SSH (puerto 22) ejecutando OpenSSH 7.4p1 Debian, protegido con credenciales por defecto de dominio público. Asimismo, se encontraron servicios HTTP que aceptaban únicamente el método POST, así como puertos asociados a comunicaciones internas del entorno ROS. La ausencia de documentación que justifique la exposición del servicio SSH y la existencia de un usuario por defecto facilitan la elevación de privilegios. Los análisis automáticos con Nessus reportaron 38 vulnerabilidades críticas, 90 altas y 41 medias y 3 bajas, derivadas principalmente del uso de librerías y paquetes obsoletos. Por lo tanto el veredicto es: FALLA.

El segundo vector (**V2. Firmware**) se centra en la inspección del sistema operativo y del firmware del dispositivo. Se evalua la posible exposición de información sensible o credenciales recuperables tanto en el firmware como en los recursos accesibles desde la web del fabricante. Además, se verifica la existencia de cuentas ocultas o no documentadas, comprobando que todos los servicios se ejecutan con los privilegios mínimos necesarios. Este análisis incluye la revisión de permisos de procesos, archivos, directorios y contenedores.

En el segundo vector, se constató que el firmware viaja comprimido en un archivo ZIP sin protección criptográfica, incluyendo un paquete DEB y una imagen del sistema. Su extracción permite acceder a los componentes internos del sistema operativo, scripts y librerías, lo que posibilita un acceso total a los contenidos. Durante el análisis se detectó la existencia de dos cuentas de usuario: root y linaro, siendo esta última de uso por defecto y fácilmente comprometible mediante herramientas de cracking. Además, se verificó que los servicios y procesos no aplican el principio de privilegio mínimo, permitiendo la obtención de acceso root mediante técnicas simples de modificación de permisos. Por lo tanto el veredicto es: FALLA.

El tercer vector (**V3. Actualizaciones**) evalua la seguridad del proceso de actualización del firmware. Se verifica la protección de la transmisión de los archivos de actualización frente a interceptaciones o manipulaciones, comprobando que el dispositivo rechaza versiones no firmadas, manipuladas o procedentes de fuentes no autorizadas. También se analiza su capacidad para impedir downgrades a versiones antiguas y la correcta implementación de mecanismos de cifrado en las descargas desde los servidores del fabricante.

El tercer vector evidenció deficiencias graves en el proceso de actualización de firmware. La transmisión se realiza sin cifrado, y los mecanismos de validación se basan en hashes MD5, los cuales son inseguros y modificables por el atacante, ya que tanto el firmware como los archivos de control viajan en texto claro. No se implementan verificaciones de firma digital ni protección frente a versiones anteriores, lo que permitiría instalar downgrades o versiones manipuladas. Por lo tanto el veredicto es: FALLA.

El cuarto vector (**V4. Comunicaciones**) aborda la configuración de las interfaces inalámbricas y la seguridad en las comunicaciones del dispositivo. Se verifica que los protocolos empleados (Wi-Fi, Bluetooth u otros) mantienen configuraciones por defecto seguras, evitando el uso de estándares obsoletos y vulnerables como WEP o WPA1, y promoviendo el uso de WPA2 o WPA3. Igualmente, se comprueba la existencia de conexiones seguras con los servicios en la nube mediante el uso de protocolos cifrados, y la correcta implementación de estos protocolos frente a ataques conocidos. Finalmente, se analizan las medidas de privacidad, confirmando que, incluso con la telemetría desactivada, el dispositivo no envia datos a servidores externos sin consentimiento.

En el cuarto vector se verificó que la red Wi-Fi inicial generada por el dispositivo usa el estándar WPA2-PSK, aunque la contraseña por defecto no es robusta ni única por lote. El dispositivo solicita su modificación en la primera

configuración, mitigando parcialmente el riesgo. Respecto al tráfico de datos, se observó que las comunicaciones entre el dispositivo y la aplicación móvil se establecen en UDP, utilizando un protocolo propietario no reconocible por Wireshark y, en ocasiones, sin protección TLS. Parte de las conexiones se dirigen a servidores externos (en dominios europeos y norteamericanos), algunos de los cuales no aceptan conexiones seguras. A pesar de que se detectó cifrado TLS 1.2 en varios endpoints, persiste incertidumbre sobre la seguridad global del canal. Además, el dispositivo mantiene actividad de red residual incluso cuando no está emparejado. Por lo tanto el veredicto es: INCONCLUYENTE.

El quinto vector (**V5. Portal de configuración**) se centra en el análisis del acceso web o portal de administración del dispositivo. Se revisan las URL expuestas y no documentadas, comprobando la posible exposición de recursos sensibles. Asimismo, se evalua la robustez de los mecanismos de autenticación y control de acceso, junto con la protección frente a vulnerabilidades comunes en aplicaciones web, como inyecciones SQL, ejecución de comandos, secuestro de sesión (session hijacking) y ataques XSS (reflejados, persistentes o de DOM). También se verifica la presencia de medidas como tokens CSRF, tiempos de sesión y gestión segura de cookies.

El quinto vector no aplica en este caso, ya que el dispositivo carece de un portal web de administración o interfaz de configuración accesible vía navegador. La gestión se realiza íntegramente mediante la aplicación móvil y las APIs de backend, por lo que no existen vectores web tradicionales a evaluar. Siendo el veredicto: NO APLICA.

El sexto vector (**V6. Aplicación móvil**) aborda el análisis de la aplicación asociada al dispositivo en sus versiones Android e iOS. Se evalua su nivel de seguridad general, verificando la configuración de red, la validez de los certificados y la integridad del código. Se analizan las comunicaciones entre la aplicación, el dispositivo y los servicios en la nube, asegurando que estén cifradas y protegidas mediante técnicas como SSL pinning. También se examina el almacenamiento de información sensible en el dispositivo móvil, la posible inclusión de código malicioso o permisos excesivos, y la presencia de secretos embebidos en el código fuente (como contraseñas, claves o tokens). Finalmente, se inspeccionan los registros (logs) generados por la aplicación para detectar posibles filtraciones de datos sensibles.

El sexto vector mostró un comportamiento significativamente más seguro que los componentes embebidos. El análisis estático y dinámico reveló una puntuación de riesgo global media-baja, con algunos avisos críticos relacionados con cifrados inseguros (modo ECB, MD5/SHA1, ausencia de SSL pinning efectivo) y exposición parcial de endpoints en el código. Las conexiones identificadas emplean TLS 1.2 y 1.3 en la mayoría de los casos, y las bases de datos locales no contienen información sensible en texto claro. Se detectó la presencia de permisos elevados (acceso a cámara, micrófono y red), coherentes con las funcionalidades del producto. El análisis antivirus y de comportamiento no mostró presencia de malware, y la aplicación mantiene buena reputación en los servidores destino. El veredicto es: PASA.

El séptimo vector (**V7. Autenticación**) tiene como finalidad evaluar la fortaleza de los mecanismos de acceso al dispositivo y a sus servicios asociados. Se realizan pruebas de resistencia ante ataques de fuerza bruta, verificando los límites de intentos y la presencia de bloqueos temporales. Asimismo, se comprueba la unicidad y robustez de las contraseñas preinstaladas, verificando si figuraban en brechas conocidas o diccionarios públicos. Finalmente, se analiza el proceso de restablecimiento a valores de fábrica, confirmando que no se generan credenciales genéricas o predecibles tras el reinicio.

El séptimo vector reveló debilidades en la seguridad de acceso. El uso de credenciales por defecto conocidas (usuario linaro, contraseña pública) permite obtener acceso remoto por SSH. Aunque el sistema requiere el cambio de contraseña durante la primera configuración, la falta de mecanismos de limitación de intentos y la posibilidad de escalar privilegios a root mediante comandos simples suponen un riesgo elevado. No obstante, el restablecimiento a valores de fábrica obliga a establecer una nueva contraseña, lo que reduce parcialmente el impacto. Por lo tanto el veredicto es: FALLA.

El octavo y último vector (**V8. Otros ataques / Ataques físicos**) considera la exposición del dispositivo ante manipulaciones directas. Se revisa la existencia de interfaces físicas innecesarias, garantizando la minimización de puertos y conectores accesibles que pudieran ser utilizados para comprometer el sistema. Asimismo, se evalua la configuración de las interfaces USB, asegurando que el dispositivo limita adecuadamente las clases de módulos permitidos y restrinja la interacción con periféricos no autorizados.

Finalmente, el octavo vector mostró que el dispositivo no presenta puertos externos evidentes, aunque incluye una interfaz de depuración UART protegida por tornillos, que permite el acceso directo al sistema mediante conexión física por USB. Este canal alternativo actúa de manera similar al acceso SSH, ofreciendo control total sobre el entorno si se conecta físicamente al hardware. El veredicto es: FALLA.

El análisis integral del dispositivo, el cual se muestra en la Tabla III, evidencia una superficie de ataque amplia y un número considerable de vulnerabilidades críticas concentradas en los vectores de firmware, autenticación y actualizaciones. En total, el 71,43 % de los vectores evaluados presentan fallos, un 14,29 % se consideran satisfactorios (PASA) y otro 14,29 % permanecen inconclusos. La aplicación móvil muestra una madurez superior en términos de seguridad, mientras que la infraestructura embebida (firmware y comunicaciones internas) carece de medidas modernas de protección, como cifrado robusto, verificación de integridad o aislamiento de privilegios.

En conjunto, el dispositivo no cumple con las expectativas de seguridad para un entorno conectado y orientado a menores, requiriendo una revisión profunda de su arquitectura de firmware, su mecanismo de actualización y su control de accesos. Aun así, la presencia de cifrado TLS y de un flujo de autenticación inicial obligatorio en la aplicación móvil constituye una base adecuada para futuras mejoras.

Tabla III
RESUMEN DE RESULTADOS POR VECTOR DE PRUEBA

Vector	Descripción	Veredicto
V1	Análisis de vulnerabilidades	FALLA
V2	Firmware y sistema operativo	FALLA
V3	Actualizaciones de firmware	FALLA
V4	Comunicaciones	INCONCLUYENTE
V5	Portal de configuración	N/A
V6	Aplicación móvil	PASA
V7	Autenticación	FALLA
V8	Ataques físicos	FALLA

VII. CONCLUSIONES

La evaluación realizada permite constatar que los juguetes conectados presentan un panorama heterogéneo en términos de ciberseguridad. Aunque algunos dispositivos muestran avances notables en el uso de cifrados seguros y autenticación inicial obligatoria, otros mantienen configuraciones básicas, servicios innecesarios o componentes software obsoletos que amplían significativamente su superficie de ataque. En particular, el análisis del dispositivo objeto de estudio ha evidenciado carencias críticas en la gestión del firmware, las actualizaciones y los mecanismos de autenticación, lo que coincide con las conclusiones obtenidas en otros estudios sectoriales.

Cabe destacar que los resultados de la metodología presentada en este trabajo se difundieron en el ámbito divulgativo mediante el informe oficial de INCIBE (2024), "Estudio de la ciberseguridad en juguetes conectados" [21]. Dicho estudio, desarrollado durante la campaña de Navidad 2024, confirmó una tendencia general en la que más del 20 % de los juguetes analizados presentaban deficiencias graves de seguridad y cerca de una cuarta parte arrojaban resultados inconclusos debido a la falta de transparencia técnica de los fabricantes. En consecuencia, solo algo más de la mitad de los dispositivos superaban el umbral del 50 % de pruebas favorables, lo que refleja la inmadurez del ecosistema IoT orientado al público infantil.

Estos hallazgos refuerzan la necesidad de integrar la ciberseguridad como requisito de diseño en el desarrollo de dispositivos IoT, especialmente en aquellos destinados a menores, donde la protección de datos y la integridad de la comunicación resultan esenciales.

La investigación demuestra que la ciberseguridad en los juguetes inteligentes debe ser tratada con la misma rigurosidad que en cualquier otro dispositivo conectado, ya que las vulnerabilidades técnicas en estos entornos pueden tener un impacto directo en la privacidad y seguridad del menor. La metodología propuesta contribuye a estandarizar las fases de auditoría y ofrece una base objetiva sobre la que seguir construyendo procesos de certificación y evaluación comparativa en el ámbito de la IoT infantil.

AGRADECIMIENTOS

Quisiera agradecer a todas las personas que contribuyeron a la realización de este artículo. Además de expresar mi gratitud al Plan de Recuperación, Transformación y Resiliencia de la Unión Europea (Next Generation) que por medio del proyecto "Seguridad del Internet de las Cosas en Entornos Domésticos y Empresariales en el Contexto de la Tecnología 5G-IoT" han financiado este proyecto.

REFERENCIAS

[1] D. Narciandi-Rodriguez, J. Aveleira-Mata, M. T. García-Ordás, J. Alfonso-Cendón, C. Benavides, and H. Alaiz-Moretón, "A cybersecurity review in iot 5g networks," *Internet of Things*, vol. 30, p. 101478, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660524004190

[2] P. Kumari and A. K. Jain, "A comprehensive study of ddos attacks over iot network and their countermeasures," *Computers & Security*, vol. 127, p. 103096, 4 2023.

[3] B. Sudharsan, D. Sundaram, P. Patel, J. G. Breslin, and M. I. Ali, "Edge2guard: Botnet attacks detecting offline models for resource-constrained iot devices," *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events, PerCom Workshops 2021*, pp. 680–685, 3 2021.

[4] A. E. Agazzi, "Smart home, security concerns of iot," *Cornell University*, 2020.

[5] M. Azrour, J. Mabrouki, A. Guezzaz, A. Kanwal, and H. U. Khan, "Internet of things security: Challenges and key issues," *Security and Communication Networks*, vol. 2021, 2021. [Online]. Available: https://doi.org/10.1155/2021/5533843

[6] M. A. J. Jamali, A. Heidari, P. Allahverdizadeh, F. Norouzi, and B. Bahrami, "Iot security," *EAI/Springer Innovations in Communication and Computing*, pp. 33–83, 2020. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-18468-1_3

[7] ETSI, "En 303 645 - v2.1.1 - cyber; cyber security for consumer internet of things: Baseline requirements," 6 2020.

[8] M. A. Khan and K. Salah, "Iot security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, vol. 82, pp. 395–411, 5 2018.

[9] T. Hall, I. Poyner, A. Phillips, R. Marshall, and G. Markall, Eds., *IoT Security Assurance Framework*, 3rd ed. IoT Security Foundation, 2021.

[10] T. Rajmohan, P. H. Nguyen, and N. Ferry, "A decade of research on patterns and architectures for iot security," *Cibersecurity*, vol. 5, 2022. [Online]. Available: https://doi.org/10.1186/s42400-021-00104-7

[11] J. Aveleira-Mata, D. Narciandi-Rodriguez, I. Garcia-Rodriguez, J. Alfonso-Cendon, S. Rubio-Martin, and H. Alaiz-Moreton, "Auditoría funcional de juguetes iot con bluetooth le," *Jornadas de Automatica*, vol. 46, 2025.

[12] ETSI, "Ts 103 701 - v1.1.1 - cyber; cyber security for consumer internet of things: Conformance assessment of baseline requirements," 2021. [Online]. Available: https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

[13] M. Fagan, K. N. Megas, K. Scarfone, and M. Smith, "Iot device cybersecurity capability core baseline," National Institute of Standards and Technology (NIST), NIST Interagency/Internal Report NISTIR 8259A, 2020.

[14] Department for Digital, Culture, Media & Sport (DCMS), "Code of practice for consumer iot security," oct 2018. [Online]. Available: https://www.gov.uk/government/publications/code-of-practice-for-consumer-iot-security

[15] OWASP Foundation, "Owasp iot security verification standard (isvs)," 2025. [Online]. Available: https://github.com/OWASP/IoT-Security-Verification-Standard-ISVS

[16] J. Valente and A. A. Cardenas, "Security & privacy in smart toys," in *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy (IoT S&P '17)*. Association for Computing Machinery (ACM), 2017, pp. 19–24.

[17] G. Chu, N. Apthorpe, and N. Feamster, "Security and privacy analyses of internet of things children's toys," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 978–985, 2019.

[18] L. G. de Carvalho and M. M. Eler, "Security requirements for smart toys," in *Proceedings of the 19th International Conference on Enterprise Information Systems (ICEIS 2017)*. SCITEPRESS, 2017.

[19] J. Feldbusch, V. Pavliv, N. Akbari, and I. Wagner, "No transparency for smart toys," in *Privacy Technologies and Policy*. Springer, 2024. [Online]. Available: https://doi.org/10.1007/978-3-031-68024-3_112

[20] R. Boqué, A. Maroto, J. Riu, and F. X. Rius, "En iso/iec 17025." [Online]. Available: https://www.iso.org/obp/ui#iso:std:iso-iec:17025:ed-3:v2:es

[21] INCIBE, "Estudio de la ciberseguridad juguetes conectados," Tech. Rep., 01 2024. [Online]. Available: www.incibe.es/sites/default/files/espacios/ed2026/laboratorio/EstudioINCIBE_ciberseguridad_juguetes_conectados.pdf

TrustAIoT: de la confiabilidad sociotécnica a la detección temprana ecosistémica

Carlos Mario Braga
Grp. Invest. Alarcos
Unv. de Castilla-La Mancha
UCLM, Ciudad Real
CarlosMario.Braga1@alu.uclm.es

Manuel A. Serrano
Grp. Invest. Alarcos
Unv. de Castilla-La Mancha
UCLM, Ciudad Real
Manuel.Serrano@uclm.es

Eduardo Fernández-Medina
Grp. Invest. GSyA
Unv. de Castilla-La Mancha
UCLM, Ciudad Real
Eduardo.FdezMedina@uclm.es

***Resumen*—La creciente integración entre la inteligencia artificial y el internet de las cosas ha dado lugar a ecosistemas cada vez más complejos y autónomos, que plantean nuevos desafíos de seguridad, confianza y gobernanza, donde las soluciones técnicas tradicionales resultan insuficientes para abordar la interacción entre factores tecnológicos, regulatorios y éticos. La falta de metodologías que gestionen de forma integral estas dimensiones ha impulsado la búsqueda de marcos que unifiquen la protección técnica con la trazabilidad, la sostenibilidad y la resiliencia operativa de los sistemas inteligentes.**

En este contexto, el presente artículo presenta y contextualiza el trabajo *TrustAIoT: A framework for building trustworthy AIoT platforms*, publicado en Internet of Things Journal (2025), que constituye la contribución central de una línea de investigación orientada a la construcción de ecosistemas confiables y seguros. El modelo se fundamenta en un proceso iterativo de seis subprocesos y una *guía de correspondencia* que vincula dichas actividades con los niveles funcionales del sistema, dando lugar a una arquitectura de ocho capas que integra las dimensiones éticas, legales y técnicas de la confianza a lo largo del ciclo de vida de las plataformas AIoT.

El contraste con el *Industrial Internet Reference Architecture* (IIRA) permitió verificar la coherencia del modelo y su compatibilidad con los marcos industriales existentes. El análisis evidenció que *TrustAIoT* amplía el concepto de confiabilidad al incorporar mecanismos para su evaluación y mejora continua durante todo el ciclo de vida del sistema. Esta extensión redefine la protección digital desde una perspectiva ecosistémica, en la que la seguridad se entiende como la capacidad del sistema para mantener su equilibrio sociotécnico y coherencia operativa frente a cambios o amenazas, consolidando una gestión de la confianza integrada y sostenible.

***Index Terms*—IA confiable, AIoT, confianza, ciberseguridad, gobernanza, arquitectura sociotécnica, detección temprana**

I. Introducción

La convergencia entre la inteligencia artificial y el internet de las cosas ha dado lugar al paradigma AIoT, en el que la autonomía, la interoperabilidad y el tratamiento masivo de datos redefinen los principios clásicos de seguridad y confianza [1, 2]. Estos ecosistemas inteligentes, caracterizados por su distribución, heterogeneidad y capacidad de aprendizaje continuo, plantean nuevos desafíos que trascienden los límites de la ciberseguridad tradicional [3, 4]. En particular, la necesidad de gestionar simultáneamente aspectos técnicos, legales y éticos convierte la confiabilidad en un problema de naturaleza sociotécnica, donde la protección no se reduce a controles o normas, sino que depende del equilibrio entre las distintas dimensiones del sistema [5–7].

Los marcos industriales y arquitectónicos existentes, como el *Industrial Internet Reference Architecture (IIRA)* [8], han sido fundamentales para estandarizar la organización funcional de los sistemas inteligentes. No obstante, su alcance se centra principalmente en los aspectos técnicos y operativos, sin abordar de manera explícita los factores humanos, regulatorios y éticos que condicionan la confianza en estos entornos [9, 10]. Esta limitación ha motivado la búsqueda de modelos capaces de integrar dichas dimensiones dentro de un mismo proceso de gobernanza, articulando la seguridad y la confiabilidad como elementos interdependientes [11, 12].

En este contexto, el presente artículo contextualiza el trabajo *"TrustAIoT: A framework for building trustworthy AIoT platforms"*, publicado en Internet of Things Journal (2025) [13], contribución central de una línea orientada a la gobernanza confiable en ecosistemas inteligentes. El modelo se apoya en un proceso iterativo de seis subprocesos y en una *guía de correspondencia* que vincula dichas actividades con los niveles funcionales del sistema, generando una arquitectura de ocho capas para gestionar de forma integrada la confianza y la seguridad durante todo el ciclo de vida. A diferencia de los enfoques centrados en la ingeniería o el cumplimiento normativo, *TrustAIoT* propone una visión sociotécnica que articula las dimensiones éticas, legales y técnicas en un único proceso operativo, convirtiendo la confiabilidad en un atributo dinámico y verificable.

Este trabajo se apoya en los avances presentados en los congresos JNIC 2024 [11] y RECSI 2024 [12], donde se propuso una *taxonomía sociotécnica de la confianza* y se tuvo la oportunidad de obtener las primeras validaciones y comentarios de la comunidad científica sobre su planteamiento. Estos foros resultaron esenciales para contrastar y afinar los fundamentos del modelo, poniendo de manifiesto la importancia de los congresos nacionales como espacios de discusión y maduración de líneas de investigación.

En un paso posterior, la taxonomía fue formalizada y publicada en Expert Systems with Applications (2025) [7], donde se establecieron los principios metodológicos y la estructura conceptual que servirían de base para la construcción de un marco aplicable a entornos reales. Sobre esa base se desarrolló el trabajo *TrustAIoT: A framework for building trustworthy AIoT platforms* [13], que constituye la materialización práctica de la visión inicial. La plataforma desarrollada con este marco permitió validar su aplicabilidad en la gestión integrada de la confianza y la seguridad, y contrastar el proceso generado con el IIRA [8]. Este contraste mostró que, aunque ambos modelos comparten principios estructurales, *TrustAIoT* amplía el alcance del concepto de confiabilidad, incorporando de

forma explícita dimensiones de ética, sostenibilidad y buenas prácticas como pilares operativos del sistema. De este modo, la confianza deja de basarse exclusivamente en parámetros técnicos para convertirse en una propiedad global del ecosistema, sustentada en la coherencia entre sus componentes tecnológicos, normativos y sociales.

En conjunto, este trabajo consolida la confiabilidad y la seguridad como propiedades estructurales del ecosistema digital, gestionadas mediante retroalimentación y equilibrio entre las dimensiones sociotécnicas del sistema. A partir de esta base, el resto del artículo se organiza del siguiente modo: la Sección II expone los fundamentos sociotécnicos y cibernéticos del modelo; la Sección III presenta el marco *TrustAIoT*, incluyendo la arquitectura, el proceso metodológico y la guía de correspondencia que articula su coherencia estructural; la Sección IV desarrolla la discusión e implicaciones del modelo, destacando su aportación a la gestión de la confiabilidad y la seguridad desde una perspectiva ecosistémica, y finalmente, la Sección V resume las principales conclusiones e introduce un primer ejemplo de aplicación de estas ideas en el ámbito de la ciberseguridad, como caso de estudio que muestra cómo los principios de equilibrio y trazabilidad del marco pueden aplicarse de forma natural a la detección de ataques.

II. Fundamentos sociotécnicos y cibernéticos del modelo

El crecimiento de los ecosistemas AIoT ha potenciado la eficiencia operativa en sectores como la industria, la energía o la salud, pero también ha incrementado la complejidad de los entornos de seguridad [1, 2]. La heterogeneidad de dispositivos, la autonomía de los agentes y el flujo continuo de datos exigen un enfoque de protección que vaya más allá de los controles técnicos tradicionales [3, 4]. En estos sistemas, la confiabilidad se convierte en un requisito transversal que integra dimensiones éticas, legales y tecnológicas, y cuya gestión requiere modelos capaces de articularlas de forma coherente a lo largo del ciclo de vida.

Los marcos de referencia existentes, como el IIRA [8], han establecido estructuras de referencia útiles para la interoperabilidad y la gestión técnica de sistemas IoT complejos. Sin embargo, su alcance se centra principalmente en las capas funcionales y operativas, sin contemplar de forma explícita los factores humanos, regulatorios o de sostenibilidad que condicionan la confianza en los entornos inteligentes actuales [9, 10]. Esta limitación ha impulsado el desarrollo de modelos ampliados que integren la gobernanza ética, la trazabilidad regulatoria y la responsabilidad social como elementos estructurales del diseño y la operación [5, 6].

Bajo esta premisa, el marco *TrustAIoT* se fundamenta en una taxonomía sociotécnica de la confianza, desarrollada en trabajos previos [7], que define la interacción entre tres dominios interdependientes: el tecnológico, el legal y el ético. Esta interacción constituye el núcleo de un modelo de gestión de la confianza donde la seguridad deja de entenderse como un conjunto de medidas aisladas para abordarse como un proceso dinámico de equilibrio y retroalimentación.

La concepción sociotécnica de esta taxonomía asume que los sistemas inteligentes no son únicamente artefactos tecnológicos, sino ecosistemas de información y decisión en los que los componentes humanos, normativos y técnicos se co-determinan. Esta visión se inspira en la cibernética de Wiener [14], desde la cual la confianza y la seguridad emergen de la retroalimentación continua entre los niveles del sistema, permitiendo entender la protección como una propiedad dinámica basada en la autorregulación, más que como un estado estático. Este principio constituye la base de la línea de investigación y se materializa en el marco *TrustAIoT*, donde el proceso metodológico y la guía de correspondencia articulan dichas dimensiones para generar una arquitectura capaz de gestionar la confiabilidad como una propiedad estructural y evolutiva de los ecosistemas digitales.

III. Marco *TrustAIoT*: arquitectura y proceso

El marco *TrustAIoT* [13] se fundamenta en un proceso metodológico iterativo diseñado para gestionar de manera integrada la confianza, la seguridad y la sostenibilidad en ecosistemas AIoT. Este proceso, junto con la guía de correspondencia, da origen a una arquitectura de ocho capas que permite abordar la confiabilidad como un atributo transversal desde la concepción hasta la operación y mejora continua del sistema.

La Figura 1 muestra la visión general del proceso *TrustAIoT*. Este ciclo de vida está estructurado en seis subprocesos interrelacionados, *vision, assessment, design, development, policies y deployment*, que permiten construir y mantener plataformas AIoT confiables. Cada subproceso incluye tareas de evaluación y revisión continua, de modo que los resultados de una fase retroalimentan a las siguientes, garantizando la trazabilidad y la mejora constante del sistema.

El esquema refleja la naturaleza iterativa y adaptativa del marco, en la que las decisiones técnicas, legales y éticas se integran en un mismo flujo de trabajo. Así, el proceso combina la planificación estratégica (visión y evaluación), la materialización operativa (diseño y desarrollo) y la gobernanza posterior (políticas y despliegue), conformando un mecanismo de retroalimentación cíclica que permite a las plataformas mantener la coherencia y la confianza a lo largo de su ciclo de vida.

El marco *TrustAIoT* se basa en un proceso metodológico iterativo y una guía de correspondencia que articulan una arquitectura de ocho capas, diseñada para gestionar de forma integrada la confianza, la seguridad y la sostenibilidad en ecosistemas AIoT. Esta guía vincula las actividades del proceso con los niveles arquitectónicos, asegurando la coherencia entre decisiones metodológicas, diseño estructural y gobernanza del sistema.

La arquitectura organiza los componentes del sistema según su papel en la generación y mantenimiento de la confianza. A las capas clásicas del IIRA, *Business, Operations, Information, Application* y *Control*, se suman tres niveles que refuerzan la gestión continua de la confiabilidad: *Ética y sostenibilidad*, que incorpora principios éticos y de transparencia; *Infraestructura de soporte*, que añade mecanismos de auditoría y monitorización; y *Mejores prácticas*, que consolida conocimiento organizativo mediante directrices y mejora continua.

Esta ampliación proporciona un marco holístico donde las dimensiones técnicas, legales y éticas se articulan como componentes interdependientes. Cada capa contribuye a la

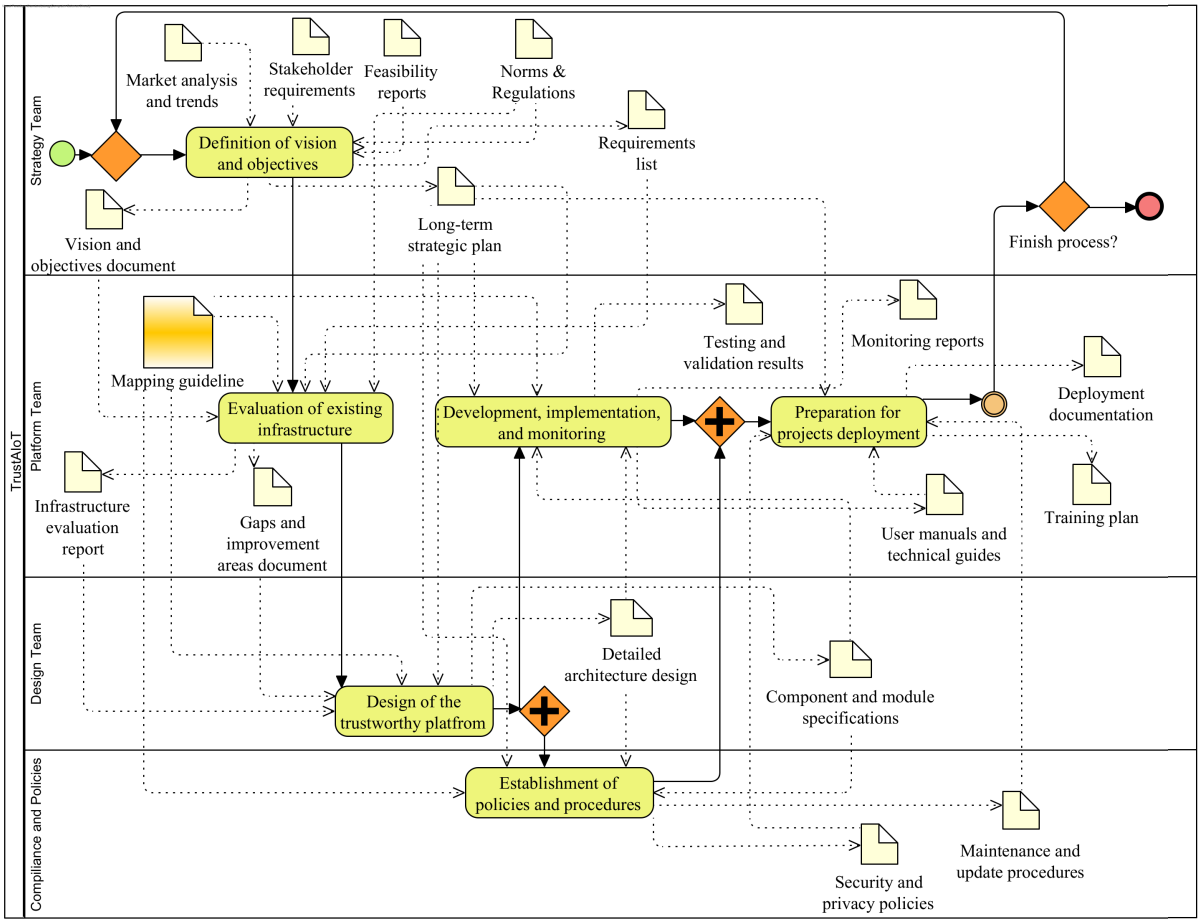


Figura 1. Proceso *TrustAIoT*, con los seis subprocesos interrelacionados que integran evaluación, diseño, desarrollo y gobernanza a lo largo del tiempo.

gobernanza global del sistema mediante un flujo bidireccional de información que preserva la coherencia entre niveles operativos y normativos.

El proceso metodológico consta de seis subprocesos iterativos, *visión, evaluación, diseño, desarrollo, políticas* y *despliegue*, que traducen los principios de confianza en acciones operativas. En la fase de visión se definen los objetivos y expectativas; la evaluación establece indicadores de riesgo y validación ética; el diseño y desarrollo integran protección, privacidad y sostenibilidad; las políticas gestionan identidades, acceso y gobernanza de datos; y el despliegue consolida los mecanismos de auditoría y mejora continua.

Este ciclo no es lineal, sino adaptativo y evolutivo, de manera que los resultados de una fase alimentan las decisiones de la siguiente. Así, la confianza se gestiona como un proceso vivo, sustentado en la trazabilidad de las decisiones, la transparencia y la capacidad del sistema para mantener su equilibrio operativo ante cambios o incidentes.

La integración entre el proceso iterativo y la guía de correspondencia da lugar a la arquitectura de ocho capas que define *TrustAIoT*. Esta arquitectura no es un esquema previo, sino el resultado de alinear las actividades metodológicas con los niveles estructurales del sistema, permitiendo que la confiabilidad emerja como una propiedad verificable y adaptable del ecosistema AIoT.

IV. Discusión e implicaciones del modelo

El contraste con el IIRA permite situar *TrustAIoT* como una evolución natural de los marcos industriales tradicionales. Más que una sustitución, representa una extensión de su lógica estructural hacia la confiabilidad sociotécnica, integrando de forma explícita los factores éticos, legales y organizativos en la gobernanza de sistemas inteligentes.

Este enfoque supone un cambio de paradigma en la gestión de la seguridad y la confianza, en lugar de entender la protección como un conjunto de mecanismos independientes, *TrustAIoT* la concibe como una propiedad dinámica del ecosistema, mantenida mediante procesos de retroalimentación continua entre los niveles técnico, legal y humano. Esta articulación permite que la confiabilidad se consolide como un atributo verificable y sostenible del sistema, capaz de adaptarse a los cambios normativos, tecnológicos y organizativos sin perder coherencia estructural y operativa.

Operativamente, el modelo integra auditorías éticas y de cumplimiento normativo, reduciendo la complejidad de certificación en sectores como la industria, la energía o el transporte. Asimismo, su carácter adaptable lo convierte en una base metodológica para extender la gestión de la confiabilidad a otros dominios, como los sistemas de aprendizaje federado o las infraestructuras críticas. En este sentido, *TrustAIoT* sienta las bases de una *seguridad ecosistémica*, entendida como la capacidad del sistema para mantener su coherencia y equilibrio sociotécnico frente a cambios o amenazas.

V. Conclusiones

Este trabajo presenta el modelo *TrustAIoT* [13], contribución central de una línea de investigación orientada a la confiabilidad y la seguridad en ecosistemas inteligentes. El marco propone una integración coherente de las dimensiones éticas, legales y técnicas consolidada en una arquitectura de ocho capas que surge de un proceso iterativo de seis subprocesos y una guía de correspondencia, que garantiza la coherencia estructural entre las actividades metodológicas y los niveles arquitectónicos. De este modo, la confianza deja de entenderse como un atributo estático para convertirse en un proceso operativo, medible y trazable, capaz de adaptarse a las condiciones cambiantes de los entornos AIoT.

Más allá de su alineamiento con marcos industriales como el IIRA, *TrustAIoT* amplía la interoperabilidad técnica hacia una visión sociotécnica y ecosistémica de la seguridad, ofreciendo una base verificable para la gobernanza confiable de plataformas AIoT y garantizando trazabilidad, transparencia y sostenibilidad a lo largo de su ciclo de vida

Las implicaciones de este planteamiento trascienden el ámbito arquitectónico. El principio de equilibrio sociotécnico y de retroalimentación continua sobre el que se fundamenta *TrustAIoT* abre una nueva vía para la detección y respuesta temprana ante perturbaciones en ecosistemas digitales. Esta línea, actualmente en desarrollo, explora la detección ligera como una *ruptura de simetría ecosistémica*, donde los desequilibrios estructurales en el comportamiento del sistema actúan como indicadores anticipados de anomalías o ataques. Los primeros resultados de esta aproximación [15], aplicados a aprendizaje federado, confirman la viabilidad del enfoque al detectar ataques de puerta trasera en etapas tempranas sin afectar la precisión del modelo, consolidando una trayectoria de investigación que conecta la confiabilidad con una ciberseguridad adaptativa.

Agradecimientos

Esta publicación no habría sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MCIN/AEI/ 10.13039/501100011033 Unión Europea (Chist-Era Program); KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MCIN/AEI/10.13039/ 501100011033/ FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la Junta de Comunidades de Castilla La Mancha, y el Fondo Europeo de Desarrollo Regional FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; y RED2024-154240-T financiado por MICIU/AEI/10.13039/501100011033.

Referencias

[1] Z. Lv, Y. Han, A. K. Singh *et al.*, "Trustworthiness in industrial IoT systems based on artificial intelligence," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 2, pp. 1496–1504, 2020.

[2] S.-V. Oprea and A. Bâra, "An edge-fog-cloud computing architecture for IoT and smart metering data," *Peer-to-Peer Networking and Applications*, vol. 16, no. 2, pp. 818–845, 2023.

[3] G. Rathee, S. Garg, G. Kaddoum *et al.*, "TrustSys: Trusted decision making scheme for collaborative artificial intelligence of things," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 1059–1068, 2022.

[4] S. Afzal-Houshmand, D. Papamartzivanos, S. Homayoun *et al.*, "Explainable artificial intelligence to enhance data trustworthiness in crowd-sensing systems," in *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. IEEE, 2023, pp. 568–576.

[5] L. Floridi *et al.*, "AI4People—an ethical framework for a good AI society: opportunities, risks, principles, and recommendations," *Minds and machines*, vol. 28, pp. 689–707, 2018.

[6] A. Jobin, M. Ienca, and E. Vayena, "The global landscape of AI ethics guidelines," *Nature machine intelligence*, vol. 1, no. 9, pp. 389–399, 2019.

[7] C. M. Braga, M. A. Serrano, and E. Fernández-Medina, "Towards a methodology for ethical artificial intelligence system development: A necessary trustworthiness taxonomy," *Expert Systems with Applications*, vol. 286, p. 128034, 2025.

[8] S.-W. Lin, B. Miller, J. Durand, R. Joshi, P. Didier, A. Chigani, R. Torenbeek, D. Duggal, R. Martin, G. Bleakley *et al.*, "Industrial internet reference architecture," *Industrial Internet Consortium (IIC), Tech. Rep*, 2015.

[9] High-Level Expert Group on AI, "Ethics guidelines for trustworthy AI," 2019, european Commission. Accessed: 2024-05-21. [Online]. Available: https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai

[10] J. Morley, L. Floridi, L. Kinsey *et al.*, "From what to how: an initial review of publicly available AI ethics tools, methods and research to translate principles into practices," *Science and engineering ethics*, vol. 26, no. 4, pp. 2141–2168, 2020.

[11] C. M. Braga, M. A. Serrano, and E. F. Medina, "Hacia una propuesta de metodologıa para el desarrollo de proyectos de ia confiables," *Jornadas Nacionales de Investigación en Ciberseguridad (JNIC). Sevilla, Espana: Universidad de Sevilla. Escuela Técnica Superior de Ingenierıa Informática*, pp. 270–277, 2024.

[12] C. M. Braga, Á. Suárez Bárcena, M. A. Serrano, E. Fernández-Medina *et al.*, "Confiabilidad en sistemas aiot," *XVIII Reunión Española sobre Criptología y Seguridad de la Información: XVIII RECSI, León 23-25 octubre 2024*, pp. 65–71, 2025.

[13] C. M. Braga, Á. Suarez-Bárcena, M. A. Serrano, and E. Fernández-Medina, "TrustAIoT: A framework for building trustworthy AIoT platforms," *Internet of Things*, p. 101751, 2025.

[14] N. Wiener, *The human use of human beings: Cybernetics and society*. Cambridge, Massachusetts: The riverside press, 1950.

[15] C. M. Braga, M. A. Serrano, and E. Fernández-Medina, "Early detection of backdoor attacks in federated learning via ecosystemic symmetry breaking," in *Proceedings of the IEEE/ACM 12th International Conference on Big Data Computing, Applications and Technologies*

(BDCAT '25). Nantes, France: Association for Computing Machinery, 2025.

From DP to BDP: Noise Recalibration for Correlation-Resilient Privacy Guarantees

Patricia Guerra-Balboa*
Karlsruhe Institute of Technology, Germany
patricia.balboa@kit.edu

Martin Lange*
Karlsruhe Institute of Technology, Germany
lange@martin-lange.eu

Javier Parra-Arnau
Universitat Politècnica de Catalunya, Spain
javier.parra@upc.edu

Thorsten Strufe
Karlsruhe Institute of Technology, Germany
strufe@kit.edu

***Abstract*—This paper is an extended summary of previously published results [1].**

Dependencies among data records are inherent in most data analysis tasks, posing a fundamental challenge to differential privacy (DP). Privacy risks increase sharply when data are correlated and DP often underestimates information leakage and fail to fully protect sensitive details. Bayesian Differential Privacy (BDP) extends the DP framework to account for correlations; however, existing BDP mechanisms typically incur substantial utility loss, limiting their practical applicability.

We analyze arbitrary and structured correlation models, including Gaussian multivariate distributions and Markov chains, deriving formal and empirical utility guarantees for BDP. Our contributions include theoretical links between DP and BDP and a novel methodology to adapt DP mechanisms to meet the requirements of BDP. Through evaluations on real-world databases, we demonstrate that our novel theorems enable the design of BDP mechanisms that maintain competitive utility, paving the way for practical privacy-preserving data practices in correlated settings.

I. Introduction

While differential privacy (DP) [2] became the de-facto standard in the field of data protection, it underestimates information leakage risk in the presence of correlations, an issue both theoretically established [3], [4] and empirically demonstrated through real-world attacks [5]. To overcome this limitation, Bayesian differential privacy (BDP) [4] appears as an extension of DP that explicitly accounts for correlations among data records, hence provides stronger protection in correlated settings, while preserving DP's desirable properties such as post-processing and composability.

Although BDP provides a solid theoretical framework for analyzing privacy under data dependencies, its practical applicability remains limited. Only a few mechanisms satisfy this notion [4], [6], and these are restricted to specific settings, such as binary-state Markov chains with symmetric transitions [6], or suffer from overly conservative utility, as noise scales linearly with the database size. The only proposed improvements require relaxing BDP guarantees through adversarial assumptions [4]. This scarcity and narrow scope cast doubt on the viability of BDP as a general privacy notion.

Hence, we examine BDP's utility from both theoretical and practical perspectives, analyzing its limitations and proposing new strategies to reduce utility loss while maintaining BDP privacy guarantees. Particularly, we present theoretical bounds on the accuracy of BDP mechanisms and derive specific utility guarantees when certain correlation models are assumed.

Prior impossibility results [3], [1] demonstrate that achieving utility under BDP without distributional assumptions is fundamentally limited. Motivated by this, we begin by analyzing the minimal assumption of bounding the number of correlated records. Our analysis yields a negative result—a tight bound showing that the required noise grows linearly with the number of correlated records—which motivates further exploration into whether additional distributional assumptions can improve utility. We then examine the applicability of BDP to both discrete and continuous correlation models. For the discrete case, we analyze data following a Markov chain and, for continuous data, we analyze multivariate Gaussian correlation. We focus on these two particular correlation models following previous work in BDP [4] and due to their relevance in many real-world applications such as medical, location, or activity data [1].

For each correlation model studied, we prove novel theorems that bound the BDP leakage of a DP mechanism. Notably, our BDP leakage bound for Gaussian multivariate models is tighter than that provided in [4], and our correlation model is broader. These privacy bounds provide a systematic way to build BDP mechanisms by adjusting the parameters of existing DP mechanisms. Using this approach, we propose novel BDP mechanisms based on Laplace noise. Furthermore, we calculate the accuracy of our BDP mechanisms showing the improved accuracy compared to scenarios where protection is required against any correlation.

Finally, we provide insight into how our theoretical results apply in practice to real-world data containing Gaussian and Markov correlations. This allows us to confirm that our results enhance the utility of BDP mechanisms in actual applications.

Our contributions are summarized as:

- We prove a bound on the BDP leakage of a DP mechanism with a fixed number of arbitrarily correlated records (the *general bound*), showing it is tight.
- We derive a tighter BDP leakage bound for DP mechanisms under multivariate Gaussian correlations, improving on the general bound and prior work. This provides a systematic method for constructing more accurate BDP

* Equal contribution.

mechanisms tailored to Gaussian dependencies.
- We derive a BDP leakage bound for DP mechanisms under Markovian correlations, improving the general bound when transition probabilities are similar. This enables the design of mechanisms that are more accurate than prior approaches in Markov settings.

All proofs are in the long version of [1] (arXiv:2506.21308).

II. BACKGROUND

The bounded formulation of DP [2] assumes that the database consists of a finite number n of rows, $D = (x_1, \ldots, x_n)$ drawn from the joint distribution of the random vector $\mathbf{X} = (X_1, \ldots, X_n)$, where each row represents data associated with an individual, sampled from a universe of records $\mathcal{X}$. We use $[n] := \{1, \ldots, n\}$ to denote the set of indices. For a subset $K = \{i_1, \ldots, i_k\} \subseteq [n]$, we define the subvector $\mathbf{X}_K \in \mathcal{X}^k$ as $\mathbf{X}_K := (X_{i_1}, \ldots, X_{i_k})$. In particular, $\mathbf{X}_{-i}$ denotes $\mathbf{X}_K$ with $K = [n] \setminus \{i\}$. In DP the attacker is assumed to know all records except for a target index $i \in [n]$, for which all possible values x_i and x_i' must be indistinguishable. Given the output of $\mathcal{M}$ represented by the random variable Y, the DP leakage is

$$\mathrm{DPL}(\mathcal{M}) = \sup_{x_i, x_i', \mathbf{x}_{-i}, S} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_{-i} = \mathbf{x}_{-i}, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_{-i} = \mathbf{x}_{-i}, X_i = x_i']},$$

where the supremum is taken over all the possible target values $x_i, x_i' \in \mathcal{X}$, all the possible known vector values $\mathbf{x}_{-i} \in \mathcal{X}^{n-1}$ and all the measurable sets $S \subseteq \mathrm{Range}(\mathcal{M})$. A mechanism $\mathcal{M}$ is called ε-DP when $\mathrm{DPL}(\mathcal{M}) \leq \varepsilon$.

Extending previous definition to other adversaries with different known indexes K, we obtain the *adversary-specific Bayesian differential privacy leakage*:

$$\mathrm{BDPL}_{(K,i)} = \sup_{x_i, x_i', \mathbf{x}_K, S} \ln \frac{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i]}{\Pr[Y \in S \mid \mathbf{X}_K = \mathbf{x}_K, X_i = x_i']}.$$

Hence, a mechanism $\mathcal{M}$ satisfies *ε-Bayesian differentially privacy* if $\mathrm{BDPL}(\mathcal{M}) = \sup_{K,i} \mathrm{BDPL}_{(K,i)}(\mathcal{M}) \leq \varepsilon$.

One of the best known ε-DP mechanisms is the Laplace mechanism [2]: Let $f : \mathcal{X}^n \to \mathbb{R}^k$ be a query function and its *sensitivity* defined as

$$\Delta f := \sup_{d_H(D, D')=1} ||f(D) - f(D')||_1,$$

where d_H denotes the Hamming distance. Given $\Delta f < \infty$, the Laplace mechanism is defined as $\mathcal{M}_{\varepsilon,f}(D) = f(D) + (Z_1, \ldots, Z_k)$ where Z_i are i.i.d. random variables that follow the Laplace distribution centered at 0 and with scale $\frac{\Delta f}{\varepsilon}$.

While $\mathcal{M}_{\varepsilon,f}$ provides ε-DP, adding noise to the output of a function f undoubtedly has an impact on utility. A well-established metric for quantifying the utility of a private mechanism is the (α, β)-accuracy [2]: A randomized mechanism $\mathcal{M}$ is *(α, β)-accurate with respect to function f* if for all databases $D \in \mathcal{X}^n$ we have

$$\Pr[|\mathcal{M}(D) - f(D)| \geq \alpha] \leq \beta.$$

In other words, with probability at least $1 - \beta$, the output of $\mathcal{M}$ lies within an interval of radius α around the true value. Here, α measures the allowable error, while β quantifies the failure probability, hence, smaller values of either indicate higher accuracy. For example for any $\beta \in (0, 1]$ the Laplace mechanism $\mathcal{M}_{\varepsilon,f}$ is (α, β)-accurate with respect to f with

$$\alpha = \ln\left(\frac{1}{\beta}\right) \frac{\Delta f}{\varepsilon}. \tag{1}$$

This accuracy result for the Laplace mechanism is tight [2].

III. LIMITED NUMBER OF CORRELATED VARIABLES

To protect against potential correlations without making distributional assumptions a mechanism must satisfy BDP with respect to all possible correlation distributions π, a condition we call protection under *arbitrary correlation*. However, Kifer and Machanavajjhala [3] showed that under this assumption, any Pufferfish notion, including BDP, collapses to free-lunch privacy implying a complete loss of utility. However, it is often reasonable in practice to assume that only subsets of records are correlated. For instance, in the context of genomic data, an individual's genome is strongly correlated with that of their relatives, but not with the entire population [7]. Hence, we assume that only m of n records in the database are correlated with each other, formally: We say the random vector $\mathbf{X} = (X_1, \ldots, X_n)$ has *at most $m \leq n$ correlated random variables* if there exist disjoint sets of indices $C_1, \ldots, C_r$ that make up $[n] = \bigcup_{l=1}^{r} C_l$ so that each set C_l has maximum cardinality $m \geq |C_l|$ for any $l \in [r]$, and for any $l \in [r]$, the random variables $\{X_j \mid j \in C_l\}$ are independent of the remaining random variables $\{X_j \mid j \in [n] \setminus C_l\}$.

This definition considers multiple groups of up to m correlated records as long as they do not "overlap", i.e., the records in one group are independent of the records in the other groups. Otherwise, we do not make any further assumptions about the distribution of the data. Then, we obtain:

Theorem III.1 (The General Bound)**.** *Let $\mathbf{X}$ be a random vector with at most $m \leq n$ correlated variables. Then, any ε-DP mechanism with input data drawn from $\mathbf{X}$ is $m\varepsilon$-BDP.*

This bound may seem overly pessimistic, seemingly assuming perfect positive correlation. However, as we prove in [1], the bound remains tight even when this extreme case is excluded. Specifically, we provide a counterexample in which the bound holds even when ρ is arbitrarily small, i.e., the variables do not deterministically determine one another. This confirms both the tightness of our result and that the bound cannot be improved, even in the absence of perfect correlation.

Theorem III.1 enables to use $(\frac{\varepsilon}{m})$-DP mechanisms as ε-BDP mechanisms. However, by direct application of Equation (1), we obtain that for the same β, the error α of the Laplace mechanism increases proportionally with the number of correlated records when moving from ε-DP to ε-BDP, without further distributional assumptions. For example, if $m = 2$, the error α doubles when transitioning from DP to BDP. If the DP mechanism's error is small, this increase may be acceptable. However, utility sharply decreases as m grows.

Since Theorem III.1 bound on BDPL is tight, the utility bound cannot be improved. This motivates the next two sections, where we investigate whether additional assumptions on the correlation model can lead to tighter bounds, enabling reduced noise and improved utility.

IV. Multivariate Gaussian Correlation

For a database of n records drawn from a Gaussian distribution, the joint distribution of the entire dataset can be modeled as a multivariate Gaussian, which naturally captures any linear correlations between records [8].

Since such assumption implies the data domain to be unbounded, we need to apply a pre-processing that clips the data to a given interval to ensure bounded sensitivities. Formally, given $I = [a, b] \subset \mathbb{R}$, we define the *clipping function* as $c_I(D)_i = \max(a, \min(b, D_i))$. We also define $\mathcal{M}$ to be $\varepsilon\ell_1$-private if $\mathcal{M}$ satisfies that $\mathrm{DPL}(\mathcal{M})(x, x') \leq \ell_1(x, x')$. Under these assumptions we obtain:

Theorem IV.1 (The Gaussian Bound)**.** *Let $\mathcal{M}$ with data domain $\mathbb{R}^n$ be an $\varepsilon\ell_1$-private mechanism with input data drawn from a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma_\rho)$ with mean $\mu \in \mathbb{R}^n$, maximum number of correlated records $m \leq n$ and limited covariance matrix $\Sigma_\rho \in \mathbb{R}^{n\times n}$ such that $\rho(m-2) < 1$ is the maximum correlation coefficient. Then, the clipped mechanism $\mathcal{M}_I = \mathcal{M} \circ c_I$ is*

$$\left(\frac{m^2}{4(\frac{1}{\rho} - m + 2)} + 1\right) M\varepsilon\text{-BDP.}$$

where M is the diameter of the interval I.

Theorem IV.1 allows us to systematically build a BDP mechanism by recalibrating the noise of a DP mechanism when $\rho(m-2) < 1$. For instance, given the clipped Laplace mechanism $\mathcal{M}_I$ that adds noise to a data point $x \in \mathbb{R}$ following $\mathrm{Lap}(\frac{M}{\tau})$, where

$$\tau = \varepsilon \frac{4(\frac{1}{\rho} - m + 2)}{m^2 + 4(\frac{1}{\rho} - m + 2)}, \tag{2}$$

we obtain an ε-BDP mechanism.

Yang et al. [4] show that for Gaussian Markov random fields, a clipped $\mathcal{M}_{\varepsilon,f}$ satisfies $(nM\varepsilon)$-BDP, matching the general bound. Theorem IV.1 covers this case, since such fields are multivariate Gaussians. Our bound is strictly tighter whenever it improves on the general bound.

V. Markov Chain Correlation Model

In this section we investigate the impact of correlations following a Markov model on the privacy leakage and utility of BDP mechanisms. We adopt the definition of a Markov chain from [9], which specifically refers to finite, time-homogeneous Markov chains, i.e., those with finite state spaces and time-invariant transition probabilities. Under this definition we obtain the following result:

Theorem V.1 (The Markov Chain Bound)**.** *Let $\mathcal{M} : \mathcal{S}^n \to \mathcal{Y}$ be an ε-DP mechanism. Let the databases follow a Markov chain with transition matrix $P \in \mathbb{R}^{s\times s}$ and initial distribution $w \in \mathbb{R}^s$ with the following properties:*

(H1) *For all $x, y \in \mathcal{S}$ we have $P_{x,y} > 0$ and,*
(H2) *$wP = w$.*

Then, $\mathcal{M}$ is an $(\varepsilon + 4\ln\gamma)$-BDP mechanism where

$$\gamma := \frac{\max_{x,y\in\mathcal{S}} P_{xy}}{\min_{x,y\in\mathcal{S}} P_{xy}}.$$

(H1) requires all transition probabilities to be strictly positive, ensuring that the Markov chain is irreducible and aperiodic, and thus admits a unique stationary distribution w [9]. (H2) assumes the initial distribution equals this stationary one, a standard simplification in applications such as weather forecasting or electricity demand modeling [1], which is asymptotically satisfied as any initial distribution converges geometrically fast to w.

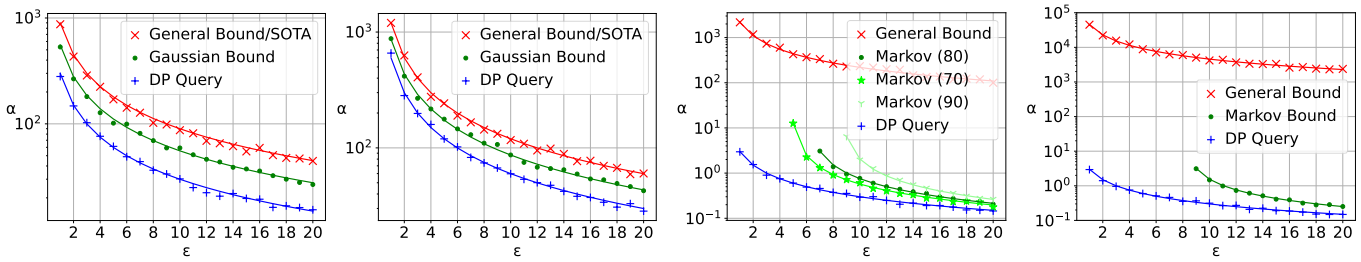
While prior work provides a mechanism for BDP protection of lazy binary Markov chains with a symmetric transition matrix [6], we present the first direct and general relationship between DP and BDP leakage for arbitrary Markov chains, including non-binary ones. Theorem V.1 enables the systematic design of BDP mechanisms by adjusting the noise of an existing DP mechanism: $\varepsilon' = \varepsilon - 4\ln(\gamma)$. Note that the best achievable BDPL using Theorem V.1 is $\varepsilon = 4\ln(\gamma)$, since $\varepsilon' \geq 0$. Consequently, the minimum achievable ε is fundamentally constrained by the structure of the underlying Markov model, specifically by the maximum transition ratio γ. We provide both theoretical and empirical analyses of the effect of this constraint in Figure 2 and Section VI.

VI. Utility Analysis

In this section, we present both theoretical and empirical evidence demonstrating the advantages of our novel bounds for Laplace noise calibration in sum queries. Our theoretical analysis is database-agnostic and framed in terms of (α, β)-accuracy (proofs in [1]). Empirically, we measure utility using the upper bound of a 95% confidence interval $(1-\beta)$ for the absolute query error, which serves as a practical analogue to α. Smaller upper bounds indicate higher utility, while close alignment with the theoretical α confirms strong agreement between theory and experiments. We evaluate our mechanisms on four real-world datasets, Galton and FamilyIQ for the Gaussian model and Electricity and Activity for the Markov one (see [1] for details).

Figure 1 presents the results including our Gaussian/Markov-specific bounds, the state-of-the-art BDP bounds, and the DP Laplace mechanism for sum queries. We plot the DP mechanism as the baseline for the best possible utility; however, it is important to note that DP does not offer meaningful protection in this experiment, given correlation. Among the correlation-protecting mechanisms, those that use our novel bound consistently outperform the s-o-t-a mechanism [4] for all ε in all datasets. Note that we plot all results on a logarithmic scale. This makes it harder to visually see the substantial reduction of error achieved by our mechanisms–particularly for small values of ε. For instance, for $\varepsilon = 1$ the error is reduced by more than 400 units for both IQ datasets and 200 inches for the Galton, and in the Activity dataset the general bound results in a 10^5 times larger error than that of our proposed Markov chain bound. This is because the general bound scales with the size of the database n, while our Markov bound is independent of n, highlighting its huge benefit for large datasets.

The only previous BDP mechanism for Markov chains [6], focuses on lazy binary Markov models with a symmetric transition matrix, i.e., the probability of staying in the same state P_{ss}–self-transition probability–is constant for $s \in \{0, 1\}$, and $P_{ss} > 0.5$. In this regime, $\gamma = \frac{P_{ss}}{1-P_{ss}}$. Since we did not



(a) Galton, $n = 897$ $m = 3$ (b) FamilyIQ, $n = 868$, $m = 2$. (c) Electricity, $n = 731$. (d) Activity, $n = 17\,568$.

Figure 1. Utility results. Lines show theoretical error at $\beta = 5\%$ and markers indicate empirical 95% upper bounds.

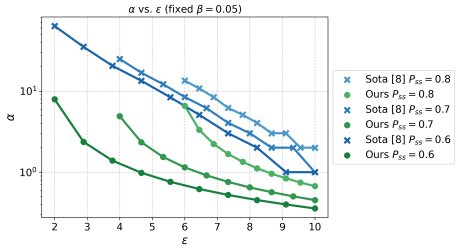


Figure 2. Accuracy of our mechanism vs. the one proposed in [6] for $n = 700$ and various self-transition probabilities P_{ss}.

find any real dataset fulfilling these requirements, we compute only formal utility results finding that our mechanism achieves superior (α, β)-accuracy, as shown in Figure 2. It is important to note that while their mechanism supports arbitrary BDPL, ours applies only for $\varepsilon \geq 4\ln(\gamma)$, which in Electricity 70 kWh yields the most favorable case with a minimum $\varepsilon = 4.9$, while 90 kWh imposes the weakest bound with a minimum $\varepsilon = 8.45$. However, our approach generalizes to arbitrary Markov chains, whereas theirs is limited to lazy, symmetric binary models. In the intersection of both applicability domains, our use of Laplace-based recalibration yields improved utility. While the error of mechanisms based on the general bound increases sharply, the error of both the Markov chain–based mechanism and the standard DP mechanism remains stable.

VII. Conclusion

We studied the utility of BDP mechanisms for correlated data, addressing prior limitations by analyzing broader correlation models and providing a detailed exploration of privacy-utility trade-offs. We established new connections between DP and BDP, showing that any ε-DP mechanism satisfies $m\varepsilon$-BDP for a correlated group of size m, and that this bound is tight. We improved upon it for multivariate Gaussian and Markov models, deriving correlation-specific BDP bounds that offer stronger utility than state-of-the-art approaches. In particular, our Markov-based bounds remain robust regardless of dataset size, unlike general bounds that scale poorly with the number of correlated records.

While BDP without a specific correlation model remains impractical, our results show that better utility is achievable without weakening the adversary model in scenarios where correlations are limited, Gaussian, or follow a Markov chain with sufficiently similar transitions. Overall, our Theorems III.1, IV.1 and V.1 advance both the theoretical and practical understanding of BDP, enabling the reuse of DP mechanisms in correlated settings and paving the way for more accurate, correlation-aware privacy mechanisms.

Acknowledgments

Work funded by Helmholtz Association (HGF) Topic Engineering Secure Systems, KASTEL SRL, and Germany's Excellence Strategy (EXC 2050/1 'CeTI'; ID 390696704). J.P-A. is a "Ramón y Cajal" fellow (RYC2021-034256-I) funded by NextGenerationEU/PRTR and MCIN/AEI/10.13039/501100011033.

References

[1] M. Lange, P. Guerra-Balboa, J. Parra-Arnau, and T. Strufe, "Balancing privacy and utility in correlated data: A study of bayesian differential privacy," *Proc. VLDB Endow.*, 2025.
[2] C. Dwork and A. Roth, *The Algorithmic Foundations of Differential Privacy*, ser. Foundations and Trends in Theoretical Computer Science, 2014.
[3] D. Kifer and A. Machanavajjhala, "Pufferfish: A framework for mathematical privacy definitions," *ACM Trans. Database Syst.*, pp. 3:1–3:36, 2014.
[4] B. Yang, I. Sato, and H. Nakagawa, "Bayesian differential privacy on correlated data," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 2015.
[5] T. Humphries, S. Oya, L. Tulloch, M. Rafuse, I. Goldberg, U. Hengartner, and F. Kerschbaum, "Investigating membership inference attacks under data dependencies," in *IEEE Computer Security Foundations Symposium (CSF)*, 2023.
[6] D. Chakrabarti, J. Gao, A. Saraf, G. Schoenebeck, and F.-Y. Yu, "Optimal local bayesian differential privacy over markov chains," 2022.
[7] N. Almadhoun, E. Ayday, and Ö. Ulusoy, "Differential privacy under dependent tuples—the case of genomic privacy," *Bioinformatics*, 2020.
[8] J. Shao, *Mathematical Statistics*, 2003.
[9] D. A. Levin and Y. Peres, *Markov chains and mixing times*. American Mathematical Soc., 2017.

Comparadores Lexicográficos en el Modelo R1CS

Marc Guzmán-Albiol*, Marta Bellés-Muñoz†‡, Carla Brugulat-Rica*, Rafael Genés-Durán*, y Jose L. Muñoz-Tapia*
*Departamento de Ingeniería Telemática, Universidad Politécnica de Cataluña (UPC), 08034 Barcelona, España
†Departamento de Tecnologías de la Información y las Comunicaciones,
Universitat Pompeu Fabra (UPF), 08018 Barcelona, España
‡Dusk Network, 1069 CD Ámsterdam, Países Bajos
Correos electrónicos: {marc.guzman.albiol, rafael.genes, carla.brugulat, jose.luis.munoz}@upc.edu, marta.belles@upf.edu

***Resumen*—Comparar dos números puede parecer una tarea trivial, pero trasladar esa operación al modelo algebraico de restricciones R1CS plantea retos interesantes. Este trabajo presenta una formulación del algoritmo de comparación lexicográfica bit a bit adaptada a dicho modelo, que permite determinar si un valor representado como un vector binario es menor que una constante fija mediante ecuaciones de rango 1 sobre un cuerpo finito. La construcción se implementa en el lenguaje `circom` y produce un circuito con 759 restricciones no lineales. Además, el trabajo ofrece un ejemplo concreto de cómo traducir cálculos no triviales al marco R1CS.**

***Index Terms*—Computación Verificable, Optimización de Restricciones, R1CS, Comparadores Binarios**

I. Introducción

Los protocolos de *computación verificable* permiten que una entidad (el *prover*) convenza a otra (el *verifier*) de que un cálculo se ha realizado correctamente sin necesidad de repetirlo [1], [2], [3]. Estos protocolos resultan esenciales en entornos con confianza limitada, como la computación en la nube o los sistemas descentralizados [4], [5], [6].

Los protocolos modernos, en particular los *SNARKs* (Scalable Non-Interactive Arguments of Knowledge), logran una verificación eficiente mediante criptografía de curvas elípticas y *pairings* bilineales [7], [8], [9]. En este contexto, los cálculos deben representarse sobre un cuerpo primo $\mathbb{F}_p$, y asegurar que todos los valores sean menores que p es esencial para evitar errores de *overflow* que pueden comprometer la solidez criptográfica del sistema [10], [11], [12].

Verificar si un valor t satisface la desigualdad $t > K$ es una operación trivial en la computación convencional, pero no en los SNARKs, donde los cálculos deben expresarse mediante un conjunto de ecuaciones cuadráticas conocido como sistema de restricciones de rango uno (R1CS, por sus siglas en inglés) [13]. Aunque el modelo R1CS puede representar cualquier cálculo determinista, traducir operaciones de comparación o estructuras de control condicional es complejo.

Este trabajo presenta una formulación del algoritmo clásico de comparación lexicográfica bit a bit adaptada al modelo R1CS. La construcción, implementada en `circom` [14], [15], produce un circuito de 759 restricciones no lineales y sirve como ejemplo de cómo expresar operaciones no triviales en este marco algebraico.

En la Sección II introducimos los preliminares necesarios sobre el modelo R1CS y el lenguaje `circom`. En la Sección III presentamos la construcción del comparador lexicográfico en R1CS y analizamos su implementación. Finalmente, en la Sección IV discutimos los resultados y conclusiones del trabajo.

II. Preliminares

En esta sección se introducen los conceptos necesarios sobre el modelo R1CS y el lenguaje `circom`, empleado para implementar el comparador propuesto.

II-A. Sistemas de Restricciones de Rango 1 (R1CS)

Fijamos un cuerpo primo $\mathbb{F}_p$ de 254 bits y un conjunto de señales $s = \{s_1, \ldots, s_n\}$ que representan las variables de un cálculo. Cada restricción de rango 1 adopta la forma

$$(A_1 s_1 + \cdots + A_n s_n) \times (B_1 s_1 + \cdots + B_n s_n) - (C_1 s_1 + \cdots + C_n s_n) = 0 \quad (1)$$

donde $A_i, B_i, C_i \in \mathbb{F}_p$ son constantes fijas. Un *sistema R1CS* es una colección de estas ecuaciones sobre el mismo conjunto de señales, y un *testigo* es una asignación de valores que las satisface todas simultáneamente. Para incluir términos independientes, se introduce una señal constante $s_0 = 1$ de modo que cualquier escalar $N \in \mathbb{F}_p$ pueda representarse como $N \cdot s_0$.

El modelo R1CS es completamente expresivo: cualquier cálculo determinista puede representarse mediante restricciones de este tipo, introduciendo variables auxiliares que descompongan operaciones complejas [13]. Sin embargo, la representación no es única, y el número de variables o restricciones influye directamente en la eficiencia de la generación y verificación de las pruebas [7].

Ciertas estructuras de control, como condicionales o bucles, carecen de soporte directo en el modelo R1CS y deben emularse mediante relaciones algebraicas. Por ejemplo, para imponer que una señal de salida r valga 1 si una entrada t es cero y 0 en caso contrario, puede utilizarse el siguiente par de restricciones:

$$t \cdot u = 1 - r, \qquad t \cdot r = 0, \quad (2)$$

donde u es una variable auxiliar que representa el inverso multiplicativo de t en $\mathbb{F}_p$.

II-B. Circom

La generación manual de sistemas R1CS de gran tamaño resulta inviable en la práctica, por lo que se recurre a lenguajes de alto nivel como `circom` [15], [14]. `circom` es un *domain-specific language* diseñado para describir cálculos verificables. Una de sus principales características es el desacoplamiento entre la definición del sistema de restricciones y la generación del testigo, lo que permite especificar las relaciones algebraicas de forma declarativa y delegar el

cálculo de los valores concretos de las señales a una fase separada. Los programas en `circom` se estructuran de forma modular, componiendo plantillas y utilizando bucles `for` que el compilador desenrolla completamente en tiempo de compilación.

El lenguaje distingue tres tipos de asignación, cada uno con un efecto distinto sobre el sistema de restricciones y la generación del testigo:

- `x <- expr`: asignación no restringida. La expresión `expr` se evalúa únicamente durante la generación del testigo y no introduce ninguna restricción en el sistema R1CS. La correspondiente ecuación debe definirse explícitamente mediante uno de los dos operadores descritos a continuación.
- `x <== expr`: asignación restringida. Además de asignar el valor al testigo, introduce una restricción que obliga a que `x` satisfaga la igualdad `x = expr` en el modelo R1CS;
- `a === b`: igualdad estricta. Impone directamente la restricción `a = b` sin modificar el valor de ninguna señal durante la etapa de generación del testigo.

Como ejemplo, la plantilla `IsZero` devuelve 1 si la entrada es 0 y 0 en otro caso, usando dos las restricciones de la Eq. 2.

```
template IsZero() {
    signal input t;
    signal output r;
    signal u;

    u <-- t!=0 ? 1/t : 0;
    r <== -t*u +1;
    t*r === 0;
}
```

Listado 1: La plantilla `IsZero()` devuelve 1 si la entrada es 0, y 0 en caso contrario.

La línea `u <- t!=0 ? 1/t : 0` calcula de forma no restringida el inverso de t para simplificar el testigo, evitando expresar explícitamente la operación de inversión en R1CS. Sin embargo, las dos restricciones siguientes garantizan que el valor asignado a `u` sea el correcto.

III. Comparaciones Lexicográficas en R1CS

Las comparaciones entre números, como la relación menor que, son frecuentes en la computación de propósito general. No obstante, expresarlas en el modelo R1CS resulta no trivial, ya que este carece de soporte nativo para desigualdades. En consecuencia, tales operaciones deben implementarse mediante técnicas algebraicas que traduzcan la lógica de comparación en restricciones de rango uno, propias del modelo R1CS. En esta sección se describe un método para traducir el algoritmo clásico de comparación lexicográfica bit a bit al lenguaje de restricciones R1CS. Este procedimiento permite codificar en el modelo R1CS una comparación estricta de la forma $t < K$, donde K es una constante conocida y t representa un valor entero de entrada.

A partir de este punto representamos tanto t como la constante K mediante vectores binarios de 254 bits en formato *little-endian*

$$t = (t_0, t_1, \ldots, t_{253})_2 := \sum_{i=0}^{253} t_i \cdot 2^i, \quad \text{y} \tag{3}$$

$$K = (K_0, K_1, \ldots, K_{253})_2 := \sum_{i=0}^{253} K_i \cdot 2^i, \tag{4}$$

donde cada bit $t_i, K_i \in \{0, 1\}$ para todo i. La constante K es conocida previamente y se interpreta como un elemento del cuerpo $\mathbb{F}_p$, mientras que t se obtiene a partir de su descomposición binaria y puede, en principio, situarse fuera del rango del cuerpo, siempre que quepa en 254 bits.

Nuestro objetivo consiste en construir una representación en R1CS lo más compacta posible que permita comparar la entrada binaria t con la constante fija K y producir la salida correcta. La Figura 1 ilustra esta operación, en la que la salida adopta el valor 1 cuando $t > K$ y el valor 0 en caso contrario.

Una primera aproximación consiste en definir un polinomio $r \in \mathbb{F}_p[x]$ tal que se cumpla $r(x) = 0$ para todo $x \in 0, \ldots, K$ y $r(x) = 1$ para todo $x \in K+1, \ldots, p-1$. Dicho polinomio existe y puede obtenerse mediante interpolación al fijar la salida deseada para cada posible entrada en $\mathbb{F}_p$. Para verificar si un valor t satisface la condición $t \leq K$, bastaría con evaluar $r(t)$ y comprobar si el resultado es cero. No obstante, este procedimiento no es práctico cuando se trabaja con cuerpos grandes. En particular, si p tiene 254 bits, la interpolación sobre todos los valores posibles conduce a un polinomio de grado cercano a 2^{254}. La traducción de un polinomio de este grado a R1CS requeriría un número descomunal de variables intermedias y restricciones cuadráticas, lo que convierte a este método en inviable en la práctica.

El algoritmo más trivial para determinar cuál de dos números binarios es mayor consiste en identificar el bit más significativo en el que difieren y comprobar cuál de ellos contiene un 1 en esa posición, ya que ese número es el mayor. Nos referimos a este método como comparación lexicográfica, ya que sigue el orden de los bits empezando por el más significativo. En la computación convencional, este enfoque puede implementarse de manera directa mediante un bucle que recorre los bits de más a menos significativo y se detiene en el primer bit diferente, donde se establece el resultado de la comparación. No obstante, adaptar este método al modelo R1CS no es tan inmediato, como mostraremos a continuación.

Para obtener la formulación en R1CS de este algoritmo, comenzamos definiendo un vector de señales intermedias que indiquen si t y K coinciden en el bit i, es decir, si $t_i = K_i$. Denotamos cada una de estas señales como δ_i, y para cada posición i se cumple lo siguiente:

$$\delta_i = \begin{cases} 1, & \text{si } t_i = K_i, \\ 0, & \text{en otro caso.} \end{cases} \tag{5}$$

En el lenguaje `circom`, la comprobación de igualdad entre los bits t_i y K_i puede implementarse mediante la plantilla `IsZero`, descrita en la Sección II-B. En particular, basta con invocar esta plantilla dentro de un bucle y utilizar como entrada la diferencia $t_i - K_i$, de modo que la salida indique si ambos bits son iguales.

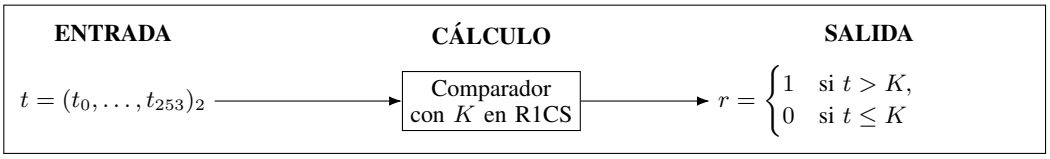


Figura 1. Diagrama del proceso de comparación entre t y K en R1CS. Se devuelve 1 si $t > K$ y 0 en caso contrario.

```
component z[NUM_BITS];

for (var i = 1; i<=NUM_BITS-1; i++) {
    Ki <-- (K >> i) & 1; // Bit actual de K
    delta[i] = IsZero();
    delta[i].in <== t[i] - Ki; // Restringe
     ↪  el valor de delta_i
}
```

Listado 2: Forzado de las señales $z\delta_i$ con `circom`.

Cada instancia de `IsZero()` establece que $z_i = 1$ cuando $t_i = K_i$ y $z_i = 0$ en caso contrario, almacenando el resultado en `z[i].out`.

El siguiente paso consiste en imponer en R1CS la lógica de decisión que asegura que el bit utilizado para resolver la comparación sea el más significativo en el que t y K difieren. Para ello se comienza con el bit más significativo de ambos números. Si los valores en esa posición no coinciden, el resultado de la comparación se determina directamente a partir de ellos. En cambio, si son iguales, la comparación se traslada al siguiente bit menos significativo, y el proceso continúa de forma sucesiva.

Para formalizar esta lógica introducimos una restricción sobre las señales δ_i de la forma

$$r = (1-\delta_{253})\cdot\gamma_i + \delta_{253}\cdot(\text{restricciones sobre el siguiente bit}),$$

donde γ_i representa una señal intermedia que captura el resultado de la comparación en el bit i:

$$\gamma_i = \begin{cases} 1, & \text{si } t_i > K_i, \\ 0, & \text{en otro caso.} \end{cases} \tag{6}$$

Dado que trabajamos con valores binarios, la definición de γ_i puede escribirse en forma algebraica como:

$$\gamma_i = t_i \cdot (1 - K_i). \tag{7}$$

La misma construcción se aplica a todas las posiciones de bit, lo que permite establecer una relación recursiva de restricciones:

$$r_i = (1 - \delta_i) \cdot \gamma_i + \delta_i \cdot r_{i-1}, \quad i \in \{253, \ldots, 1\}. \tag{8}$$

Podemos reescribir la expresión (8) en forma R1CS como:

$$0 = \delta_i \cdot (r_{i-1} - \gamma_i) - (r_i - \gamma_i). \tag{9}$$

Esta ecuación garantiza que la señal r_i adopte el valor del resultado parcial anterior r_{i-1} solo cuando los bits coinciden (es decir, $\delta_i = 1$), y el de la comparación actual γ_i en caso contrario.

Podemos simplificar la ecuación (9) observando que, por definición, $\delta_i = 1$ únicamente cuando $t_i = K_i$, lo que implica que $\gamma_i = t_i(1 - K_i) = 0$. En consecuencia, el producto $\delta_i \cdot \gamma_i$ es siempre nulo y el término correspondiente puede eliminarse sin alterar el comportamiento del circuito. La restricción puede expresarse así en su forma simplificada:

$$0 = \delta_i \cdot r_{i-1} - (r_i - \gamma_i), \tag{10}$$

lo que reduce la expresión a su versión más compacta sin pérdida de corrección lógica.

El caso base de la recursión se establece directamente en el bit menos significativo, donde p_0 viene dado por la comparación entre t_0 y K_0:

$$r_0 = \gamma_0 = t_0 \cdot (1 - K_0). \tag{11}$$

La recurrencia culmina en el bit más significativo, de modo que r_{253} representa la salida de toda la comparación:

$$r = r_{253}. \tag{12}$$

De esta manera, el valor r condensa la información de todos los pasos intermedios y refleja correctamente el resultado de la comparación lexicográfica.

El Listado 3 presenta el bucle principal que traduce de forma directa el conjunto de restricciones previamente definido al lenguaje `circom`.

```
Ki <-- (K >> 0) & 1;
r[0] <== t[0]*(1 - Ki);

for (var i = 1; i<=NUM_BITS-1; i++) {
  Ki <-- (K >> i) & 1;
  delta[i] = IsZero();
  delta[i].in <== t[i] - Ki;
  gamma[i] <== t[i]*(1 - Ki);
  r[i] <-- (1 - z[i].out)*gamma[i] +
   ↪  delta[i].out*r[i-1]; // Ec. (1)
  0 === z[i].out * r[i-1] - (r[i] -
   ↪  gamma[i]); // Ec. (2)
}
```

Listado 3: Bucle principal del enfoque lexicográfico en `circom`.

Pasamos ahora a evaluar la eficiencia de este enfoque en función del número de restricciones R1CS necesarias. La relación recursiva definida para los r_i en la Ecuación 8 introduce una restricción por cada bit en el rango $i \in \{253, \ldots, 1\}$, lo que supone un total de 253 restricciones. A ello se suman las restricciones asociadas al cálculo de los indicadores de igualdad δ_i en cada una de las 253 posiciones. Como se indicó en la Sección II-A, cada comprobación de igualdad requiere dos restricciones, lo que añade 506 restricciones adicionales. El número total asciende así a $253 + 506 = 759$.

En la definición del conjunto de restricciones también aparecen expresiones como $\gamma_i = t_i \cdot (1 - K_i)$ o $\delta_i.\texttt{in} = t_i - K_i$, que pueden observarse en los listados de `circom`.

```
gamma[i] <== t[i] * (1 - Ki);
delta[i].in <== t[i] - Ki;
```

Listado 4: Ejemplo de expresiones lineales.

Estas expresiones, al ser lineales, no incrementan el número de restricciones efectivas, ya que el compilador de `circom` las simplifica automáticamente. En consecuencia, el número total de restricciones R1CS necesarias para implementar la comparación lexicográfica en este modelo se mantiene en 759.

IV. Discusión

Comparar un valor representado en bits con una constante fija es una operación elemental en la mayoría de entornos de programación, pero trasladarla al modelo R1CS introduce una serie de limitaciones que obligan a replantear su implementación desde una perspectiva algebraica. En este contexto, el desarrollador no dispone de estructuras de control convencionales, como condicionales o ramificaciones, y debe expresar toda la lógica del algoritmo mediante ecuaciones de rango 1 sobre un cuerpo finito.

El enfoque lexicográfico propuesto reproduce fielmente la lógica tradicional: examinar los bits de mayor a menor peso y determinar el resultado de la comparación en el primer punto de diferencia. Sin embargo, al llevar esta idea al marco R1CS, cada paso debe representarse con variables auxiliares y restricciones que simulen, de forma puramente algebraica, tanto la comprobación de igualdad entre bits como la propagación del resultado parcial de la comparación. Esta traducción genera una estructura recursiva en la que cada bit añade un pequeño bloque de restricciones que dependen del anterior.

El circuito resultante requiere un total de 759 restricciones no lineales, de las cuales la mayoría provienen de las comprobaciones de igualdad necesarias para preservar la coherencia del orden lexicográfico. Aunque esta cifra puede parecer elevada, el procedimiento ofrece una formulación correcta y verificable del algoritmo clásico dentro del modelo R1CS, y constituye un punto de partida sólido para futuras optimizaciones.

Más allá del caso concreto del comparador, el diseño presentado ilustra una metodología general para trasladar operaciones no triviales al marco de las pruebas de conocimiento cero. Este tipo de transformaciones representa uno de los desafíos centrales en la construcción de protocolos eficientes y seguros. Por ello, el comparador desarrollado no solo resuelve una necesidad práctica, sino que también sirve como ejemplo didáctico de cómo adaptar algoritmos tradicionales a un modelo algebraico estrictamente determinista como R1CS.

Agradecimientos

Esta investigación fue financiada por el Instituto Nacional de Ciberseguridad (INCIBE) a través de la Cátedra Carismática INCIBE-UPC y por el proyecto español "Verifiable Computations and Applications" PID2024-159480OB-I00.

Referencias

[1] L. Babai, L. Fortnow, L. A. Levin, and M. Szegedy, "Checking computations in polylogarithmic time," in *Proceedings of the Twenty-Third Annual ACM Symposium on Theory of Computing*, ser. STOC '91. New York, NY, USA: Association for Computing Machinery, 1991, p. 21–32. [Online]. Available: https://doi.org/10.1145/103418.103428

[2] S. Micali, "CS proofs (extended abstracts)," in *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*. IEEE Computer Society, 1994, pp. 436–453. [Online]. Available: https://doi.org/10.1109/SFCS.1994.365746

[3] S. Goldwasser, Y. T. Kalai, and G. N. Rothblum, "Delegating computation: Interactive proofs for muggles," *J. ACM*, vol. 62, no. 4, Sep. 2015. [Online]. Available: https://doi.org/10.1145/2699436

[4] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in *Advances in Cryptology – CRYPTO 2010*, T. Rabin, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 465–482.

[5] R. Canetti, B. Riva, and G. N. Rothblum, "Practical delegation of computation using multiple servers," in *Proceedings of the 18th ACM Conference on Computer and Communications Security*, ser. CCS '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 445–454. [Online]. Available: https://doi.org/10.1145/2046707.2046759

[6] M. Nabi, S. Avizheh, and R. Safavi-Naini, "Fides: A system for verifiable computation using smart contracts," in *Financial Cryptography and Data Security. FC 2022 International Workshops*, S. Matsuo, L. Gudgeon, A. Klages-Mundt, D. Perez Hernandez, S. Werner, T. Haines, A. Essex, A. Bracciali, and M. Sala, Eds. Cham: Springer International Publishing, 2023, pp. 448–480.

[7] J. Groth, "On the size of pairing-based non-interactive arguments," in *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II*, ser. Lecture Notes in Computer Science, M. Fischlin and J. Coron, Eds., vol. 9666. Springer, 2016, pp. 305–326.

[8] A. Gabizon, Z. J. Williamson, and O. Ciobotaru, "PlonK: Permutations over Lagrange-bases for oecumenical noninteractive arguments of knowledge," Cryptology ePrint archive, paper 2019/953, 2019. [Online]. Available: https://eprint.iacr/2019/953

[9] D. F. Aranha, Y. E. Housni, and A. Guillevic, "A survey of elliptic curves for proof systems," Cryptology ePrint Archive, Paper 2022/586, 2022.

[10] V. I. 0xPARC Community, Ethereum Foundation, "Auditing report for `circom-bigint`," October 2022, https://veridise.com/wp-content/uploads/2023/02/VAR-circom-bigint.pdf.

[11] S. Pailoor, Y. Chen, F. Wang, C. Rodríguez, J. Van Geffen, J. Morton, M. Chu, B. Gu, Y. Feng, and I. Dillig, "Automated detection of under-constrained circuits in zero-knowledge proofs," *Proc. ACM Program. Lang.*, vol. 7, no. PLDI, Jun. 2023. [Online]. Available: https://doi.org/10.1145/3591282

[12] H. Wen, J. Stephens, Y. Chen, K. Ferles, S. Pailoor, K. Charbonnet, I. Dillig, and Y. Feng, "Practical security analysis of Zero-Knowledge proof circuits," in *33rd USENIX Security Symposium (USENIX Security 24)*. Philadelphia, PA: USENIX Association, Aug. 2024, pp. 1471–1487. [Online]. Available: https://www.usenix.org/conference/usenixsecurity24/presentation/wen

[13] R. Gennaro, C. Gentry, B. Parno, and M. Raykova, "Quadratic span programs and succinct NIZKs without PCPs," in *Advances in Cryptology – EUROCRYPT 2013*, T. Johansson and P. Q. Nguyen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 626–645.

[14] M. Bellés-Muñoz, J. Baylina, V. Daza, and J. L. Muñoz-Tapia, "New privacy practices for blockchain software," *IEEE Software*, vol. 39, no. 3, pp. 43–49, 2022.

[15] M. Bellés-Muñoz, M. Isabel, J. L. Muñoz-Tapia, A. Rubio, and J. Baylina, "Circom: A circuit description language for building zero-knowledge applications," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 6, pp. 4733–4751, 2022.

Un resumen de "Meta-feature-based Data Preprocessing for Machine Learning through Correlation Structures and Cost-aware Benchmarking: A Case Study in Cybersecurity"

Noemí DeCastro-García
Departamento de Matemáticas. Universidad de León
Campus de Vegazana s/n 24007 León
ncasg@unileon.es

David Escudero García
Research Institute of Applied Sciences in Cybersecurity (RIASC)
Universidad de León. Campus de Vegazana s/n 24007 León
descg@unileon.es

***Resumen*—Este trabajo presenta un enfoque de *meta-aprendizaje* para optimizar el preprocesamiento de datos en modelos de aprendizaje automático aplicados a la ciberseguridad. Se propone un marco basado en la correlación entre *meta-features* del conjunto de datos, y métricas de rendimiento de los modelos, considerando además el coste y el impacto de modificar dichas características. El estudio utiliza un conjunto de datos de direcciones IP maliciosas, evaluando tres algoritmos de aprendizaje automático, y cinco transformaciones de preprocesado de datos. Los resultados evidencian que el análisis de meta-características puede guiar de forma efectiva la selección de estrategias de preprocesamiento, mejorando significativamente el ajuste y la robustez de los modelos. De este modo, el marco propuesto permite determinar de forma empírica cuándo el margen de mejora de los modelos de aprendizaje automático mediante transformaciones u optimización es mínimo porque el conjunto de datos ha llegado a su máximo potencial.**

***Index Terms*—Meta-aprendizaje, meta-características, aprendizaje automático, ciberseguridad**

I. Introducción

El aprendizaje automático (ML, por sus siglas en inglés) se ha consolidado como una herramienta fundamental para la detección de amenazas, análisis de tráfico y clasificación de comportamientos anómalos en ciberseguridad. En este contexto, la selección de estrategias adecuadas de preprocesamiento de datos sigue siendo un desafío, ya que influye directamente en la precisión, estabilidad y capacidad de generalización de los modelos [1]. Este trabajo es un resumen del artículo [2], donde se aborda esta problemática mediante un enfoque de meta-aprendizaje o *meta-learning* (MtL, por sus siglas en inglés) orientado a identificar cómo las propiedades estadísticas y estructurales de los datos (meta-características o *meta-features*) se correlacionan con el rendimiento de los modelos de ML. Se plantea la hipótesis de que analizar estas correlaciones permite determinar qué transformaciones de preprocesamiento son más efectivas para un conjunto de datos determinado, optimizando el uso de recursos y mejorando la robustez del modelo ante perturbaciones.

El MtL es una subárea del ML que busca desarrollar modelos capaces de aprender de la experiencia previa, generalizando conocimiento obtenido de la aplicación de algoritmos de ML a diferentes conjuntos de datos. En lugar de centrarse en instancias individuales, el MtL analiza *meta-datos* que describen conjuntos de datos completos y su rendimiento frente a distintos algoritmos o configuraciones [3]. El proceso habitual en el uso de MtL consta de tres fases principales. Primero, se construye un *meta-dataset*, donde cada *meta-ejemplo* representa un conjunto de datos previamente utilizado en una tarea de ML. Las variables independientes son las *meta-features*, que describen las propiedades estructurales y estadísticas del dataset, mientras que las variables dependientes o *meta-targets* suelen ser métricas de rendimiento (por ejemplo, precisión o F_1-score). En segundo lugar, se entrena un modelo de meta-aprendizaje sobre este meta-dataset, buscando aprender relaciones entre las meta-características y el rendimiento observado. Este modelo puede emplearse para recomendar algoritmos, seleccionar hiperparámetros, o predecir el rendimiento esperado de un algoritmo en un nuevo conjunto de datos [1].

Las meta-características son el núcleo del MtL, ya que permiten caracterizar un conjunto de datos sin necesidad de ejecutar el proceso completo de entrenamiento. Pese a los avances en el estudio de las meta-features, la literatura existente presenta tres limitaciones principales. En primer lugar, la mayoría de los estudios se enfocan en la definición y clasificación de meta-características, pero no en su impacto empírico sobre la eficacia de transformaciones concretas de datos. En segundo lugar, muchos trabajos asumen condiciones experimentales ideales como datasets balanceados o particiones estándar 70/30, lo cual resulta poco realista en dominios como la ciberseguridad. Finalmente, el coste computacional asociado a la obtención y modificación de meta-características ha sido poco estudiado, aunque es un factor crítico para optimizar recursos en entornos operativos reales.

El presente trabajo aborda estas limitaciones proponiendo un marco empírico para analizar la correlación entre meta-características y métricas de rendimiento, incorporando una categorización por coste e impacto de modificación, y aplicándolo a un caso de estudio real en ciberseguridad. Además de contribuir a aumentar la eficacia de los sistemas de detección y defensa basados en inteligencia artificial, un resultado adicional de este estudio es que el análisis de las meta-features puede emplearse como una herramienta para estimar el potencial máximo de mejora que un conjunto de datos puede ofrecer antes de aplicar modelos de ML. Al estudiar las correlaciones entre las propiedades internas del

Tabla I
DISTRIBUCIÓN DE INSTANCIAS POR COMBINACIÓN DE ENTRENAMIENTO–PRUEBA (*ttc*).

ttc	Instancias entrenamiento (%)	Instancias prueba (%)
0	16 200 (38.66 %)	25 696 (61.34 %)
1	16 200 (21.88 %)	57 827 (78.11 %)
2	25 696 (61.34 %)	16 200 (38.66 %)
3	25 696 (30.76 %)	57 827 (69.23 %)
4	57 827 (78.11 %)	16 200 (21.88 %)
5	57 827 (69.23 %)	25 696 (30.76 %)

dataset y las métricas de rendimiento, es posible identificar cuándo las limitaciones del ajuste de los modelos provienen de la estructura o calidad de los datos y no del modelo empleado. Esta capacidad de diagnóstico previo resulta especialmente valiosa en ciberseguridad, donde la información disponible sobre amenazas o tráfico malicioso puede ser limitada o incompleta. Con ello, se evita invertir recursos en entrenamientos poco productivos y se promueve la toma de decisiones basada en evidencia sobre la necesidad de enriquecer o ampliar los datos antes de construir nuevos modelos de detección.

Este artículo se organiza de la siguiente manera: en la sección 2, se describen los materiales y métodos utilizados en la experimentación. En la sección 3, se desarrollan los resultados elementales del artículo original. Finalmente, se incluyen las conclusiones y referencias.

II. SECCIÓN EXPERIMENTAL

II-A. Conjuntos de datos

El estudio se realizó sobre un conjunto de datos real de 99 720 eventos asociados a direcciones IP, disponible en [4] clasificados según su nivel de maliciosidad de forma multiclase. El conjunto de datos original tiene 5 features base (FB), que contienen información de geolocalización y tiempo de las direcciones IP. A partir del dataset original se definieron dos grupos más de características, orientados a capturar distintos aspectos del comportamiento de las IPs. Por un lado, se calcularon features que introducen variables relacionadas con la estructura de red y la distribución de la maliciosidad (FN). Por otro lado, se calcularon features obtenidas a través del cálculo de frecuencias normalizadas de los valores de IP y país por clase de maliciosidad (FF). A partir de estos tres grupos descritos, que han sido extensamente estudiados en [5], [6], se generaron combinaciones que dan lugar a diferentes configuraciones de atributos: FB, FN, FF, $FB \cup FN$, $FB \cup FF$, $FN \cup FF$, $FB \cup FN \cup FF$. Cada una de estas siete configuraciones se combinó con seis proporciones (ttc) distintas de entrenamiento y prueba, derivadas de divisiones temporales del mes de registro en tres intervalos de diez días, véase *Tabla I*. El resultado final son 42 datasets (7 configuraciones de características $\times$ 6 proporciones ttc), lo que permite analizar cómo las variaciones en la estructura de los datos afectan a las meta-características y el rendimiento de los modelos.

II-B. Meta-features

Con el fin de estandarizar la descripción de las meta-features, en [3] se ha desarrollado una taxonomía que clasifica las meta-características en seis grandes grupos, de los que es posible calcular 84 meta-features mediante la herramienta desarrollada en [7]:

1. Generales: información básica del conjunto de datos (número de instancias, atributos, clases, proporciones, etc.).
2. Estadísticas: descriptores numéricos de los atributos, como media, desviación típica, curtosis, o rango.
3. Teóricas de la información: medidas derivadas de la entropía de Shannon, información mutua o correlación entre atributo y clase.
4. Basadas en modelos: propiedades de modelos simples (por ejemplo, profundidad de árboles de decisión o número de nodos).
5. De *landmarking*: rendimiento de clasificadores básicos usados como indicadores del comportamiento de algoritmos más complejos.
6. Otras: medidas de agrupamiento, complejidad o morfología del espacio de datos.

Una de las cuestiones innovadoras del trabajo [2] es la clasificación de las 84 meta-features de la taxonomía según dos dimensiones: coste de modificación, y grado de control sobre el impacto de la modificación. El coste de modificación mide la dificultad o el esfuerzo computacional necesario para alterar el valor de una meta-característica a través de transformaciones de datos. El nivel 1 corresponde a bajo coste; es decir, meta-características fácilmente modificables mediante operaciones simples de escalado o normalización. Por el contrario, el nivel 3 representa alto coste ya que implican alterar la estructura del conjunto de datos. La otra dimensión está enfocada a evaluar el control de impacto, que refleja el grado en que el efecto de una modificación puede predecirse o controlarse. En el nivel 1 se clasifican las meta-features consideradas de bajo control; es decir, en los que la modificación afecta de manera indirecta y poco predecible otras propiedades del dataset. En el caso de las meta-features clasificadas como de alto control o nivel 3, el impacto es claramente predecible, como en transformaciones que ajustan valores de media o varianza mediante escalado.

II-C. Técnicas de pre-procesado

Para comprobar cómo la modificación de ciertas meta-características influye en el rendimiento de los modelos, se aplicaron cinco técnicas de preprocesamiento de datos:

1. *Scaler (normalización)*: normalización de los valores de las características al rango $[-1, 1]$ mediante *min–max*.
2. *PCA (Análisis de Componentes Principales)*: reducción de dimensionalidad por componentes principales
3. *SelectKBest (selección de características)*: selección del 50 % de los atributos con mayor información mutua respecto a la variable objetivo.
4. *Corr Increase (aumento de correlación)*: aumento de correlación duplicando atributos con signo opuesto, generando una copia de cada atributo con signo opuesto ($X_i' = -X_i$) para incrementar artificialmente la correlación entre variables.
5. *Quantile Transform (transformación por cuantiles)*: transformación no lineal de atributos a distribución uniforme.

II-D. Análisis

La decisión sobre qué transformaciones de preprocesamiento aplicar y sobre qué meta-características actuar se basó en un análisis estadístico de correlaciones entre las meta-features y las métricas de rendimiento de los modelos de ML. Para cada uno de los 42 conjuntos de datos generados, se calcularon las métricas de desempeño de tres modelos representativos (AutoSklearn[1], GMM [8] y XGB[9]), incluyendo *Accuracy*, *F1-score*, *Coeficiente de Correlación de Matthews (MCC)*, *Sensibilidad* y *Especificidad*. Por otro lado, para cada uno de esos 42 conjuntos de datos, se calcularon 84 meta-características con [7]. A continuación, se evaluó la relación entre cada meta-característica y cada métrica de rendimiento mediante el coeficiente de correlación de Pearson. Se consideraron como relevantes aquellas correlaciones que resultaron estadísticamente significativas con todas las métricas con un nivel de significancia de $\alpha = 0{,}05$. El análisis de correlaciones entre las 84 meta-características y las métricas de rendimiento de los modelos permitió identificar, para cada algoritmo, las combinaciones de entrenamiento–prueba (*ttc*) con mayor número de correlaciones estadísticamente significativas y aquellas con coeficientes de correlación más altos. Este procedimiento permitió identificar las meta-características con mayor influencia sobre el rendimiento de los modelos, así como las combinaciones de entrenamiento–prueba (*ttc*) con mayor potencial de mejora mediante preprocesamiento. Finalmente, las transformaciones se aplicaron únicamente en los casos en los que las meta-características correladas que se habían obtenido eran de bajo o medio coste de modificación y se volvieron a calcular las métricas de rendimiento.

Finalmente, se incorporó la métrica de robustez *Rank Graduation Robustness (RGR)* del marco *SAFE AI* [10] para evaluar la estabilidad del modelo ante perturbaciones de entrada, aspecto clave en la detección de ciberataques.

III. Resultados y discusión

En la *Tabla II* se encuentra el resumen de correlaciones entre meta-características y métricas de rendimiento.

En conjunto, la decisión sobre en qué conjuntos de datos tiene más potencial aplicar transformaciones se basó en tres factores combinados: el porcentaje de meta-features correlacionadas, la magnitud media de la correlación, y la factibilidad de modificación según coste e impacto. En el caso de AutoSklearn, solo la partición ttc = 4 presentó meta-características significativamente correlacionadas, con una correlación media alta. Para GMM, se observaron correlaciones en varias particiones. Dado que muchas de estas meta-características también eran de tipo estadístico y de información teórica, con bajo coste y impacto predecible, se hipotetizó que las transformaciones de preprocesado podrían ayudar particularmente en particiones con menor proporción de entrenamiento (0–1), donde existía mayor margen de mejora. En XGB, el número de correlaciones significativas fue menor. Tras aplicar las transformaciones de preprocesado de datos, se volvieron a calcular las métricas de rendimiento. Los resultados elementales sobre las ganancias obtenidas se incluyen en la *Tabla III*.

Los resultados revelaron que las transformaciones de tipo *Quantile Transform* y *Scaler* produjeron mejoras estadísticamente significativas en la mayoría de las métricas. Por un lado, el modelo GMM mostró los mayores incrementos de rendimiento, superando incluso los mejores resultados originales alcanzados por XGB. El algoritmo XGB presentó mejoras marginales, indicando haber alcanzado su límite de desempeño en este conjunto de datos. Las mejoras más notorias se observaron en la métrica MCC, especialmente en configuraciones con proporciones de entrenamiento reducidas.

Además, el análisis de robustez SAFE mostró que las transformaciones *Scaler* y *Quantile Transform* incrementaron el valor promedio de RGR en los modelos GMM, indicando una mayor resiliencia frente a perturbaciones de entrada, propiedad deseable en sistemas de detección de amenazas cibernéticas.

Desde la perspectiva de la ciberseguridad, los resultados del estudio presentan varias implicaciones relevantes, tanto a nivel técnico como operativo. En primer lugar, el análisis de meta-características aplicado a datos de tráfico IP permite identificar qué transformaciones de preprocesamiento mejoran el rendimiento de los modelos de ML. En segundo lugar, el enfoque basado en MtL resulta especialmente útil en contextos con conjuntos de datos desbalanceados o limitados, ya que permitió mejorar el rendimiento incluso en configuraciones con escaso entrenamiento, lo que sugiere su aplicabilidad en la detección temprana de nuevas amenazas o en sistemas con datos incompletos.

Además, un análisis más en profundidad de los resultados permite seleccionar las transformaciones de datos más adecuadas según la naturaleza de las meta-features correlacionadas. Cuando las meta-características más correlacionadas eran de tipo estadístico, son mejores las transformaciones como Scaler o Quantile Transform. Si las correlaciones se concentraban en meta-features informacionales, suele resultar mejor SelectKBest.

IV. Conclusiones

El trabajo demuestra que el análisis de meta-características es una herramienta eficaz para guiar el preprocesamiento de datos en sistemas de ML orientados a ciberseguridad. En el contexto de la ciberseguridad, la capacidad de identificar los límites informativos de un conjunto de datos adquiere un valor estratégico. Los sistemas de detección y respuesta ante incidentes suelen operar con datos incompletos, ruidosos o fuertemente desbalanceados entre eventos benignos y maliciosos. Poder determinar cuándo un dataset ha alcanzado su máximo potencial predictivo permite evitar la sobreoptimización de modelos de ML que no pueden mejorar debido a la falta de información discriminativa.

El enfoque propuesto permite reducir el coste computacional del preprocesamiento al identificar transformaciones con mayor relación beneficio/coste. Además, aumenta la precisión y estabilidad de los modelos, especialmente en escenarios con conjuntos de datos desbalanceados o limitados, contribuyendo al desarrollo de IA confiable y explicable.

Agradecimientos

Esta investigación resulta del Proyecto Estratégico *Ciencia de datos para un modelo de inteligencia artificial en ciberseguridad* (C073/23), fruto del convenio de colaboración suscrito entre el Instituto Nacional de Ciberseguridad (INCIBE) y la Universidad de León. Esta iniciativa se realiza en el marco

Tabla II
RESUMEN DE CORRELACIONES ENTRE META-CARACTERÍSTICAS Y MÉTRICAS DE RENDIMIENTO.

Modelo	Partición	% Meta-features	Correlación media $\lvert r \rvert$	Tipo predominante	Coste	Impacto
AutoSklearn	ttc = 4	25.0 %	0.80	Estadísticas (media, var., rango)	1	3
GMM	ttc = 0	9.5 %	0.94	Estadísticas / Info-teóricas	1–2	2–3
GMM	ttc = 1	8.3 %	0.95	Estadísticas / Info-teóricas	1–2	2–3
GMM	ttc = 5	20.2 %	0.85	Estadísticas / Correlación	1	3
XGB	ttc = 3	1.2 %	0.82	Correlación / Información mutua	2	2
XGB	ttc = 4	4.8 %	0.82	Correlación atributo-clase	2	2
XGB	ttc = 5	6.0 %	0.78	Información mutua	2	2

Tabla III
GANANCIA PROMEDIO (Δ) EN MÉTRICAS DE DESEMPEÑO TRAS EL PREPROCESAMIENTO, POR ALGORITMO DE APRENDIZAJE AUTOMÁTICO Y COMBINACIONES DE ENTRENAMIENTO–PRUEBA (*ttc*).

Modelo	ttc	Técnica	MCC	F1	Acc.	Sens.	Esp.
AutoSklearn	4	Scaler	-0.05	0.02	0.02	0.02	0.00
		PCA	-0.04	-0.06	-0.06	-0.06	-0.03
		SelectKBest	-0.10	-0.07	-0.07	-0.07	-0.03
		CorrIncrease	-0.02	-0.01	-0.01	-0.01	-0.01
		QuantileTransf	**0.06**	**0.03**	**0.03**	**0.03**	**0.00**
GMM	0	Scaler	0.23	0.23	0.23	0.23	0.08
		PCA	0.13	0.13	0.13	0.13	0.04
		SelectKBest	0.16	0.14	0.14	0.14	0.05
		CorrIncrease	0.05	0.04	0.04	0.04	0.01
		QuantileTransf	**0.30**	**0.26**	**0.26**	**0.26**	**0.09**
	1	Scaler	0.24	0.19	0.19	0.19	0.07
		PCA	0.14	0.09	0.09	0.09	0.03
		SelectKBest	0.23	0.16	0.16	0.16	0.05
		CorrIncrease	0.11	0.03	0.03	0.03	0.00
		QuantileTransf	**0.27**	**0.23**	**0.23**	**0.23**	**0.08**
	5	Scaler	0.02	0.04	0.04	0.04	0.01
		PCA	-0.02	0.01	0.01	0.01	0.00
		SelectKBest	0.02	0.04	0.04	0.04	0.01
		CorrIncrease	-0.13	-0.28	-0.28	-0.28	0.10
		QuantileTransf	**0.06**	**0.06**	**0.06**	**0.06**	**0.02**
XGB	3	Scaler	0.10	0.07	0.07	0.07	0.02
		PCA	0.05	0.04	0.04	0.04	0.01
		SelectKBest	0.06	0.05	0.05	0.05	0.02
		CorrIncrease	0.02	0.04	0.04	0.04	0.01
		QuantileTransf	**0.11**	**0.08**	**0.08**	**0.08**	**0.03**
	4	Scaler	-0.07	-0.05	-0.05	-0.05	-0.02
		PCA	-0.10	-0.07	-0.07	-0.07	-0.02
		SelectKBest	-0.24	-0.25	-0.25	-0.25	-0.08
		CorrIncrease	-0.11	-0.06	-0.06	-0.06	-0.02
		QuantileTransf	-0.03	-0.02	-0.02	-0.02	-0.01
	5	Scaler	0.00	0.00	0.00	0.00	0.00
		PCA	-0.06	-0.03	-0.03	-0.03	-0.01
		SelectKBest	-0.07	-0.05	-0.05	-0.05	-0.02
		CorrIncrease	-0.04	0.00	0.00	0.00	0.00
		QuantileTransf	**0.04**	**0.02**	**0.02**	**0.02**	**0.01**

de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation).

REFERENCIAS

[1] M. Feurer, A. Klein, K. Eggensperger, J. T. Springenberg, M. Blum y F. Hutter: "Efficient and Robust Automated Machine Learning", en *Proceedings of the 29th International Conference on Neural Information Processing Systems (NeurIPS)*, pp. 2755–2763, 2015.

[2] N. DeCastro-García y D. Escudero García: "Meta-feature-based Data Preprocessing for Machine Learning through Correlation Structures and Cost-aware Benchmarking: A Case Study in Cybersecurity", en *AIMS Mathematics*, vol. 10, n. 7, pp. 16762–16795, 2025.

[3] A. Rivolli, L. P. F. Garcia, C. Soares, J. Vanschoren y A. C. P. L. F. de Carvalho: "Meta-features for Meta-learning", en *Knowledge-Based Systems*, vol. 240, pp. 108101, 2022.

[4] N. DeCastro-García y D. Escudero García: "INCIBE-CERT Malicious IPs Dataset", *Repositorio público de datos de investigación*, disponible en: https://github.com/amunc/IP_datasets, 2021.

[5] N. DeCastro-García, D. Escudero García y M. V. Carriegos: "A Mathematical Analysis about the Geo-Temporal Characterization of the Multi-Class Maliciousness of an IP Address", en *Wireless Networks*, vol. 30, n. 6, pp. 5033–5048, 2024.

[6] D. Escudero García y N. DeCastro-García: "Transfer and Online Learning for IP Maliciousness Prediction in a Concept Drift Scenario", en *Wireless Networks*, vol. 30, n. 9, pp. 7423–7444, 2024.

[7] E. Alcobaça, F. Siqueira, A. Rivolli, L. P. F. Garcia, J. T. Oliva y A. C. P. L. F. de Carvalho: "MFE: Towards Reproducible Meta-feature Extraction", en *Journal of Machine Learning Research*, vol. 21, n. 111, pp. 1–5, 2020.

[8] D. J. Hand: "Mixture Models: Inference and Applications to Clustering", en *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, vol. 38, n. 3, pp. 384–385, 1989.

[9] T. Chen y C. Guestrin: "XGBoost: A Scalable Tree Boosting System", en *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.

[10] P. Giudici: "Safe Machine Learning", en *Statistics*, vol. 58, pp. 473–477, 2024.

Existence conditions of codes for content protection in medical environments

Marcel Fernandez
Universitat Politècnica de Catalunya
Barcelona, Spain

Sebastià Martín
Universitat Politècnica de Catalunya
Barcelona, Spain

Daniel Escanez-Exposito
University of La Laguna
38200 Tenerife, Spain
jescanez@ull.edu.es

Pino Caballero Gil
University of La Laguna
38200 Tenerife, Spain
pcaballe@ull.edu.es

***Resumen*—This work discusses lower bounds for the rate of separable codes. $\bar{t}$-separable codes are used in copyright protection schemes, where $\bar{t}$ denotes the number of illegitimate users it protects against. For the binary alphabets only the protection against two illegitimate users has been thoroughly studied. In this paper, we revisit the case $t = 2$ and discuss the case $t = 3$.**

***Index Terms*—Illegal redistribution of content in medical environments, separable codes, Lovasz local lemma**

I. Introduction

In medical environments, leakage of content is a very concerning issue. Once many individuals are in possession of a given content, let's say for diagnostic purposes, if that content is redistributed it might be very difficult to find the source of the leak. In this case, a way to offer protection is to make each copy of the content unique. This can be done by embedding a different mark in each copy. This work does not deal on how to embed marks, we assume this can be done by means of a watermarking algorithm. Instead, we focus on the structure of the marks to be embedded.

To get a feeling about how this structure should be, notice that since each copy of a certain content contains a unique mark then this content can be traced back in an unambiguous way. Therefore, naive redistribution is clearly deterred. Unfortunately, two or more users can get together and create a new copy from their copies. This new copy is called a pirate copy and is the one that they redistribute. This attack is called a *collusion* attack. Separable codes [1], [2] are used to offer protection against collusion attacks [3], [4].

I-A. $\bar{t}$-separable codes

The marks in the previous section will be the code words of a code. We represent a code C as an $M \times n$ matrix with binary entries. Take a set U of at most $\bar{t}$ rows. From this set we can obtain another set D_U using the following rules:

- If a column in U consists of all zeros, then all words in D_U have a zero in that position. Likewise, if a column consists of all ones, then all words in D_U have a zero in that position.
- If a column in U contains zeros and ones, then words in D_U can have a zero or a one in that position.

We take D_U to be the set of all possible code words that can be obtained with the previous rules. We say that matrix C has the *$\bar{t}$-separable property* if for any two *different* sets of rows, say U and V, each of size at most t, the sets D_U and D_V are also *different*. In other words, if $U \neq V$ then $D_U \neq D_V$.

I-B. Our contribution

For a state-of-the-art on bounds for separable codes, we refer the reader to the elegant expositions in [5]. There are several results about codes over q-ary alphabets, but the binary case has only been studied for the case $t = 2$. In this paper, using the Lovász Local Lemma, we study the rate of 3-separable binary codes. More precisely we show that binary 3-separable codes with positive asymptotic rate exist. This means that we can obtain codes whose size is exponential in the code length. See Theorem 11 and Corollary 12 below.

II. Previous results

We adopt the notation of [5]. Let $\mathcal{Q}$ be an alphabet of size Q. A code $\mathcal{C}$ of length n and size M is a subset of n-tuples in $\mathcal{Q}^n$, i.e. $\mathcal{C} = \{c_1, \ldots, c_M\}$. The n-tuples $c_i = (c_i(1), \ldots, c_i(n)) \in \mathcal{C}$, $1 \leq i \leq M$, are called code words. The distance between two code words c_i and c_j is the number of position in which they differ, $|\{l \in \{1, \ldots, n\} : c_i(l) \neq c_j(l)|$. The minimum distance of a code is the minimum distance between two code words and will be denoted by d. In this case, we say that $\mathcal{C}$ is an $(n, M, d)_q$ code or $(n, M)_q$ for short.

We can arrange an $(n, M)_q$ code $\mathcal{C}$ as an $M \times n$ matrix C, rows being code words. With a slight abuse of notation, we denote the row set of C as $\{c_1, \cdots, c_M\}$. A set U of t rows $\{c_{i_1}, \ldots, c_{i_t}\}$ will be denoted as U^t.

An important parameter of codes is their rate,

$$R = \frac{\log_q M}{n}. \tag{1}$$

Given $U \subseteq \mathcal{C}$, we define the ith projection set of U as

$$U(i) = \{u_j(i) \in \mathcal{Q} : u_j \in U\},\ 1 \leq i \leq n. \tag{2}$$

Also, we define the *descendant set* of U as

$$\begin{aligned}\mathrm{desc}(U) = U(1) \times \cdots \times U(n) = \\ \{z \in \mathcal{Q}^n : z(i) \in U(i),\ 1 \leq i \leq n\}.\end{aligned} \tag{3}$$

Definition 1: Let $t \geq 2$ be an integer. We say that an $(n, M)_q$ code $\mathcal{C}$ is a $\bar{t}$-separable code, $\bar{t}$-SC, if for all distinct $U, V \in \mathcal{C}$ with $|U| \leq t$ and $|V| \leq t$, we have $\mathrm{desc}(U) \neq \mathrm{desc}(V)$.

We will denote by $S(\bar{t}, n, Q)$ the largest cardinality of a $\bar{t}$-SC.

Let $R_q(n, t)$ be the optimal rate of an $(n, M)_q$ code. As is customary in Information Theory, we are interested in the asymptotic rate:

$$\underline{R}_Q(t) = \limsup_{n\to\infty} R_q(n,t). \tag{4}$$

We will use the following notation for the asymptotic rate of $\bar{t}$-SC codes:

$$s(\bar{t},Q) = \limsup_{n\to\infty} \frac{\log_q S(\bar{t},n,Q)}{n} \tag{5}$$

II-A. Lovász Local Lemma

We start with a sample space Ω. We will define *independent* random variables, say $V_1,\ldots,V_m$, taking values in Ω. In this context, we will have events $E_1,\ldots,E_b$ that are considered "bad". We will assume that all events are defined based on $V_1,\ldots,V_m$. The *scope* $\mathrm{sc}(E)$ of an event E is the minimal subset of random variables that determines its occurrence. We say that two events are dependent if their respective scopes intersect. In the sequel, these events are going to be ordered.

With the events at hand, we can define a *dependency graph*. The vertices are the events $E_1,\ldots,E_b$, and there is an edge between E_i and E_j if they are dependent. The neighborhood of a vertex E_j will be denoted by Γ_j. We will consider that no vertex belongs to its neighborhood. We denote by $s \geq 1$ the maximum degree of the graph. Therefore, $|\Gamma_j| \leq s$ for $j = 1,\ldots,b$.

The Lovász Local Lemma [6] gives a sufficient condition for the existence of a configuration, that is, an assignment to r.v. $V_1,\ldots,V_m$ that avoids all the bad events $E_1,\ldots,E_b$. Intuitively, a configuration will exist if the probability of each bad event is not too large and if the dependence among the events is limited.

Lemma 2 (Lovász Local Lemma or LLL): Using the above notation, suppose $E_1,\ldots,E_b$ are events with dependency graph G. Suppose also that there exist real numbers $x_i \in [0,1)$ assigned to the events E_i such that

$$\Pr[E_i] \leq x_i \prod_{j\in\Gamma_i}(1-x_j), \tag{6}$$

then

$$\Pr\left[\bigcap_{i=1}^{b} \overline{E}_i\right] > 0.$$

That is, all bad events can be avoided.

Usually the Lovász Local Lemma is used in its symmetric form. In our case, events are too diverse for the symmetric version. The following is an intermediate version between the general version and the symmetric and is strong enough for our purposes.

Lemma 3: Let $\Pr[E_i] \leq 1/4$ and $x_i = 2\Pr[E_i]$, for all $i = 1,\ldots,b$. Then, Lemma 2 holds if

$$\forall i,\ \sum_{j\in\Gamma_i} \Pr[E_j] \leq \frac{1}{4}. \tag{7}$$

Proof: Substituting $x_i = 2\Pr[E_i]$ in the right side of (6), we have

$$x_i \prod_{j\in\Gamma_i}(1-x_j) = 2\Pr[E_i] \prod_{j\in\Gamma_i}(1-2\Pr[E_j]) \tag{8}$$

$$\geq 2\Pr[E_i]\left(1 - \sum_{j\in\Gamma_i} 2\Pr[E_j]\right) \tag{9}$$

$$\geq 2\Pr[E_i]\left(1 - 2\cdot\frac{1}{4}\right) \tag{10}$$

$$= \Pr[E_i]. \tag{11}$$

Therefore, (6) is satisfied. The first inequality can be proved by induction, and the second one follows from the assumptions in the lemma. ∎

III. General setup

The main challenge when one uses the LLL is to devise how to define the random variables and events that characterize the object we wish to obtain.

In this paper, we focus on establishing lower bounds on the rate of binary $\bar{t}$-separable codes, with $t \in \{2,3\}$. We take Mn random variables

$$X_1,\ldots,X_{Mn}, \tag{12}$$

with Bernoulli distribution $\Pr[X_j = 0] = q$, $\Pr[X_j = 1] = 1-q$, and arrange them as an $M \times n$ matrix C, with rows $\{c_1,\ldots,c_M\}$, $c_i = \{X_{i1},\ldots,X_{in}\}$. An assigment to these random variabes is a code of size M and length n, where code words correspond to rows, that is position j of code word i corresponds to the entry of the matrix X_{ij}.

Let U_α, V_γ be a pair of different sets of rows of C, with $|U_\alpha| = \alpha$, $|V_\gamma| = \gamma$, and $W = U_\alpha \cap V_\gamma$, with $|W| = \beta$ and $0 \leq \beta < \gamma \leq \alpha \leq t$.

Let $\chi(\alpha,\gamma) = 0$, when $\alpha = \gamma$, and $\chi(\alpha,\gamma) = 1$ otherwise. For every β,γ,α such that $0 \leq \beta < \gamma \leq \alpha \leq t$, there are

$$\frac{1}{2^{\chi(\alpha,\gamma)}}\binom{M}{\alpha}\binom{\alpha}{\beta}\binom{M-\alpha}{\gamma-\beta} = \frac{1}{2^{\chi(\alpha,\gamma)}}\binom{M}{\alpha+\gamma-\beta}\binom{\alpha+\gamma-\beta}{\beta}\binom{\alpha+\gamma-2\beta}{\alpha-\beta} \tag{13}$$

such pairs, which we assume to be ordered arbitrarily.

Definition 4: We define the event $E^{\beta}(U_\alpha,V_\gamma) = E^{\alpha,\beta,\gamma}$ to occur when an assignment to the random variables in U_α, V_γ are *not* separated in the sense of Definition 1. These are the "bad" events we wish to avoid.

Observation 5: Let us comment on the most basic bad event, which happens when $|U_\alpha| = |V_\gamma| = 1$. In other words, we are considering $\alpha = \gamma = 1$, which implies $\beta = 0$. In this case, the event $E^{1,0,1}$ is bad if and only if $U_\alpha = V_\gamma$. But this cannot occur if one makes sure all code words are different. Moreover, if all code words are different, the cases $E^{\alpha,0,1}$ and $E^{2,1,2}$ will never occur. This means that avoiding all bad events $E^{1,0,1}$, we do not have to deal with events $E^{\alpha,0,1}$ and $E^{2,1,2}$.

Consider random variables

$$\mathbf{Y}_i = Y_{i,1},\ldots,Y_{i,n},\quad i = 1,\ldots,\alpha-\beta,$$
$$\mathbf{Z}_j = Z_{j,1},\ldots,Z_{j,n},\quad j = 1,\ldots,\gamma-\beta,$$
$$\mathbf{W}_k = W_{k,1},\ldots,W_{k,n},\quad k = 1,\ldots,\beta,$$

associated with two sets of code words U_α, V_γ:

$$U_\alpha = \{\mathbf{Y}_1, \ldots, \mathbf{Y}_{\alpha-\beta}, \mathbf{W}_1, \ldots, \mathbf{W}_\beta\},$$
$$V_\gamma = \{\mathbf{Z}_1, \ldots, \mathbf{Z}_{\gamma-\beta}, \mathbf{W}_1, \ldots, \mathbf{W}_\beta\}. \quad (14)$$

We denote an assignment to these random variables as

$$\{\mathbf{u}_1, \ldots, \mathbf{u}_{\alpha-\beta}, \mathbf{w}_1, \ldots, \mathbf{w}_\beta\}, \{\mathbf{v}_1, \ldots, \mathbf{v}_{\gamma-\beta}, \mathbf{w}_1, \ldots, \mathbf{w}_\beta\}.$$

An assignment is bad if

$$\mathrm{desc}(\{\mathbf{u}_1, \ldots, \mathbf{u}_{\alpha-\beta}, \mathbf{w}_1, \ldots, \mathbf{w}_\beta\}) = \mathrm{desc}(\{\mathbf{v}_1, \ldots, \mathbf{v}_{\gamma-\beta}, \mathbf{w}_1, \ldots, \mathbf{w}_\beta\}). \quad (15)$$

III-A. Dependence between bad events

We say that two events $E^{\alpha,\beta,\gamma}$, $E^{\alpha',\beta',\gamma'}$ are dependent if

$$(U_\alpha \cup V_\gamma) \cap (U_{\alpha'} \cup V_{\gamma'}) \neq \emptyset. \quad (16)$$

In other words, two events are dependent if they share at least one random variable.

We start by counting the number of events of type $E^{a,b,c}$ dependent with a given event of type $E^{\alpha,\beta,\gamma}$, and denoted by $d\left[E^{a,b,c}, E^{\alpha,\beta,\gamma}\right]$, taking into account (16).

Theorem 6: Given $a \geq c > b \geq 0$ and $\alpha \geq \gamma > \beta \geq 0$, let $\chi(a,c) = 1$ if $a = c$, and $\chi(a,c) = 0$ otherwise. Then,

$$d\left[E^{a,b,c}, E^{\alpha,\beta,\gamma}\right] \leq \quad (17)$$
$$\frac{\alpha+\gamma-\beta}{2^{\chi(a,c)}} \binom{M-1}{a+c-b-1} \binom{a+c-b}{b} \binom{a+c-2b}{a-b}. \quad (18)$$

Proof: The proof follows by (13), taking into account that two events are dependent if they share at least a row of random variables, and there are $\alpha+\gamma-\beta$ different rows in $E^{\alpha,\beta,\gamma}$. ∎

Corollary 7: Given $a \geq c > b \geq 0$ and $\alpha \geq \gamma > \beta \geq 0$, then

$$d\left[E^{a,b,c}, E^{\alpha,\beta,\gamma}\right] \leq k\, M^{a+c-b-1}, \quad (19)$$

where $k > 0$ does not depend on M.

III-B. Probability of a bad event

Let us now compute the probability $\Pr(E^{a,b,c})$ that an assignment to the random variables associated to $E^{a,b,c}$ is bad as in Definition 4.

Theorem 8: Let $a \geq c > b \geq 0$, $u \in \{0,1\}^{a-b}$, $m \in \{0,1\}^b$. Then, the probability $\Pr[E^{a,b,c}]$ event $E^{a,b,c}$ being bad is:

$$\Pr[E^{a,b,c}] = (1 + 2q^{a+c-b} + 2(1-q)^{a+c-b} - q^a - q^c - (1-q)^a - (1-q)^c)^n, \text{ if } b \geq 1. \quad (20)$$

$$\Pr[E^{a,0,c}] = (q^{a+c} + (1-q)^{a+c} + (1 - q^a - (1-q)^a) \cdot (1 - q^c - (1-q)^c))^n. \quad (21)$$

Proof: Observe that, for $E^{a,b,c}$,

$$\mathrm{desc}(\{\mathbf{u}_1, \ldots, \mathbf{u}_{a-b}, \mathbf{w}_1, \ldots, \mathbf{w}_b\}) = \mathrm{desc}(\{\mathbf{v}_1, \ldots, \mathbf{v}_{c-b}, \mathbf{w}_1, \ldots, \mathbf{w}_b\}), \quad (22)$$

if and only if the support of multi-sets $\{u_{1,k}, \ldots, u_{a-b,k}, w_{1,k}, \ldots, w_{b,k}\}$ and $\{v_{1,k}, \ldots, v_{c-b,k}, w_{1,k}, \ldots, w_{b,k}\}$, are equal, for $1 \leq k \leq n$. Therefore, we will deal with the probability of a single column being *bad*, $\Pr(C_{abc})$. We consider the following cases:

$E^{a,0,c}$:

$$\begin{aligned}\Pr(C_{a0c}) &= \Pr(C_{a0c} \mid u = 0^a) \cdot \Pr(u = 0^a) \\ &\quad + \Pr(C_{a0c} \mid u = 1^a) \cdot \Pr(u = 1^a) \\ &\quad + \Pr(C_{a0c} \mid u \notin \{0^a, 1^a\}) \cdot \Pr(u \notin \{0^a, 1^a\}) \\ &= q^c \cdot q^a + (1-q)^c \cdot (1-q)^a \\ &\quad + (1 - q^c - (1-q)^c) \cdot (1 - q^a - (1-q)^a).\end{aligned}$$

$E^{a,b,c}$, $b \geq 1$:

$$\begin{aligned}\Pr(C_{abc}) &= \Pr((C_{abc} \mid m = 0^b) \cdot \Pr(m = 0^b) \\ &\quad + \Pr((C_{abc} \mid m = 1^b) \cdot \Pr(m = 1^b) \\ &\quad + \Pr((C_{abc} \mid m \notin \{0^b, 1^b\}) \cdot \Pr(m \notin \{0^b, 1^b\}) \\ &= [\Pr((C_{abc} \mid m = 0^b, u = 0^{a-b}) \cdot \Pr(u = 0^{a-b}) \\ &\quad + \Pr((C_{abc} \mid m = 0^b, u \neq 0^{a-b}) \cdot \Pr(u \neq 0^{a-b})] \\ &\quad \cdot \Pr(m = 0^b) \\ &\quad + [\Pr((C_{abc} \mid m = 1^b, u = 1^{a-b}) \cdot \Pr(u = 1^{a-b}) \\ &\quad + \Pr((C_{abc} \mid m = 1^b, u \neq 1^{a-b}) \cdot \Pr(u \neq 1^{a-b})] \\ &\quad \cdot \Pr(m = 1^b) \\ &\quad + \Pr((C_{abc} \mid m \notin \{0^b, 1^b\}) \cdot \Pr(m \notin \{0^b, 1^b\}) \\ &= [q^{c-b}q^{a-b} + (1 - q^{c-b})(1 - q^{a-b})]\, q^b \\ &\quad + [(1-q)^{c-b}(1-q)^{a-b} \\ &\quad + (1 - (1-q)^{c-b})(1 - (1-q)^{a-b})]\, (1-q)^b \\ &\quad + 1 \cdot (1 - q^b - (1-q)^b) \\ &= 1 + 2q^{a+c-b} + 2(1-q)^{a+c-b} \\ &\quad - q^a - q^c - (1-q)^a - (1-q)^c.\end{aligned}$$

The theorem follows due to independence among columns. ∎

Our ultimate goal is to prove the existence of $\bar{t}$-separable codes of positive asymptotic rate. This means that for a given t, the size of the code has the form:

$$M_t = k_t\, r_t^n, \quad k_t > 0,\ r_t > 1.$$

In order to show that $\bar{t}$-separable codes of such form exist, we will use (7) to obtain a lower bound on M. From the expressions in Theorem 6 and Theorem 8, this does not seem easy for general values of t and q. Therefore, we will focus on $t \in \{2, 3\}$.

IV. Case $t = 2$

In order to better illustrate our strategy we will first derive an already known result [5], [7], that is, when $\bar{t} = 2$, in an alternative way.

We will deduce the same result as in [5] from the version of the Lovász Local Lemma presented in Lemma 3. This will be the starting point for the unknown case $\bar{t} = 3$.

First, let us see that the optimal value of q is $1/2$. According to (21) in Theorem 8, for events of type:

- $E^{1,0,1}$, we have:

$$\Pr(E^{1,0,1}) = [q^2 + (1-q)^2]^n,$$

where $f(q) = q^2 + (1-q)^2$ is minimum when $q = 1/2$, and so

$$\Pr(E^{1,0,1}) = \left(\frac{1}{2}\right)^n$$

- $E^{2,0,2}$, we have:

$$\Pr(E^{2,0,2}) = [q^4 + (1-q)^4 + (1 - q^2 - (1-q)^2)^2]^n.$$

Since

$$\frac{d}{dq}(q^4 + (1-q)^4 + (1 - q^2 - (1-q)^2)^2) = 8(q - 1/2)(3q^2 - 3q + 1) \quad (23)$$

and

$$8(q - 1/2)(3q^2 - 3q + 1) \geq 0 \iff q \geq 1/2,$$

the minimum is achieved when $q = 1/2$, and

$$\Pr(E^{2,0,2}) = \left(\frac{3}{8}\right)^n.$$

Next theorem gives a lower bound on the size of 2-separable codes.

Theorem 9: If $n \geq 8$ and $q = 1/2$, then there exist $\bar{2}$-separable binary codes of size

$$S(\bar{2}, n, 2) \geq \left\lfloor \frac{1}{2}\left(\frac{8}{3}\right)^{n/3} \right\rfloor. \quad (24)$$

Proof: The proof follows by applying the general version of the Lovász Local Lemma in Lemma 3.

Lemma 3 requires $\Pr[E_i] \leq 1/4$, and this happens whenever $n \geq 2$, since $\Pr[E_i] \in \left\{\left(\frac{1}{2}\right)^n, \left(\frac{3}{8}\right)^n\right\}$.

1. For events $E_i \in E^{2,0,2}$:

$$\begin{aligned}\sum_{j\in\Gamma_i} \Pr[E_j] &= \left(d\left[E^{1,0,1}, E^{2,0,2}\right] + 1\right)\left(\frac{1}{2}\right)^n \\ &+ \left(d\left[E^{2,0,2}, E^{2,0,2}\right] + 1\right)\left(\frac{3}{8}\right)^n \\ &= (4M - 9)\left(\frac{1}{2}\right)^n \\ &+ (2M^3 - 21M^2 + 79M - 105)\left(\frac{3}{8}\right)^n \end{aligned} \quad (25)$$

We would like to know which is the *largest* term in (25). Observe that,

a) Since $4M\left(\frac{1}{2}\right)^n \leq \frac{1}{4} \iff M \leq \frac{2^n}{16}$. If $M = \frac{2^n}{16}$, then

$$2M^3\left(\frac{3}{8}\right)^n = \frac{3^n}{2^{11}} > 1, \forall n \geq 7.$$

b) On the other hand,

$$2M^3\left(\frac{3}{8}\right)^n \leq \frac{1}{4} \iff M \leq \frac{1}{2}\left(\frac{8}{3}\right)^{n/3}. \quad (26)$$

c) But,

$$4M\left(\frac{1}{2}\right)^n \leq (21M^2 - 79M)\left(\frac{3}{8}\right)^n \iff M \geq \frac{4}{21}\left(\frac{4}{3}\right)^n + \frac{79}{21}. \quad (27)$$

Since $\frac{4}{21}\left(\frac{4}{3}\right)^n + \frac{79}{21} < \left\lfloor \frac{1}{2}\left(\frac{8}{3}\right)^{n/3} \right\rfloor$, for all $n \geq 8$, we conclude that, for events $E_i \in E^{2,0,2}$:

$$\sum_{j\in\Gamma_i} \Pr[E_j] < 2M^3\left(\frac{3}{8}\right)^n, \text{ when } M = \left\lfloor \frac{1}{2}\left(\frac{8}{3}\right)^{n/3} \right\rfloor.$$

2. For events $E_i \in E^{1,0,1}$ according to Theorem 6 and the previous calculation, we have:

$$\sum_{j\in\Gamma_i} \Pr[E_j] \leq M^3\left(\frac{3}{8}\right)^n, \text{ when } M \geq \left\lfloor \frac{1}{2}\left(\frac{8}{3}\right)^{n/3} \right\rfloor.$$

We conclude that the inequality (7) in Lemma 3 holds when

$$2M^3\left(\frac{3}{8}\right)^n \leq \frac{1}{4}, \text{ which leads to } M \leq \frac{1}{2}\left(\frac{8}{3}\right)^{n/3}.$$

■

The following corollary states the asymptotic rate.

Corollary 10: There exist $\bar{2}$-separable binary codes of rate

$$s(\bar{2}, 2) \geq 1 - \frac{\log_2 3}{3} \quad (28)$$

Proof: Just apply the definition of $s(\bar{2}, Q)$ in (5). ■

Note that this is the same rate obtained in Corollary 4 of [5].

V. Case $t = 3$

In this section we discuss 3-separable codes. Taking into account Observation 5, we need to consider events of type $E^{1,0,1}$, $E^{2,0,2}$, $E^{3,0,3}$, $E^{3,1,3}$, $E^{3,2,3}$, $E^{3,0,2}$, $E^{3,1,2}$, and $E^{3,2,2}$. We define the set

$$\begin{aligned}\mathcal{T}_3 = \{&(1,0,1),(2,0,2),(3,0,3),(3,1,3),\\ &(3,2,3),(3,0,2),(3,1,2),(3,2,2)\}\end{aligned}$$

V-A. Dependence between events

Recall from Section III (take into account (16)) that $d\left[E^{a,b,c}, E^{\alpha,\beta,\gamma}\right]$ denotes the number of events of types $E^{a,b,c}$ dependent with a given event of type $E^{\alpha,\beta,\gamma}$.

In general, for $\alpha \geq \gamma > \beta \geq 0$, we define:

$$\begin{cases}\chi(\alpha,\gamma) = 1, & \text{if } \alpha = \gamma \\ \chi(\alpha,\gamma) = 0, & \text{otherwise,}\end{cases} \quad (29)$$

and we have:

- $d\left[E^{1,0,1}, E^{\alpha,\beta,\gamma}\right] < \frac{\alpha+\gamma-\beta}{2\chi(\alpha,\gamma)} 6M \leq 30M.$
- $d\left[E^{2,0,2}, E^{\alpha,\beta,\gamma}\right] < \frac{\alpha+\gamma-\beta}{2\chi(\alpha,\gamma)} M^3/2 \leq \frac{5M^3}{2}.$
- $d\left[E^{3,0,3}, E^{\alpha,\beta,\gamma}\right] < \frac{\alpha+\gamma-\beta}{2\chi(\alpha,\gamma)} M^5/12 \leq \frac{5M^5}{12}.$
- $d\left[E^{3,1,3}, E^{\alpha,\beta,\gamma}\right] < \frac{\alpha+\gamma-\beta}{2\chi(\alpha,\gamma)} \frac{5M^4}{4} \leq \frac{25M^4}{4}.$
- $d\left[E^{3,2,3}, E^{\alpha,\beta,\gamma}\right] < \frac{\alpha+\gamma-\beta}{2\chi(\alpha,\gamma)} 2M^3 \leq 10M^3.$
- $d\left[E^{3,0,2}, E^{\alpha,\beta,\gamma}\right] < \frac{\alpha+\gamma-\beta}{2\chi(\alpha,\gamma)} \frac{5M^4}{6} \leq \frac{25M^4}{6}.$

- $d\left[E^{3,1,2}, E^{\alpha,\beta,\gamma}\right] < \dfrac{\alpha+\gamma-\beta}{2^{\chi(\alpha,\gamma)}} 4M^3 \leq 20M^3$.
- $d\left[E^{3,2,2}, E^{\alpha,\beta,\gamma}\right] < \dfrac{\alpha+\gamma-\beta}{2^{\chi(\alpha,\gamma)}} M^2 \leq 5M^2$.

The last inequality in each row comes from the fact that $\max \dfrac{\alpha+\gamma-\beta}{2^{\chi(\alpha,\gamma)}} = 5$, corresponding to the case $\alpha = 3, \beta = 0, \gamma = 2$.

V-B. Probability of bad events

As we did for he case $\bar{t} = 2$, we are going to:

1. For each event type $E^{a,b,c}$ with $(a,b,c) \in \mathcal{T}_3$:
   a) Compute the probability of $E^{a,b,c}$ being a bad event as a function of q, $\Pr[E^{a,b,c}](q)$.
   b) Find the value $q = q_{a,b,c}$ that minimizes $Pr[E^{a,b,c}](q)$.
2. For each $(a,b,c) \in \mathcal{T}_3$, identify the largest probability $q_{a',b',c'}$ among the probabilities that minimize the bad error probability of each event, that is,

$$q_{a',b',c'} = \max_{E^{a,b,c}} \left\{ \min_{q_{a,b,c}} \Pr[E^{a,b,c}](q_{a,b,c}) \right\} \tag{30}$$

3. For each $(a,b,c) \in \mathcal{T}_3$, compute probabilities $\Pr[E^{a,b,c}]' = \Pr[E^{a,b,c}](q_{a',b',c'})$.

More precisely,

- $\Pr[E^{3,0,3}](q) = f^n(q)$, where

$$f(q) = 2q^6 - 6q^5 + 24q^4 - 38q^3 + 24q^2 - 6q + 1.$$

Then, $f'(q) = 12(q-\frac{1}{2})g(q)$, where $g(q) = (q^4 - 2q^3 + 7q^2 - 6q + 1)$ has only two real roots $\frac{1}{2} \pm \frac{1}{2}\sqrt{8\sqrt{2} - 11} \approx \{0{,}220,\ 0{,}780\}$, and we have that:

$$q_{3,0,3} = 0{,}220 \ \text{ and } \ \Pr[E^{3,0,3}](q_{3,0,3}) \approx 0{,}490^n \tag{31}$$

- $\Pr[E^{3,1,3}](q) = f^n(q)$, where

$$f(q) = 10q^4 - 20q^3 + 14q^2 - 4q + 1.$$

Then, $f'(q) = 8(q-\frac{1}{2})(q-(\frac{1}{2}-\alpha))(q-(\frac{1}{2}+\alpha))$, $\alpha = \frac{\sqrt{5}}{20} \approx 0{,}112$, and we have that:

$$q_{3,1,3} = \frac{1}{2} - \alpha \ \text{ and } \ \Pr[E^{3,1,3}](q_{3,1,3}) \approx 0{,}614^n \tag{32}$$

- $\Pr[E^{3,2,3}](q) = f^n(q)$, where

$$f(q) = 4q^4 - 8q^3 + 6q^2 - 2q + 1.$$

Then, $f'(q) = 16(q-\frac{1}{2})^3 \geq 0 \iff q \geq \frac{1}{2}$, and we have that:

$$q_{3,2,3} = \frac{1}{2} \ \text{ and } \ \Pr[E^{3,2,3}](q_{3,2,3}) = 0{,}75^n \tag{33}$$

- $\Pr[E^{3,0,2}](q) = f^n(q)$, where

$$f(q) = 11q^4 - 22q^3 + 16q^2 - 5q + 1.$$

Then, $f'(q) = 44(q-\frac{1}{2})(q-(\frac{1}{2}-\alpha))(q-(\frac{1}{2}+\alpha))$, $\alpha = \frac{\sqrt{11}}{22} \approx 0{,}151$, and we have that:

$$q_{3,0,2} = \frac{1}{2} - \alpha \ \text{ and } \ \Pr[E^{3,0,2}](q_{3,0,2}) \approx 0{,}432^n \tag{34}$$

- $\Pr[E^{3,1,2}](q) = f^n(q)$, where

$$f(q) = 4q^4 - 8q^3 + 7q^2 - 3q + 1.$$

Then, $f'(q) = 2(q-\frac{1}{2})(8q^2 - 8q + 3) \geq 0 \iff q \geq \frac{1}{2}$, and we have that:

$$q_{3,1,2} = \frac{1}{2} \ \text{ and } \ \Pr[E^{3,1,2}](q_{3,1,2}) = 0{,}5^n \tag{35}$$

- $\Pr[E^{3,2,2}](q) = f^n(q)$, where

$$f(q) = q^2 - q + 1$$

has minimum value when $q = \frac{1}{2}$ and we have:

$$q_{3,2,2} = \frac{1}{2} \ \text{ and } \ \Pr[E^{3,2,2}](q_{3,2,2}) = 0{,}75^n \tag{36}$$

We have seen that bad event probability is maximum for events of type $E^{3,2,2}$ and $E^{3,2,3}$, and in both cases the optimal value of q is indeed $1/2$.

According to Theorem 8, when $q = 1/2$ the probabilities of a bad event ocurring are:

$$\Pr[E^{1,0,1}](\frac{1}{2}) = \left(\frac{1}{2}\right)^n, \ \Pr[E^{2,0,2}](\frac{1}{2}) = \left(\frac{3}{8}\right)^n,$$
$$\Pr[E^{3,0,3}](\frac{1}{2}) = \left(\frac{19}{32}\right)^n, \ \Pr[E^{3,1,3}](\frac{1}{2}) = \left(\frac{5}{8}\right)^n, \tag{37}$$

$$\Pr[E^{3,2,3}](\frac{1}{2}) = \left(\frac{3}{4}\right)^n, \Pr[E^{3,0,2}](\frac{1}{2}) = \left(\frac{7}{16}\right)^n,$$
$$\Pr[E^{3,1,2}](\frac{1}{2}) = \left(\frac{1}{2}\right)^n, \Pr[E^{3,2,2}](\frac{1}{2}) = \left(\frac{3}{4}\right)^n. \tag{38}$$

V-C. Lower bound of the code rate for $t = 3$

The condition in Lemma 3, when applied to an event $E^{a,b,c}$, becomes

$$\sum_E \Pr[E] \cdot d\left[E^{a,b,c}, E\right] \leq \frac{1}{4}, \tag{39}$$

which is satisfied when:

$$5[6M \cdot \Pr(E^{1,0,1}) + \frac{1}{2}M^3 \cdot \Pr(E^{2,0,2}) + \frac{5}{6}M^4 \cdot \Pr(E^{3,0,2}) +$$
$$4M^3 \cdot \Pr(E^{3,1,2}) + + \frac{1}{12}M^5 \cdot \Pr(E^{3,0,3}) +$$
$$\frac{5}{4}M^4 \cdot \Pr(E^{3,1,3}) + M^2 \cdot \Pr(E^{3,2,2}) + 2M^3 \cdot \Pr(E^{3,2,3})] \leq \frac{1}{4}, \tag{40}$$

that is, when

$$6M\left(\frac{1}{2}\right)^n + \frac{1}{2}M^3\left(\frac{3}{8}\right)^n + \frac{5}{6}M^4\left(\frac{7}{16}\right)^n + 4M^3\left(\frac{1}{2}\right)^n +$$
$$\frac{1}{12}M^5\left(\frac{19}{32}\right)^n + + \frac{5}{4}M^4\left(\frac{5}{8}\right)^n + M^2\left(\frac{3}{4}\right)^n + 2M^3\left(\frac{3}{4}\right)^n$$
$$\leq \frac{1}{20}. \tag{41}$$

Let us suppose $M \geq 12$. Then,

$$M^4 \left(\frac{7}{16}\right)^n = \left(\frac{7}{10}\right)^n M^4 \left(\frac{5}{8}\right)^n,$$
$$M^3 \left(\frac{3}{8}\right)^n = \left(\frac{1}{2}\right)^n M^3 \left(\frac{3}{4}\right)^n,$$
$$M \left(\frac{1}{2}\right)^n \leq \frac{1}{12^2} M^3 \left(\frac{1}{2}\right)^n = \frac{1}{144} \left(\frac{2}{3}\right)^n M^3 \left(\frac{3}{4}\right)^n,$$
$$M^3 \left(\frac{1}{2}\right)^n = \left(\frac{2}{3}\right)^n M^3 \left(\frac{3}{4}\right)^n,$$
$$M^2 \left(\frac{3}{4}\right)^n \leq \frac{1}{12} M^3 \left(\frac{3}{4}\right)^n, \tag{42}$$

then (41) is satisfied when

$$\frac{M^5}{12} \left(\frac{19}{32}\right)^n + aM^4 \left(\frac{5}{8}\right)^n + bM^3 \left(\frac{3}{4}\right)^n \leq \frac{1}{20}, \tag{43}$$

where $a = \frac{5}{4} + \frac{5}{6}\left(\frac{7}{10}\right)^n$, $b = 2 + \frac{1}{12} + \frac{1}{24}\left(\frac{2}{3}\right)^n + \frac{1}{2}\left(\frac{1}{2}\right)^n$.

Now,

$$M^5 \left(\frac{19}{32}\right)^n \leq M^3 \left(\frac{3}{4}\right)^n \iff M \leq \left(\sqrt{\frac{24}{19}}\right)^n \lessapprox (1{,}124)^n.$$

If $M \leq \left(\sqrt{\frac{24}{19}}\right)^n$, then $n \geq 22$, and

$$\frac{M^4 \left(\frac{5}{8}\right)^n}{M^3 \left(\frac{3}{4}\right)^n} = M \left(\frac{5}{6}\right)^n \leq \left(\frac{5\sqrt{24}}{6\sqrt{19}}\right)^n \leq \left(\frac{5\sqrt{24}}{6\sqrt{19}}\right)^{22} \implies$$
$$\implies M^4 \left(\frac{5}{8}\right)^n \leq \left(\frac{5\sqrt{24}}{6\sqrt{19}}\right)^{22} M^3 \left(\frac{3}{4}\right)^n.$$

Consequently, (41) is satisfied when $c'\, M^3 \left(\frac{3}{4}\right)^n \leq \frac{1}{20}$, where

$$c' = \left[\frac{1}{6} + \left(\frac{5\sqrt{24}}{6\sqrt{19}}\right)^{22} \left(\frac{5}{4} + \frac{5}{6}\left(\frac{7}{10}\right)^{22}\right) + \frac{25}{12} + \frac{1}{24}\left(\frac{2}{3}\right)^{22} + \frac{1}{2}\left(\frac{1}{2}\right)^{22}\right] \approx 2{,}546,$$

or, equivalently, when

$$M \geq \frac{1}{\sqrt[3]{c}} \left(\frac{4}{3}\right)^{n/3}, \text{ where } c < 51. \tag{44}$$

We observe that

$$\frac{1}{\sqrt[3]{51}} \left(\frac{4}{3}\right)^{n/3} \approx \frac{1}{\sqrt[3]{51}} (1{,}10064)^n < \left(\sqrt{\frac{24}{19}}\right)^n \approx (1{,}124)^n,$$

$\forall n \in \mathbb{N}$.

Summarizing, we have:

Theorem 11: For $n \geq 22$, $q = \frac{1}{2}$, there exist $\bar{3}$-separable binary codes of size

$$S(\bar{3}, n, 2) \geq \max\left\{12, \left\lfloor k \left(\frac{4}{3}\right)^{n/3} \right\rfloor\right\}, \tag{45}$$

where $k > \frac{1}{\sqrt[3]{51}} \approx 0{,}27$.

Observation: If $n = 22$, then $M = \left\lfloor k \left(\frac{4}{3}\right)^{n/3} \right\rfloor = 2$, and we assumed $M \geq 12$.

Proof: The proof follows from the direct application of the Lovász Local Lemma, recalling we have previously supposed $M \geq 12$. ∎

The following corollary states the asymptotic rate.

Corollary 12: There exist $\bar{3}$-separable binary codes of rate

$$s(\bar{3}, 2) \geq \frac{1}{3} \log\left(\frac{4}{3}\right). \tag{46}$$

Proof: Just apply the definition of $s(\bar{t}, Q)$ in (5). ∎

VI. Conclusion

We have established bounds for $\bar{2}$-separable codes and $\bar{3}$-separable codes. While the bound for $t = 2$ was already known by random coding expurgartion methods, the use of the Lovász Local Lemmaallows to extend the discussion to the case $t = 3$. We would like to note however that these are degenerated cases since an elaborate calculation shows that for $t = 4$ the value of q that minimizes the bad error probability in Theorem 8 is no longer $q = 1/2$, and establishing the rate in this case is no trivial task and will be the object of our further research.

Acknowledgments

The work of Marcel Fernández ansd Sebastia Martin is funded by the Recovery, Transformation, and Resilience Plan, European Union, under INCIBE Cybersecurity Chair CARISMATICA and Proyecto PID2024-156636NB-C21 (MATSE) funded by MCIN/AEI/10.13039/501100011033/ FEDER, UE. The work of Daniel Escanez-Exposito and Pino Caballero-Gil has been possible thanks to the Cybersecurity Chair sponsored by INCIBE through an initiative carried out within the framework of the funds of the Recovery, Transformation and Resilience Plan, financed by the European Union (Next Generation). It is also part of the PID2022-138933OB-I00 and 2023DIG28 IACTA research projects funded by MCIN/ AEI/10.13039/501100011033/FEDER EU, and the CajaCanarias la Caixa Foundation.

Referencias

[1] M. Cheng and Y. Miao, "On anti-collusion codes and detection algorithms for multimedia fingerprinting," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4843–4851, 2011.

[2] M. Cheng, L. Ji, and Y. Miao, "Separable codes," *IEEE Trans. Inf. Theor.*, vol. 58, no. 3, p. 1791–1803, Mar. 2012.

[3] A. Barg, G. R. Blakley, and G. A. Kabatiansky, "Digital fingerprinting codes: Problem statements, constructions, identification of traitors," *IEEE Transactions on Information Theory*, vol. 49, no. 4, pp. 852–865, 2003.

[4] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," *IEEE Transactions on Information Theory*, vol. 44, no. 5, pp. 1897–1905, 1998.

[5] Y. Gu, J. Fan, and Y. Miao, "Improved bounds for Separable codes and B_2 codes," *IEEE Communications Letters*, vol. 24, no. 1, pp. 15–19, 2020.

[6] P. Erdős and L. Lovász, "Problems and results on 3-chromatic hypergraphs and some related questions," *Infinite and finite sets*, vol. 10, pp. 609–627, 1975.

[7] M. Fernandez, J. Livieratos, and S. Martín, "Frameproof codes, separable codes and codes: Bounds and constructions." *Cryptogr. Commun.*, vol. 16, pp. 481–506, 2024.

On the Inconsistency of Linear Complexity Tests: a Surprising Case Study in the Rabbit Battery-TestU01

Jorge Munilla
Universidad de Málaga
E.T.S.I.T. 1.3.7. Málaga, Spain
munilla@ic.uma.es

Darren Hurley-Smith
University of Kent
Kennedy Building, Canterbury, UK
d.hurley-smith@kent.ac.uk

Julio Hernandez-Castro
U. Politécnica de Madrid
ETSISI, D1208, Madrid, Spain
jc.hernandez.castro@upm.es

***Abstract*—Statistical testing of random number generators (RNGs) used in cryptographic applications is critical for security. TestU01 is a test battery that represents the *de facto* state-of-the-art in statistical randomness testing. Our work highlights serious issues with TestU01' Rabbit battery, particularly its linear complexity tests. We explore a concrete sequence Ω, based on the Liouville function, which passes most tests from the Rabbit battery of TestU01 but fails the linear complexity test by quite some margin. Surprisingly, we find that minimal changes to Ω make it easily pass the test. This seriously calls into question the reliability of the linear complexity test included in the TestU01 software library and of the whole Rabbit battery.**

***Index Terms*—RNG, Linear Complexity, Statistical Testing, Discrepancy, Liouville.**

I. Introduction

Random number generation is critical for the correct functioning of security systems. Cryptography, scientific simulation and machine learning algorithms all require varying degrees of *randomness* to function correctly. In the case of cryptographic keys, a source of reliable, unpredictable data is essential.

Statistical tests of randomness have been developed to identify signs of predictability or underlying structure in RNG output. Diehard [1] is one of the oldest examples, developed by Marsaglia. This has since been extended by Brown et al. to include a wider variety of tests, now known as the Dieharder battery [2]. The National Institute of Standards and Technology (NIST) has developed their own battery called STS 2.1.2, part of Special Publication (SP) 800-22 [3] (some of which has been re-implemented in Dieharder). These batteries have formed the core of academic literature for nearly two decades, and are used in a variety of certification schemes for RNGs.

There has been, naturally, some criticism of these batteries in the past. For example, they were developed to test *Pseudo* Random Number Generators (PRNGs), which conform to Independent and Identically Distributed (IID) requirements [4]. Essentially, the algorithms used by PRNGs ensure that consecutive values are independent of each other, and that the distribution of values within a sample of sufficient size will be approximately even. *True* Random Number Generators (TRNGs) on the other hand do not conform to the IID principle, as they rely on unpredictable natural phenomena to provide entropy. Dieharder and STS 2.1.2 are still used to test such generators, but more recent developments in statistical testing have been made to address the advanced requirements of RNG validation.

L'Ecuyer et al.'s TestU01 [5] provides a broader statistical testing *toolkit*, which can be called individually or as part of one of five batteries: Alphabits, Rabbit, Small Crush, Crush and Big Crush. The first two of these are claimed to be better suited to the testing of TRNGs. They are fast and require relatively little data, compared to other tests.

The Liouville function $\Omega(n) = (-1)^{\omega(n)}$, where $\omega(n)$ is the number of prime factors (multiplicity included) of the positive integer n, is an important function in analytic number theory that is related to the Möbius function, the Riemann hypothesis and the zeta function. There is extensive literature involving the properties of Ω, and in particular attempts to prove that it exhibits (possibly local) random-like behavior. This motivated our interest in analyzing the randomness properties of Ω using randomness testing tools.

In this paper, Ω is shown to evade detection by most of the Rabbit tests. The Linear Complexity test identifies issues in Ω, but surprisingly minor modifications are found that prevent TestU01 Rabbit from identifying this new, modified but almost identical sequence as non-random. The implementation of this test in *Scomp.c* is found to have a hyper-sensitivity to the low-order bits of analysed sequences.

This paper is organised as follows: Section II outlines the existing state-of-the-art in academic literature and statistical testing, as used by manufacturers and in current standards. Section III provides an overview of the Liouville sequence. Section IV discusses the hyper-sensitivity of TestU01 to certain minor modifications of the Ω sequence. Section V compares our TestU01 results with those of NIST SP800-22, identifying flaws in both test batteries. Section VI concludes the paper.

II. Background

As previously mentioned, Dieharder and STS 2.1.2 represent some of the most commonly used statistical test batteries, but there are many others.

The Federal Information Processing Standard (FIPS) 140-1 [6] and 140-2 [7] tests were very popular, and played an important part in validating the RNGs in RFID cards. Bundesamt für Sicherheit in der Informationstechnik (BSI) has produced the AIS-20 and AIS-31 [8] documents, the latter of which outlines a test suite that incorporates FIPS 140-1

tests and several more sophisticated tests of randomness. The Mifare DESFire EV2 is an example of an RFID smartcard that has been subject to Common Criteria EAL5+ procedures [9], [10]. Certification, such as Common Criteria EAL4+ and FIPS, provides a third-party report vouching for the stated functionality of a tested security-related product [11].

Finally, TestU01 is one of the most used non-standard test batteries in both academic and industrial circles, which McCullough [12] goes as far as to state has supplanted Diehard as the de facto statistical testing suite.

A. TestU01

TestU01 is not a test battery in the traditional sense; it is a software library written in ANSI C. It includes a wide range of tests and five pre-selected test batteries that can be used over stream or file input; Alphabits, Rabbit, Small Crush, Crush and Big Crush. These batteries range from fast with minimal input requirements (Alphabits and Rabbit) to intensive test batteries requiring many tens of GB to complete their analysis (Big Crush). TestU01 operates over 32-bit subsequences, which may require that sequences generated by $>$ 32-bit generators be divided appropriately.

Maragnon et al. [13] found TestU01 to be the most comprehensive set of statistical tests made available for general use. Their research into atmospheric turbulence as an entropy source explores several test batteries, including TestU01, NIST SP800-22 [14], and AIS-31 [8].

Alphabits and Rabbit are fast and require relatively small sequences of random bits to perform their tests. Both take as input a number of bits and are comprised of four and fifteen tests respectively. These are further sub-divided into repetitions of tests with different parameters, a total of nine different configurations for Alphabits and twenty-six for Rabbit. Of particular interest in this work is the Linear Complexity test (included in the file *Scomp.c*).

B. Scomp of TestU01: Linear Complexity

Scomp.c is a library of tests included in TestU01, which includes functions to analyze the linear complexity of a sequence through its corresponding *profile*. For an n-bit sequence, this Linear Complexity Profile (LCP) refers to the values and jumps that the linear complexity presents when computed for sub-sequences from 1 to n bits. This is computed using the Berlekamp-Massey algorithm [15], [16], which provides the linear complexity of the given binary sequence, along with the minimal connection polynomial of a LFSR (Linear Feedback Shift Register) able to produce such sequence. The order of the polynomial (number of coefficients minus 1) equals to the linear complexity, and longer LFSRs are assumed to imply greater randomness.

The Berlekamp-Massey algorithm is an iterative algorithm: to compute the feedback polynomial of the sub-sequence of length j, the algorithm uses the previously computed feedback polynomial for the sub-sequence of length $j-1$. It first checks if the polynomial generator obtained for the sub-sequence of length $j-1$ can also generate the sequence when the j-th bit is included. If so, the *discrepancy* equals 0 and the feedback polynomial is not modified, otherwise the *discrepancy* equals 1 and the feedback polynomial must be changed. In this latter case, two options are possible; the order of the polynomial is maintained, or it increases. These discrepancy results form the discrepancy sequence.

The Berlekamp-Massey algorithm has, in the general case, complexity $\mathcal{O}(n^2)$ for processing binary sequences of period n. An adapted fast version due to Blackburn [17] achieves complexity $\mathcal{O}(n(\log n)^2 \log\log n)$. For binary sequences of period $n = 2^t$, Games and Chan [18] presented a fast algorithm with complexity $\mathcal{O}(n)$. This algorithm has been generalized for sequences over GF(p^m) of period $n = p^t$ [19].

For a random sequence of length n, the following holds:

1) Expected Linear Complexity [20]: Scomp.c does not check directly the mean and variance of the expected linear complexity (Eq. 1 and Eq. 2) but it does through the analysis of the LCP. More specifically, as explained next, it checks the expected number of jumps and their distribution.

$$LC_\mu = \frac{n}{2} + \frac{9+(-1)^{n+1}}{36} - \frac{1}{2^n}\left(\frac{n}{3}+\frac{2}{9}\right) \tag{1}$$

$$\begin{aligned} LC_{\sigma^2} =& \frac{86}{81} - \frac{1}{2^n}\left(\frac{(27+(-1)^n)n}{54} + \frac{81+(-1)^n}{81}\right) \\ & - \frac{1}{2^{2n}}\left(\frac{n^2}{9} + \frac{4n}{27} + \frac{4}{81}\right) \end{aligned} \tag{2}$$

Additionally, *Scomp.c* does not use the entire sequence or the number specified with nb in *main.c*. Instead, the total number of bits n used for computing the linear complexity is calculated as follows:

$$\begin{aligned} n &= n^* - mod(n^*, 32) \quad \text{with} \\ n^* &= min(NLIM2 + 2\sqrt{n_b}, n_b, NLIM1) \end{aligned} \tag{3}$$

where mod and min return the remainder after division and minimum value respectively, $NLIM1 = 300000$ and $NLIM2 = 10000$. For n_b=1k, 992 bits are processed (multiple of 32, 31x32). By comparison, for $n_b = 10k$, *Scomp.c* uses 10k bits, and for nb = 1M just 2k bits more; i.e. 12k bits.

2) Expected Number of Jumps of LCP [20]: Using the mean and the variance for the expected number of jumps (Eq. 4 and Eq. 5), *Scomp.c* computes a $Z_{score} = (Value - \mu)/\sigma$, and a p-value, using the normal cumulative distribution function, as $p = 1 - normcdf(Z_{score})$. Then, it checks that the latter is within the confidence margin C. For a confidence margin of $C = 99\%$ (that is what TestU01 uses), $0.005 \geq p_{value} \geq 0.995$ (= 1%) so $|Z_{score}| < 2.58$. Similarly, for a confidence margin of 95%: $0.025 \geq p_{value} \geq 0.975$ and $|Z_{score}| < 1.96$.

$$NJ_\mu = \frac{n}{4} + \frac{9+(-1)^{n+1}}{24} - \frac{1}{3\cdot 2^n} \tag{4}$$

$$NJ_{\sigma^2} = \frac{n}{8} - \frac{3+(-1)^n}{17+(-1)^n} + \frac{n}{6\cdot 2^n} + \frac{13+(-1)^{n+1}}{36\cdot 2^n} - \frac{1}{9\cdot 2^{2n}} \tag{5}$$

3) Size (or height) of the jumps of LCP [20]: The jump heights for a random binary sequence are distributed according to the following geometric distribution (Eq. 6):

$$P(X = k) = \frac{1}{2^k} \tag{6}$$

Scomp.c performs a χ^2 goodness of fit test for the observed distribution over the theoretical one. The degrees of freedom (DoF) of the χ^2 test are $DoF = Nbins - 1$, where $Nbins$ stands for the number of bins. To guarantee this χ^2 test is meaningful and properly implemented, *Scomp.c* sets the number of bins so that there is a minimum of 10 expected jumps per bin, as follows:

$$Nbins = \lfloor \lg_2(N/10) \rfloor \tag{7}$$

where N is the actual (observed) total number of jumps.

Performing the χ^2 test requires to compute the χ_{score} value as:

$$\chi_{score} = \sum_{k=1}^{Nbins} \frac{(ExpJ_W(k) - OJ_W(k))^2}{ExpJ_W(k)} \tag{8}$$

where $OJ_W(k)$ is the observed number of jumps in the k-th bin, except for the case $OJ_W(Nbins)$ that accounts for all the jumps with height$\geq Nbins$. $ExpJ_W(k)$ is the expected number of jumps in the k-th bin, which can be computed using Eq. 9, where $Prob(k)$ is calculated using Eq. 10,

$$ExpJ_W(k) = Prob(k) \cdot N \tag{9}$$

$$\text{Prob}(k) = \begin{cases} \frac{1}{2^k} \text{ for } 1 \leq k \leq Nbins - 1 \\ \frac{1}{2^{Nbins-1}} \text{ for } k = Nbins \end{cases} \tag{10}$$

Once the χ^2 test is completed, the corresponding p-value is computed, using the χ-square cumulative distribution function, as $p = 1 - chi2cdf(\chi_{score}, DoF)$. This test fails if the resulting $p_{value} < 1 - C$, where C is the confidence level. Although DoF will depend on the actual number of jumps (applying Eq. 7), it can be approximated as:

$$DoF = \lfloor (\lg_2((n/4)/10)) \rfloor - 1. \tag{11}$$

Thus, for example, for $nb = n = 10$k and $C = 99\%$, we have $DoF \approx 6$ and to pass the test χ-score < 16.8. For $C = 95\%$, χ-score < 12.6. Similarly, for $nb = 1$M, we have $n = 12$k and $DoF \approx 7$, with χ-score < 18.4 and χ-score < 14.07 for $C = 99\%$ and $C = 95\%$, respectively.

C. NIST SP800-22 Linear Complexity test implementation

NIST provides a Linear Complexity test as part of SP800-22 [14]. This test, similarly to that of *Scomp.c*, compares the LCP with the expected distribution over truly random sequences with a χ^2 test. The SP800-22 implementation hard codes the degrees of freedom ($K = 6$) into the test.

SP800-22 deviates from the approach taken by L'Ecuyer et al. in several ways. The overall objective of the test remains the same, but there are some significant differences in the statistics used:

1) Divide an n-bit sequence into N independent blocks of M bits, where $n = MN$.

Table I: Bins (V_x) for SP800-22's Linear Complexity test [14]

T_i	Bin
$T_i \leq -2.5$	V_0
$-2.5 < T_i \leq -1$	V_1
$-1.5 < T_i \leq -0.5$	V_2
$-0.5 < T_i \leq 0.5$	V_3
$0.5 < T_i \leq 1.5$	V_4
$1.5 < T_i \leq 2.5$	V_5
$T_i > 2.5$	V_6

2) The Berlekamp-Massey algorithm is used to determine the linear complexity (L_i) of each block.
3) The theoretical mean μ is computed, exactly as outlined in Eq. 1.
4) Compute $T_i = (-1)^M \cdot (L_i - \mu) + 2/9$.
5) Record values of T_i in 7 bins conforming to Table I.
6) Compute $\chi^2(obs) = \sum_{i=0}^{K} \frac{(v_i - H\pi_i)^2}{N\pi_i}$ with probabilities $\pi_0 = 0.010417$, $\pi_1 = 0.03125$, $\pi_2 = 0.125$, $\pi_3 = 0.5$, $\pi_4 = 0.25$, $\pi_5 = 0.0625$, and $\pi_6 = 0.020833$.
7) Compute the final p-value = $chi2cdf(\frac{K}{2}, \frac{\chi^2(obs)}{2})$.

Unlike TestU01, this version of the Linear Complexity test does not compute a Z_{score} for the estimated number of jumps. Nor does it report back the maximum jump to the user. The resulting p-value and proportion of passed tests are the only statistics reported to the user. This leads to some variation in the performance of these tests, both in execution time and in the results they show over the sequences we have tested in this work.

III. The Liouville sequence ($\Omega(n)$)

As introduced above, the Ω function analyzed here is a bijection into a binary sequence in $\{0, 1\}^*$ of the Liouville function: $\Omega(n) = (-1)^{\omega(n)}$, for $n \geq 1$, where $\omega(n)$ returns the numbers of divisors with multiplicity (e.g. $\omega(8) = \omega(12) = 3$, $\omega(9) = \omega(10) = 2$ and $\omega(p) = 1$ for every prime). It can be easily generated, which greatly eases the replication of our findings.

A. Analysis of the Ω Sequence

The Ω sequence does not pass TestU01 Rabbit. It fails the Linear Complexity Test (*Scomp.c*) with exceptionally bad p-values; the number of reported jumps is significantly above the expected value, and its distribution is also very peculiar. Both the RTT [21] and L'Ecuyer implementations of the test report out-of-bound p-values, which are consistently and extremely low. RTT reports a p-value below its minimum threshold of 10^{-8}, while TestU01 outputs an eps warning (indicating that the p-value is out of range for accurate analysis, but in the region of 10^{-300}). Although both of these values are exceptionally bad, all other tests in the Rabbit battery pass without any difficulties and within expected values. The following figures show the statistics of the LCP for the first 10^6 bits of Ω.

Fig. 1 shows the histogram with the deviations between the expected (Eq. 1) and the observed values for the LCP of Ω. This has a very narrow distribution when compared to what is expected of a truly random one. A random sequence will produce a decreasing, but non-zero value across all x-axis values, whilst Ω is restricted to the -1.23 to 1.23 band.

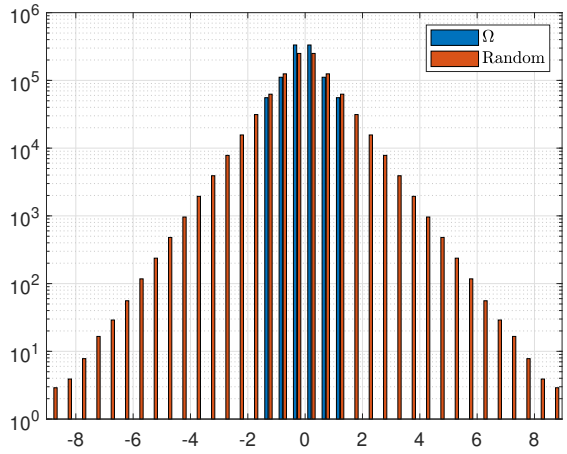


Figure 1: Histogram (log) of deviations between expected and observed LCP for first 10^6 bits.

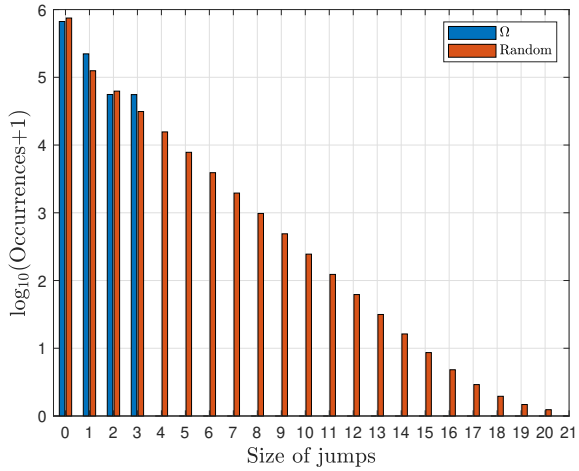


Figure 2: Histograms of LCP jumps, for the 10^6 first bits.

Fig. 2 shows the histogram (in logarithmic scale) of the LCP jumps for Ω, and compare it with the expected values for a random sequence. Since Ω does not exhibit jumps larger than 3, the total number of jumps is significantly higher for the LCP of Ω, approximately $(1/3)n$, than for a random sequence, which is around $(1/4)n$ (Eq. 4).

For these values, the corresponding Z_{score} and χ_{score} for the number of jumps and the distribution of these jumps, respectively, are $Z_{score} = 235$ and $\chi_{score} > 100000$ ($numbins = 14$). These values are very far from the expected values of $Z_{score} < 1.96$ and $\chi_{score} < 22.3$ to pass the test with $n = 10^6$ and $C = 95\%$.

IV. Hyper-sensitivity of TestU01 Linear Complexity test

In this section, we analyze the effect on the LCP when Ω is subtly modified. One would expect minor changes to the sequence to lead to minor changes to the test output, but we will see that this is not the case. These counterintuitive results cast a shadow of doubt about the adequacy of TestU01's Rabbit Linear Complexity test and on the whole Rabbit suite as well.

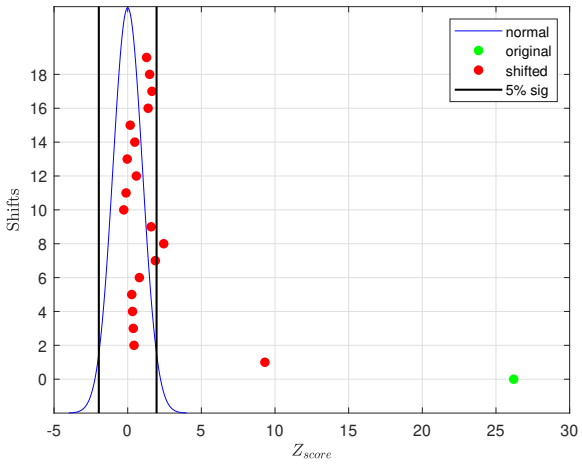


Figure 3: Z-score of the number of jumps for shifted Ω

A. Ω *shifts*

As discussed in the previous section, Ω presents an extremely poor behaviour from the point of view of the Linear Complexity test in TestU01's Rabbit, failing it on different implementations with astronomically low p-values. One would naturally expect that if Ω is shifted left i positions (i.e. i bits are removed from the beginning of the sequence and replaced with the following ones) the result of the Linear Complexity test should be the same that over the original Ω sequence, particularly if i is small compared to $|\Omega| = 10^6$. After all, both sequences are identical, except for the missing initial i bits, so it is within reason to expect that if Ω is astronomically failing the linear complexity test, the shifted version should do so as well. Surprisingly, that is not the case. This simple shift makes the LCP of the modified sequence approach that expected of a truly random sequence. The effective length of these sequences (as used in the *Scomp.c* test) is $n = 12000$. For this n, the expected number of bins is 8, and therefore the limit values to pass the test are $|Z_{score}| < 1.96$ and $\chi_{score} < 14$.

Fig. 3 shows the Z_{score} for the number of jumps for shifted versions of Ω. The scores for each 1-bit shift applied to the Ω sequence are represented by red or green dots. It takes very small shifts to cause the number of jumps to fall within acceptable bounds (5% of significance). Compared with shift=0 and shift=1, shift=2 is a dramatic improvement. After shift=10, the results of this sub-test are consistently within the confidence bound (disappearing the abnormally high number of jumps), indicating that this element of *Scomp.c* would be passed.

Fig. 4 shows the observed χ_{score} of the height of the jumps when testing the shifted Ω sequence. Shift=5 or greater is enough to pass this particular sub-test. The χ_{score} improves rapidly, but shift=4 or lower lay far outside of the confidence threshold. Though the improvement in sub-test results is not as rapid as that observed in Fig. 3, any shift greater than 5 consistently lies within the confidence interval of this sub-test.

These results indicate an unjustified and surprising input hyper-sensitivity in TestU01's Linear Complexity test; shifting as few as 10-bits leftwards allows a sequence like this to go from astronomically failing to easily passing these tests of

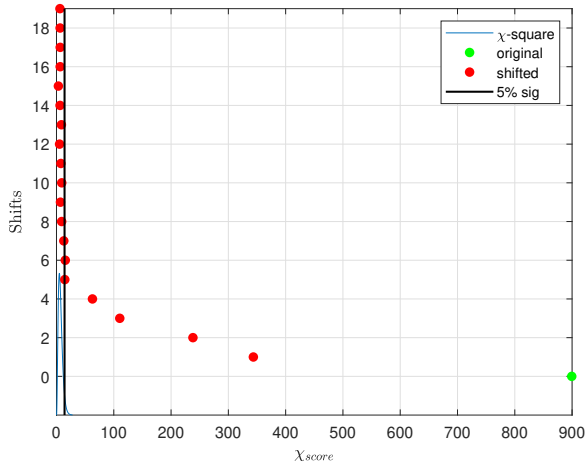


Figure 4: χ_{score} of the height of the jumps for shifted Ω

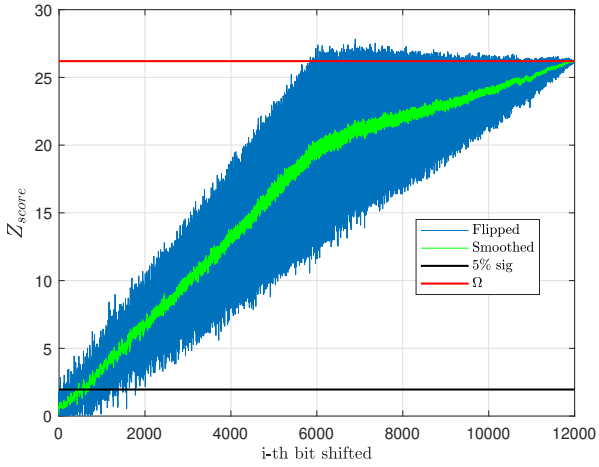


Figure 5: Z_{score} for Ω bit flips

randomness.

This is also quite damning, as it allows such sequences to defeat the Rabbit battery. The recipe is simple: just use the Ω sequence, which passes all other Rabbit tests but catastrophically fails the linear complexity test, because it has poor linear features, and shift it 5 bits to the left. Then you will have an equally poor sequence, with the same linear problems, but that this time fools all of Rabbit's randomness test. This can be observed in Table II, where we summarise the results of different shifts of Ω with TestU01. For reference; P=Pass, 5=Linear Complexity test failure (*both), 26=RandomWalk1 test failure. Shift=5 passes all tests, as do shift=8 and shift=9, and from shift=12 on, the shifted versions of Ω pass Rabbit consistently.

B. Ω *flips*

Another extremely simple transformation of Ω that should not essentially modify its randomness or linearity properties is to flip one of its bits (particular case of k-error linear complexity analysis [19]). Experimentally, it seems that flipping its low-order bits has a wider impact that changing bits later in the sequence. That is, flipping the low-order bits has an impressive effect, but such effect vanishes gradually when the order of the bit increases. In particular, the low-order bits of the Ω sequence can be flipped to generate derivative sequences which pass TestU01's Rabbit.

Fig. 5 and 6 show the evolution of the Z_{score} and χ_{score}, respectively, when the i-th bit of Ω is flipped. As the specific values change from one derivative sequence to other, smoothed values have been included for easy viewing of the tendencies. It is clear that the values of the derivatives sequences with the low-order bits flipped are similar to those exhibited in random sequences, and consequently pass the test, while this effect becomes less pronounced as bits of higher order are flipped, with values approaching to those of the original Ω, and sequences failing the test by wide margins. This finding supports the previous hypothesis that the first bits are the key contributor to the failure of the Rabbit battery by Ω sequence. That is, it highlights the hyper-sensitivity of *Scomp.c* to low-order bit manipulation.

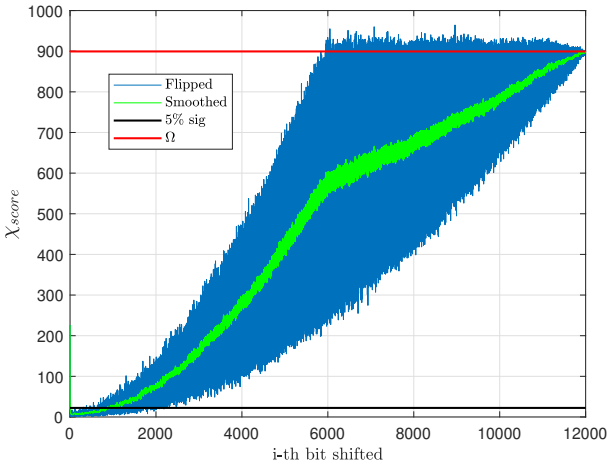


Figure 6: Number of jumps

V. Comparison of TestU01 and NIST SP800-22 Linear Complexity Tests

The Ω sequence has also been tested using the NIST implementation of SP800-22 [3], [14] (STS 2.1.2). Ten sub-sequences, each 100,000 bits in length, were tested. The proportion results reflect the number of these sub-sequences which pass the SP800-22 implementation of the Linear Complexity test. These sub-sequences are selected sequentially from the tested output, to a user-defined length. The total sequence length is therefore $L_i \cdot n$ where L_i is the length of one sub-sequences and n is the number of sub-sequences tested.

The Ω sequence and its derivatives passed the Linear Complexity test ($p = 0.7334$, proportion 10/10) but fails Maurer's Universal Statistical [22] test. This is a surprising result, as the Linear Complexity test of TestU01 was the only test which identified issues with Ω. The reason for this difference lies in the values assessed by each version of the test. TestU01's approach only checks the distribution of the LCP jumps against a model distribution known to represent a random sequence. NIST's approach calculates a Z_{score} derived from the number of jumps observed between sub-

Table II: Results of shifted Ω with TestU01's Rabbit

Shift	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
Errors	5*	5*	5	5	5,26	P	26	26	P	P	P	26	P	P	P	P	P	P	P	P

sequence complexity, providing the opportunity for the test to observe the uneven distribution of low-order complexity in the Ω sequence.

Thus, although the fact that the low-order bits of Ω and bit-flipped derivative sequences have no effect on the results is a positive finding in favour of the NIST test, SP800-22's implementation of the Linear Complexity test fails to identify Ω or its derivatives as non-random. However, SP800-22 consistently identifies Ω-derived sequences as non-random using Maurer's Universal Statistical test.

VI. Conclusion

This paper has shown that a sequence derived from the factorisation of natural numbers (Ω), though non-random and reported as so by *Scomp.c* in TestU01's Rabbit, can still pass statistical tests of randomness with some minor modifications, and hence casts serious doubts on the validity of these tests. TestU01's Rabbit battery is incapable of detecting these modified sequences, reporting results similar to those expected from a random sequence. SP800-22's implementation of the Linear Complexity test is unable to detect non-randomness in this sequence either.

Scomp.c exhibits a hyper-sensitivity to the initial bits of a sequence (since LCP does so). This is the reason for its inability to detect non-randomness in Ω variants. Inverting a single bit in Ω can result in a pass on this test. Shifting bits left has the same effect.

Comparison with SP800-22's implementation of the Linear Complexity test shows that the choice of statistics used to evaluate a sequence can vary greatly between different software. This has a significant impact on the outcome of the test; SP800-22 linear complexity test is not sensitive to low-order bit patterns, but also does not identify that Ω (or its derivatives) are non-random. TestU01's implementation is extremely sensitive to low-order bits, to the point where single bit flips can change the result of a test conducted over tens of thousands of bits. Both implementations have significant issues, as highlighted by these findings.

Acknowledgements

This work was possible thanks to the PID2022-138933OB-I00 ATQUE research project, and to the C065/23 Cybersecurity Chair of the University of La Laguna and INCIBE, funded by MCIN/AEI/10.13039/501100011033, and the Recovery, Transformation, and Resilience Plan (Next Generation), financed by the European Union.

References

[1] G. Marsaglia, "A current view of random number generators," in *Computer Science and Statistics, Sixteenth Symposium on the Interface. Elsevier Science Publishers, North-Holland, Amsterdam*, 1985, pp. 3–10.

[2] R. G. Brown, D. Eddelbuettel, and D. Bauer, "Dieharder: A random number test suite," *Open Source software library, under development*, 2013.

[3] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Booz-Allen and Hamilton Inc Mclean Va, Tech. Rep., 2001.

[4] S. Zhu, Y. Ma, T. Chen, J. Lin, and J. Jing, "Analysis and improvement of entropy estimators in NIST SP 800-90B for Non-IID entropy sources," *IACR Transactions on Symmetric Cryptology*, pp. 151–168, 2017.

[5] P. L'Ecuyer and R. Simard, "TestU01: A C library for empirical testing of random number generators," *ACM Transactions on Mathematical Software (TOMS)*, vol. 33, no. 4, p. 22, 2007.

[6] Secure Hash Standard, "FIPS Pub 180-1," *National Institute of Standards and Technology*, vol. 17, p. 15, 1995.

[7] ——, "FIPS PUB 180-2," *National Institute of Standards and Technology*, 2002.

[8] W. Killmann and W. Schindler, "AIS 31: Functionality classes and evaluation methodology for true (physical) random number generators, version 3.1," *Bundesamt fur Sicherheit in der Informationstechnik (BSI), Bonn*, 2001.

[9] D. Hurley-Smith and J. Hernandez-Castro, "Bias in the Mifare DESFire EV1 TRNG," in *International Workshop on Radio Frequency Identification: Security and Privacy Issues*. Springer, 2016, pp. 123–133.

[10] ——, "Certifiably Biased: An In-Depth Analysis of a Common Criteria EAL4+ Certified TRNG," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 4, pp. 1031–1041, 2018.

[11] ——, "Great Expectations: A Critique of Current Approaches to Random Number Generation Testing & Certification," in *International Conference on Research in Security Standardisation*. Springer, 2018, pp. 143–163.

[12] B. McCullough, "A review of TESTU01," *Journal of Applied Econometrics*, vol. 21, no. 5, pp. 677–682, 2006.

[13] D. G. Marangon, G. Vallone, and P. Villoresi, "Random bits, true and unbiased, from atmospheric turbulence," *Scientific reports*, vol. 4, p. 5490, 2014.

[14] A. Rukhin, J. Soto, and J. Nechvatal, "A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST DTIC Document," *NIST SP800-22*, 2010.

[15] E. Berlekamp, "Binary BCH codes for correcting multiple errors," *Algebraic Coding Theory*, 1968.

[16] J. Massey, "Shift-register synthesis and BCH decoding," *IEEE transactions on Information Theory*, vol. 15, no. 1, pp. 122–127, 1969.

[17] S. R. Blackburn, "Fast rational interpolation, Reed-Solomon decoding, and the linear complexity profiles of sequences," *IEEE Trans. Information Theory*, vol. 43, no. 2, pp. 537–548, 1997. [Online]. Available: https://doi.org/10.1109/18.556111

[18] R. Games and A. Chan, "A fast algorithm for determining the complexity of a binary sequence with period2^n(Corresp.)," *IEEE Transactions on Information Theory*, vol. 29, no. 1, pp. 144–146, January 1983.

[19] H. Chen, "Fast algorithms for determining the linear complexity of sequences over GF(p/sup m/) with period 2/sup t/n," *IEEE Transactions on Information Theory*, vol. 51, no. 5, pp. 1854–1856, May 2005.

[20] G. Carter, "Aspects of local linear complexity. Ph.D. thesis," Ph.D. dissertation, University of London, 1989.

[21] L. OBRÁTIL, "The automated testing of randomness with multiple statistical batteries," Ph.D. dissertation, Masarykova univerzita, Fakulta informatiky, 2017.

[22] U. M. Maurer, "A universal statistical test for random bit generators," *Journal of cryptology*, vol. 5, no. 2, pp. 89–105, 1992.

Identifying Cybersecurity Vulnerabilities in Water and Resource Recovery Facilities (WRRFs)

Ignasi Rodríguez-Roda
LEQUIA, Institute of the Environment, University of Girona
C/ Mª Aurèlia Capmany, 69, 17003 Girona, Spain
ignasi.rodriguezroda@udg.edu

David Palma-Heredia
LEQUIA, Institute of the Environment, University of Girona
C/ Mª Aurèlia Capmany, 69, 17003 Girona, Spain

Laura Arantegui
Càtedra INCIBE, UdG

Sandra Bonillo
Càtedra INCIBE, UdG

Steven Kemp
Grup de Recerca en Justícia i Penologia
Departament de Dret Públic, UdG

***Abstract*—Cybersecurity in Water and Resource Recovery Facilities (WRRFs) has gained importance as recent trends in digitalisation have increased operational efficiency but also exposure and vulnerability to cyber attacks. Through a literature review and interviews with Spanish experts, current top cyber threats to WRRFs and water sector cybersecurity maturity are examined. Results show a lack of available data on cyber threats, as well as a lack of expertise in cybersecurity and that management of the digital infrastructure of WRRFs can be improved. The new legislative framework regarding cyber-incidents underscores that it is necessary to carefully map the digital infrastructure and minimise human error to foster better protection against cyber-threats in WRRFs. This research stems from the Water Digitalization and Cybersecurity Chair (C014/24), the result of a collaboration agreement between the National Cybersecurity Institute (INCIBE) and the University of Girona. This initiative is carried out within the framework of the Recovery, Transformation and Resilience Plan, funded by the European Union (Next Generation).**

***Index Terms*—critical infrastructures; cybersecurity; digitalization; wastewater; WRRF**

I. Introduction

Digital transformation of water systems allows water infrastructures to operate more efficiently [1–3]. In particular, Water and Resource Recovery Facilities (WRRFs) have been greatly enhanced through the implementation of digital technologies in both physical and biological processes. Various examples can be found in the use and optimisation of automated control systems for complex processes such as aeration, or performance prediction from a dynamic influent.

However, from the perspective of cyber-physical-human-systems, concerns have arisen regarding cybersecurity in critical infrastructure like the urban water cycle because of the potential damage attacks can cause to society [4, 5]. Critical infrastructure refers to any asset or system, which is essential for the maintenance of vital societal functions, health, safety, security, economic or the well-being of people, and the disruption or destruction of which would have a significant impact as result of the failure to maintain those functions (EU briefing 2021). It comprises sectors, such as energy or finance, as specified in the legislative framework known as NIS1 (Network and Information Systems), which was recently expanded with the introduction of the NIS2 legal framework through Directive (EU) 2022/2555. This is relevant to WRRFs because, whilst the NIS1 only included the drinking water sector, the NIS2 also includes the wastewater sector [6]. Thus, WRRFs currently face increasing pressure to adapt to both cybersecurity threats and the corresponding legislative framework.

The objective of this work is to identify the most common types of cyberattacks against WRRFs, as well as the typical vulnerabilities within WRRFs that allow these to occur. Moreover, we will analyse the cybersecurity maturity of the water sector to deal with current threats and what can be done to improve responses.

II. Materials and Methods

Cyber-events databases from different countries, technical reports and scientific publications were explored to gather evidence data of the frequency and impact of cyber-attacks on WRRFs. Also, interviews with expert practitioners from the water and wastewater sector in Spain were conducted to identify sector-specific concerns and threats from the practitioners' perspective.

III. Results and Discussion

A. Vulnerability of WRRFs to cyber-attacks

Identification of digital assets of WRRFs is a crucial task to understand where and how cyber attacks might occur.

As water utilities are digitalized, they are also getting involved into the IT-OT-IoT convergence paradigm. The fundamental proposal about this convergence is that all digital devices must be organised and connected as one system [7]. The IoT framework proposes, for example, four main layers of sensing, communication network, data processing and applications (and in some cases a fifth layer regarding business and customer management). This convergence has been given increasing attention because of its potential to enhance supply chains [8].

As shown in Figure 1, cybersecurity threats such as ransomware and DDoS affect elements from both IT and OT-related layers (mostly they target key elements such as the communication network). However, there is a lack of fundamental knowledge about cybersecurity in the water sector [9]. This situation can be another barrier for the correct implementation of Water and Resource Recovery Facilities (WRRF) technologies.

Ransomware is often introduced via social engineering techniques such as phishing. It usually targets servers containing sensitive data and OT-monitoring and control systems

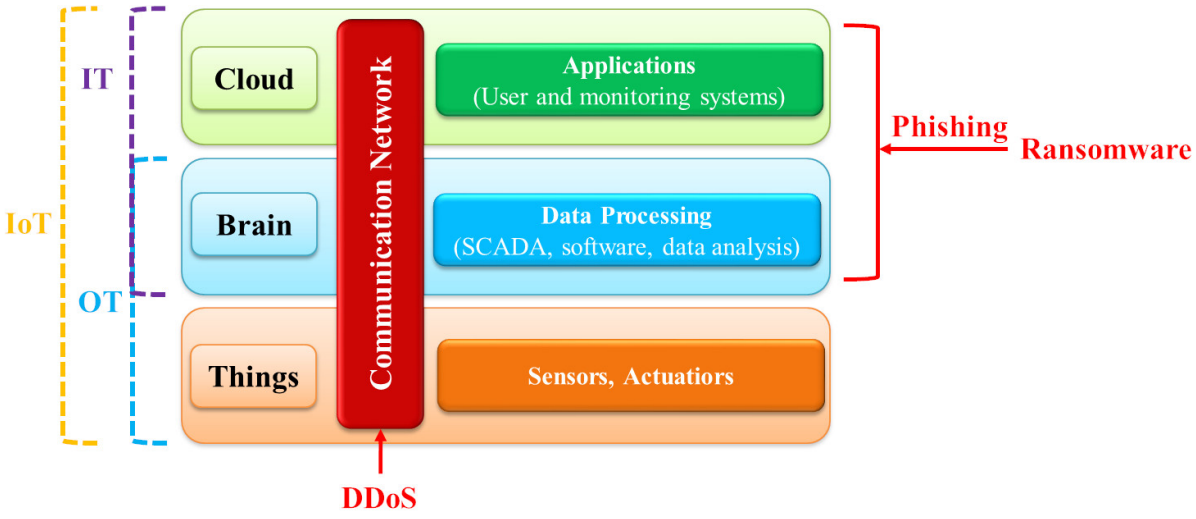


Figure 1. IT, OT and IoT convergence. IoT layers (Sensors, Communication, Data Processing and Applications) are intertwined between the IT and OT conventional frameworks.

such as SCADAs. On the other hand, DDoS mostly aims to disrupt communication network in order to freeze the target's operations. Both DDoS and ransomware can lead to threatening scenarios. For example, through ransomware, cyber-attackers can access control systems and modify operational parameters, whilst DDoS disables the communication network of the victim so it cannot correct the controlling malfunction. This combination of OT-sequestration and manipulation and network disabling can lead to potential widespread operational damage to the plant and/or to the environment. As the IT environment is steadily increasing protection against vulnerabilities, the professionals interviewed fear that the hackers could turn to OT devices, performing attacks which could have devastating effects not only in economic or reputational terms, but also for public health.

B. Data availability of cyber-attacks at critical infrastructures

The following databases were accessed: the Cyber Events Database [10], with a total of 13,471 cases for the period 2018-2023, out of which 3,577 took place in Europe, and from those only 19 were related to the water and wastewater sector; the European Repository of Cyber Incidents [11], which contains 2,658 cases for the years 2000-2023, 965 affecting the European region and only 18 related with the water and wastewater sector.

According to the ENISA Threat Landscape 2024 report [12], the top three incidents by threat type for last year were DDoS (41%), ransomware (26%) and data-related incidents (19%) against critical infrastructures.

There were reports of several DDoS threat type incidents affecting the OT of multiple water utilities in the US, a wastewater plant in Poland, a hydroelectric dam in France and water pumps in Israel. These are some examples of DDoS-related vulnerabilities in the water sector, which highly depends on the performance of digital technologies.

C. Interviews

15 interviews with professionals from the Spanish water and wastewater sector. According to results of these interviews, there is a consensus about cyber-attacks becoming increasingly frequent in these critical infrastructures. However, there is a lack of publicly available data. Keeping a high level of awareness about the damaging potential of the attacks is crucial. Furthermore, as with all cyber-attacks against critical infrastructures, they can have a very negative effect on public opinion. This may be why these events have rarely been registered and, thus, there is a lack of historical data regarding cyber-attacks against water infrastructures. Current legislative changes, like the European Directive NIS2, aim to foster transparency and quality data collection about cyber-attacks on critical infrastructures. Nonetheless, there is very little knowledge about the NIS2 requirements among the professionals in the water sector.

Some of the most common consequences of lack of knowledge and awareness about cybersecurity in the WRRF sector are:

1) Digital infrastructures may be outdated and frequently poorly managed (such as fragmentation of digital assets in independent IT and OT departments).
2) Lack of specialized staff with knowledge of cybersecurity or mapping of digital assets.
3) Lack of tools to detect cyber-attacks, so they often go unnoticed.

IV. Conclusions

WRRF vulnerabilities have been identified across different types of digital assets. Some of these vulnerabilities come from a lack of knowledge and awareness of the cybersecurity threat, but there is already enough evidence to raise the alarm. However, further data regarding WRRFs cyber attacks and more in-depth analysis of responses from the interviewed practitioners' are required. Still, some preliminary conclusions can be drawn in this regard.

First, it is necessary to know a WRRFs' digital assets in detail: mapping all the elements, connections and data flows is the first step to understanding how a cyber-attack can impact the WRRF. Furthermore, it is recommended to set aside the increasingly obsolete concepts of IT and OT and adopt IoT

frameworks to enhance management of the digital system as a whole.

On the other hand, phishing is the most frequent way for ransomware cyber-attacks, and it appeals directly to professionals through social engineering and impersonation. The common proposed solutions against phishing involve user activity registers and careful supervision of human resources. However, protection ultimately relies on the acquisition of minimal cybersecurity awareness and knowledge by all professionals, users and entities interacting with any part of the IoT architecture, as well as internal reaction protocols that limit the potential success of social engineering attacks. The mandatory compliance with the NIS2 Directive is regarded by professionals in the sector as an opportunity for a systematic update of cybersecurity measures, which would also benefit from the support of senior management thanks to the requirement of this legal framework. Periodic external audits are a very important pillar for identifying shortcomings, as are the assessments provided by consultancies specialized in NIS2 requirements. The final transposition of the Directive, expected by the end of 2025, has not yet taken place in Spain, and this will considerably delay the sector's updating process. However, one of the expectations is that the National Security Scheme (ENS) will act as an essential baseline, providing certain compliance guarantees to entities that already follow its guidelines.

Acknowledgements

The authors would like to thank the INCIBE chair of Digitalization and Water Cybersecurity and the TED2021 AEI CLEPSIDRA project. LEQUIA has been recognized as consolidated research groups by the Catalan Government, code 2021-SGR-1352.

References

[1] E. Torfs, N. Nicolaï, S. Daneshgar, J. B. Copp, H. Haimi, D. Ikumi, B. Johnson, B. B. Plosz, S. Snowling, L. R. Townley *et al.*, "The transition of wrrf models to digital twin applications," *Water Science and Technology*, vol. 85, no. 10, pp. 2840–2853, 2022.

[2] P. Małka, K. Gaska, E. Wysowska, K. Kudlik, and J. Ciuła, "Intelligent infrastructure (critical) of the water supply network for collective water supply systems–a case study," *Desalination and Water Treatment*, vol. 274, pp. 1–6, 2022.

[3] M. Moy de Vitry, M. Y. Schneider, O. F. Wani, L. Manny, J. P. Leitão, and S. Eggimann, "Smart urban water systems: what could possibly go wrong?" *Environmental Research Letters*, vol. 14, no. 8, p. 081001, 2019.

[4] M. Malatji, A. L. Marnewick, and S. von Solms, "Cybersecurity policy and the legislative context of the water and wastewater sector in south africa," *Sustainability*, vol. 13, no. 1, p. 291, 2020.

[5] J. You, "Strengthening cybersecurity of water infrastructure through legislative actions," *JAWRA Journal of the American Water Resources Association*, vol. 58, no. 2, pp. 282–288, 2022.

[6] European Commission. (2024) The NIS2 Directive Explained. European Commission. [Online]. [Online]. Available: https://nis2directive.eu/

[7] C. Ebert, "Convergence of it, ot, and the iot," *Computer*, vol. 57, no. 8, pp. 108–112, 2024.

[8] M. Rahimi, M. Maghsoudi, and S. Shokouhyar, "The convergence of iot and sustainability in global supply chains: Patterns, trends, and future directions," *Computers & Industrial Engineering*, vol. 197, p. 110631, 2024.

[9] R. Thomani, A. Marnewick, S. von Solms, and M. Malatji, "Cybersecurity knowledge requirements for a water sector employee," in *International Symposium on Human Aspects of Information Security and Assurance*. Springer, 2022, pp. 91–105.

[10] Center for International and Security Studies at Maryland (CISSM). (2024) Cyber Events Database. School of Public Policy, University of Maryland. [Online]. [Online]. Available: https://cissm.umd.edu/research-impact/publications/cyber-events-database-home

[11] (2022) European Repository of Cyber Incidents (EuRepoC). European Repository of Cyber Incidents. [Online]. [Online]. Available: https://eurepoc.eu/database/

[12] European Union Agency for Network and Information Security (ENISA), "ENISA Threat Landscape 2024," https://www.enisa.europa.eu/publications/enisa-threat-landscape-2024, European Union Agency for Network and Information Security (ENISA), Tech. Rep., 2024, [Online].

Auditoría de Seguridad en Dispositivos IoT BLE: Técnicas de Sniffing Pasivo y Ataques GATT

Jose Aveleira-Mata
Grupo de Invest. SECOMUCI
Universidad de León
dnarr@unileon.es

Diego Narciandi-Rodríguez
Grupo de Invest. SECOMUCI
Universidad de León
jose.aveleira@unileon.es

Martín Bayón-Gutiérrez
Inst. de Invest. RIASC
Universidad de León
martin.bayon@unileon.es

Natalia Prieto-Fernández
Grupo de Invest. SECOMUCI
Universidad de León
nprif@unileon.es

Isaías García-Rodríguez
Grupo de Invest. ALBA
Universidad de León
isaias.garcia@unileon.es

Carmen Benavides
Inst. de Invest. RIASC
Universidad de León
carmen.benavides@unileon.es

***Abstract*—Este trabajo presenta un banco de pruebas reproducible para el análisis de seguridad en dispositivos IoT basados en Bluetooth Low Energy (BLE). El entorno se compone de tres nodos ESP32 con diferentes configuraciones de seguridad y de una estación de análisis equipada con nRF Sniffer y Wireshark, lo que permite capturar de forma pasiva sesiones completas del perfil de atributos genérico (GATT). Sobre este testbed se implementan ataques en tres fases: reconocimiento, escucha pasiva del tráfico y explotación activa mediante manipulación de comandos e inyección de cargas malformadas. A partir de las capturas se construye un conjunto de datos etiquetado de tráfico BLE–GATT de forma similar a escenarios reales, que incluye características temporales y de protocolo. El objetivo es facilitar la evaluación de técnicas de detección de anomalías y el endurecimiento de implementaciones BLE de bajo coste.**

***Index Terms*—Bluetooth Low Energy, IoT, GATT, sniffing pasivo, ataques a dispositivos embebidos, detección de anomalías, datasets de tráfico de red**

I. Introducción

Bluetooth Low Energy (BLE) se ha consolidado en la última década como una de las tecnologías inalámbricas de corto alcance más extendidas en el ecosistema de Internet de las Cosas (IoT). [1] Diseñada por el Bluetooth Special Interest Group (SIG) para aplicaciones de bajo consumo y bajo coste, BLE permite mantener alcances de comunicación similares a Bluetooth clásico reduciendo drásticamente el gasto energético y el tamaño de los dispositivos. Esta combinación de bajo consumo, integración nativa en la mayoría de sistemas operativos móviles y disponibilidad de chipsets económicos ha favorecido su adopción masiva en ámbitos como salud y bienestar, deporte y fitness, domótica, balizas de localización, entretenimiento en el hogar y soluciones de seguimiento de activos. [2]

En este contexto, multitud de dispositivos IoT de propósito general (pulseras de actividad, balizas, sensores ambientales, cerraduras inteligentes o pequeños robots educativos, entre otros) utilizan BLE como canal principal de comunicación con aplicaciones móviles o pasarelas domésticas [3], [4]. El modelo basado en el perfil GATT Generic Atribute Profile (perfil de atributos Genéricos), permite implementar fácilmente servicios de lectura y escritura de datos, notificaciones de eventos y control remoto de actuadores. Esta flexibilidad ha convertido BLE en un componente clave de muchas arquitecturas IoT de bajo coste y recursos limitados.

Sin embargo, esta ubicuidad ha venido acompañada de una creciente preocupación por la seguridad y la privacidad de las comunicaciones BLE. La literatura reciente recoge múltiples vulnerabilidades asociadas tanto al propio estándar como, sobre todo, a implementaciones incompletas o mal configuradas en dispositivos comerciales. [5], [6] Diversos trabajos y encuestas sistemáticas han puesto de manifiesto el uso de mecanismos de emparejamiento débiles (por ejemplo, Just Works), la ausencia de autenticación robusta y configuraciones de cifrado opcional que, en la práctica, se desactivan o se emplean de forma incorrecta, dejando el canal expuesto a ataques de escucha pasiva, suplantación e inyección de mensajes [7].

Los dispositivos de consumo dirigidos a usuarios finales no expertos (como wearables, dispositivos de salud personales o juguetes conectados) constituyen un caso especialmente sensible. Auditorías de seguridad sobre este tipo de productos han identificado prácticas inseguras en sus interfaces inalámbricas, incluyendo el uso de direcciones estáticas rastreables, aceptación de conexiones no autorizadas y falta de protección adecuada en las comunicaciones, lo que facilita la interceptación del tráfico y el control no autorizado del dispositivo [8], [9]. Específicamente, una investigación reciente, llevada a cabo en el contexto de una campaña de concienciación navideña promovida por el Instituto Nacional de Ciberseguridad, analizó 26 juguetes conectados, incluyendo aquellos con conectividad Bluetooth Low Energy, mediante técnicas de *sniffing* de tráfico BLE. El objetivo fue determinar si las comunicaciones se establecían de forma cifrada. Los resultados indicaron que, en un número considerable de dispositivos, el tráfico BLE se transmitía en texto plano o con mecanismos de autenticación rudimentarios. Esta deficiencia expone datos potencialmente sensibles y permite a un atacante en proximidad observar e incluso modificar los mensajes intercambiados [10].

Una parte de estos problemas se explica por las propias características del ecosistema: muchos de estos dispositivos

son de bajo coste, con capacidades de cómputo y energía limitadas, y se desarrollan bajo una fuerte presión de tiempo de salida al mercado [11]. En estas condiciones, las funcionalidades de seguridad proporcionadas por el estándar BLE (como los distintos modos de emparejamiento y cifrado) se perciben a menudo como opcionales o costosas de integrar, y se opta por configuraciones mínimas (por ejemplo, ausencia de interfaz de usuario para introducir PINs o claves, reliance en Just Works, o enlaces sin cifrado) que reducen la superficie de interacción pero dejan el canal de comunicación expuesto. Trabajos específicos sobre dispositivos BLE sin capacidades de entrada/salida han demostrado que es posible diseñar mecanismos de autenticación ligeros, lo que sugiere que estas decisiones responden más a prioridades de diseño y negocio que a limitaciones técnicas insalvables [12], [13].

En términos de seguridad, la consecuencia directa de estas decisiones es que el enlace BLE puede convertirse en una superficie de ataque relevante. Un atacante en rango de radio puede realizar eavesdropping sobre tráfico no cifrado, inferir el protocolo de aplicación y, en muchos casos, establecer conexiones no autorizadas para enviar comandos arbitrarios al dispositivo [14]. Más aún, cuando los datos recibidos por BLE son procesados por una aplicación móvil o por servicios en la nube, los cuales constituyen un punto de entrada potencial para actividades maliciosas: si la aplicación no valida adecuadamente dichos datos, los transforma en estructuras o expresiones ejecutables, o incorpora vulnerabilidades clásicas de gestión de memoria, el canal BLE se convierte en un vector para explotar fallos lógicos o incluso vulnerabilidades de mayor impacto, dependiendo de la implementación concreta. [15]

Motivado por estos resultados y por observaciones previas que evidencian la ausencia de cifrado en una parte significativa de los dispositivos IoT basados en BLE de bajo coste, este trabajo propone un entorno experimental controlado, basado en microcontroladores ESP32, para analizar de forma sistemática los riesgos asociados a comunicaciones BLE sin cifrado ni autenticación fuerte. A partir de este banco de pruebas, se evalúan escenarios de escucha pasiva, *replay* e inyección de comandos sobre enlaces BLE en claro, con el objetivo de caracterizar su impacto y, al mismo tiempo, generar trazas de tráfico BLE etiquetadas que puedan utilizarse como *datasets* para entrenar y evaluar sistemas de detección de intrusiones y mecanismos de identificación de patrones de ataque.

II. Métodos y materiales

En el entorno experimental se utilizaron tres placas de desarrollo basadas en el microcontrolador ESP32 (Espressif Systems), con CPU Xtensa LX6 de 32 bits a 240 MHz, 520 KB de SRAM y 4 MB de memoria Flash externa. Todas ellas integran conectividad Bluetooth dual mode (Bluetooth Classic y Bluetooth Low Energy 4.2) [16].

Cada ESP32 asumió un rol específico en la topología BLE Tabla I

El firmware de los dispositivos se desarrolló con Arduino Core for ESP32 versión 2.0.14, utilizando la pila NimBLE (versión 1.4.0) como implementación de Bluetooth Low Energy. Esta combinación permitió configurar de forma flexible los roles GAP (central y peripheral) y definir servicios y características GATT [17] personalizados, manteniendo un consumo de memoria ajustado y un código fácilmente reproducible.

A. Servicios GATT y protocolo de aplicación

Los periféricos ESP32_P1 y ESP32_P2 exponen un servicio GATT propietario que emula un dispositivo IoT controlado por una aplicación móvil. Este servicio agrupa la funcionalidad básica del dispositivo en torno a dos características principales:

Por un lado, una **característica de comandos**, utilizada por el nodo central para enviar órdenes al dispositivo. Esta característica admite operaciones de escritura con y sin respuesta (WRITE y WRITE_NO_RESPONSE) y se ha configurado para permitir la escritura sin requisitos de autenticación, de forma que cualquier cliente conectado puede transmitir comandos válidos.

Por otro lado, una **característica de estado y notificaciones**, a través de la cual el periférico informa de su estado interno y de los cambios relevantes en la operación. Esta característica puede leerse y, además, permite al dispositivo enviar notificaciones asíncronas al central cuando se produce un evento o se actualiza alguna variable interna. Al igual que en el caso anterior, el acceso se ha configurado sin autenticación, lo que facilita el análisis del tráfico y, al mismo tiempo, reproduce el comportamiento observado en numerosos dispositivos BLE de bajo coste [18].

B. Hardware de captura y estación de análisis

Para la captura pasiva del tráfico BLE se emplearon dos dongles nRF52840 (Nordic Semiconductor, modelo PCA10059) configurados con el firmware oficial nRF Sniffer for Bluetooth LE versión 4.1.1. Ambos dispositivos se conectaron como interfaces de captura externas en Wireshark mediante el mecanismo extcap.

El primer sniffer se configuró en modo de captura general sobre el entorno de laboratorio, siguiendo la conexión entre el nodo central y los periféricos y registrando de forma continua todo el tráfico relevante. Las trazas generadas con este dispositivo se utilizaron principalmente para la construcción de los conjuntos de datos, ya que incluyen tanto el comportamiento normal del sistema como las sesiones bajo ataque.

El segundo sniffer se utilizó como punto de observación específico para la preparación de los ataques desde el

TABLE I
Dispositivos y configuración de seguridad en el testbed BLE.

Dispositivo	Rol en el testbed	Seguridad	Dirección MAC BLE
ESP32 #1	Dispositivo sin autenticación (ESP32_P1)	Sin autenticación ni cifrado	Dirección aleatoria resolvible
ESP32 #2	Dispositivo con autenticación débil (ESP32_P2)	Emparejamiento con PIN de 6 dígitos en claro	Dirección aleatoria resolvible
ESP32 #3	Central / maestro legítimo	Credenciales embebidas en el cliente	Dirección aleatoria resolvible

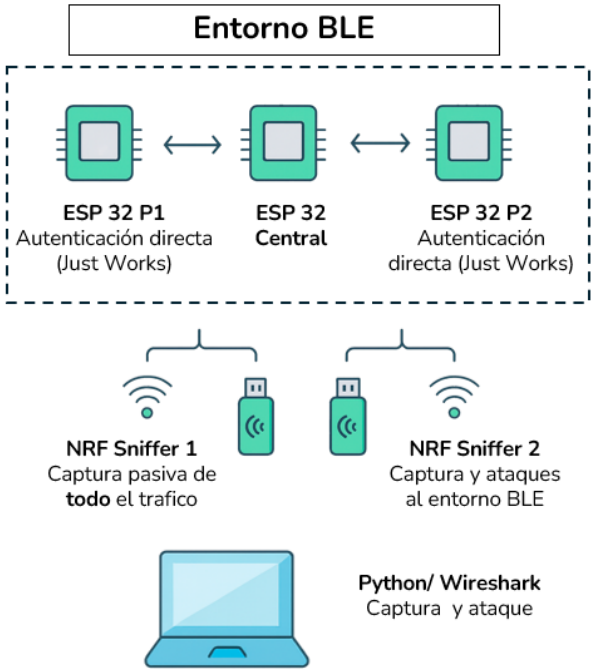


Fig. 1. Entorno BLE captura y ataques

equipo de escritorio. Este dispositivo se configuró de forma que permitiera capturar sesiones individuales y analizar con detalle las secuencias de comandos intercambiadas durante las comunicaciones legítimas. A partir de estas capturas se extrajeron los patrones necesarios para implementar posteriormente los ataques de replay e inyección de comandos desde el PC.

La estación de análisis fue un equipo con sistema operativo Windows 11 Pro, sobre el que se instaló Wireshark 4.2.0 y su herramienta en línea de comandos tshark, utilizada para la extracción automatizada de campos y la conversión de las capturas PCAP a formatos tabulares. El procesado posterior de los datos se realizó con Python 3.11, utilizando principalmente las bibliotecas bleak para la interacción BLE, pandas y numpy.

Aunque el entorno se ejecutó sobre una plataforma Windows, la configuración descrita es extrapolable a sistemas GNU/Linux que dispongan de soporte para Wireshark, nRF Sniffer y Python, por lo que el banco de pruebas puede reproducirse en otros entornos de análisis con cambios mínimos.

III. Entorno de pruebas y ataques

Sobre la base de la configuración hardware descrita en la sección anterior, en este apartado se presenta el entorno de pruebas sobre el que se ejecutan los distintos escenarios de ataque. La Figura 1 resume la topología experimental del banco de pruebas, formada por los tres dispositivos ESP32, la estación de análisis/atacante y los dos dongles nRF52840 destinados a la captura simultánea de tráfico BLE. El diagrama permite visualizar de forma compacta los roles de cada nodo, los enlaces BLE establecidos y los puntos de observación desde los que se registran tanto las comunicaciones legítimas como las sesiones bajo ataque.

Sobre este entorno se definió un flujo de ataque en **tres fases sucesivas**. En la primera, el atacante realiza el reconocimiento del entorno y la enumeración de los servicios y características GATT expuestos; en la segunda, captura y analiza los comandos legítimos intercambiados entre el central y los periféricos; y en la tercera, utiliza dicha información para llevar a cabo ataques activos de replay e inyección de comandos, explotando la ausencia de cifrado y autenticación fuerte.

A. Fase 1 – Reconocimiento y enumeración de dispositivos y servicios GATT

En una primera fase se lleva a cabo el reconocimiento del entorno BLE y la enumeración de los servicios GATT expuestos por los dispositivos detectados. Para ello se emplea el script ble_scanner.py, basado en la librería *Bleak*, que realiza un escaneo pasivo de tramas de *advertising* durante intervalos de entre 10 y 30 segundos. A partir de estas capturas, el escáner construye un listado de todos los nodos BLE visibles y permite filtrarlos por distintos criterios. En el contexto de este trabajo se seleccionan principalmente los ESP32, utilizando filtros por nombre lógico (cadenas que contienen “ESP32” o “IoT”), por prefijo de dirección MAC asociado al fabricante Espressif (EC:E3:34) y por la presencia de un UUID de servicio propietario. No obstante, un atacante real podría utilizar el mismo procedimiento para elegir, dentro de ese listado, cualquier otro dispositivo objetivo en función de su nombre comercial, fabricante o servicios anunciados.

Una vez seleccionados los objetivos, el escáner establece conexiones BLE temporales con cada dispositivo y procede a la enumeración sistemática de su jerarquía GATT, registrando para cada servicio y característica su UUID, *handle* y propiedades de acceso (READ, WRITE, NOTIFY, etc.). Con esta información se identifican tres tipos de debilidades especialmente relevantes para las fases posteriores. Además, se recopilan métricas auxiliares como el RSSI promedio y la estabilidad de la señal, junto con el listado completo de servicios, características y *handles* relevantes, que se almacenan en el fichero scan_results.json para su uso en las fases de captura de comandos y explotación activa.

B. Fase 2 – Captura pasiva de tráfico y extracción de comandos

En la segunda fase se analiza el tráfico intercambiado entre el central legítimo y los periféricos mediante captura pasiva sobre el enlace BLE. Para ello se emplea un dongle nRF52840 con el firmware nRF Sniffer for Bluetooth LE, configurado en modo *follow* sobre la dirección MAC del periférico objetivo. Una vez iniciada la captura, se fuerza el establecimiento de una nueva conexión entre el ESP32 central y el periférico, de modo que el sniffer pueda registrar el paquete CONNECT_IND, sincronizarse con la secuencia de *frequency hopping* y seguir la sesión completa. Durante varios minutos se deja operar el sistema en condiciones normales y se almacena la traza resultante en un fichero .pcapng.

El foco del análisis se sitúa en las operaciones GATT, ya que en BLE todo el protocolo de aplicación se materializa como lecturas, escrituras y notificaciones sobre características. A partir de las capturas se extraen, mediante tshark y el script analyze_pcap_commands.py, los paquetes ATT de tipo *Write Command* junto con sus *handles* y cargas útiles. Para

ello se emplean filtros sobre el protocolo ATT (btatt en la sintaxis de Wireshark), lo que permite aislar de la traza únicamente las operaciones GATT relevantes (por ejemplo, btatt.opcode == 0x52 para las escrituras sin respuesta). Esta información permite reconstruir los comandos de alto nivel enviados al dispositivo. De este modo, la fase 2 proporciona evidencia de que un observador pasivo, sin interferir en la comunicación legítima, puede recuperar la semántica del protocolo propietario y preparar los ataques activos que se implementan en la fase 3.

C. Fase 3 Ataques activos y explotación de vulnerabilidades

En la tercera fase se evalúa de forma práctica el impacto de las debilidades identificadas en las fases anteriores mediante la ejecución de ataques activos sobre el periférico BLE.

Ataque de manipulación de comandos GATT, en el que un cliente malicioso, tras establecer una conexión GATT no autenticada con el periférico [19]. , reutiliza los *handles* y cargas útiles identificados en la fase de captura para enviar secuencias de comandos SET_MODE, SET_BRIGHTNESS y RESET. Según la combinación y la frecuencia de envío, estos comandos permiten forzar modos de funcionamiento no deseados, degradar el servicio mediante agotamiento de recursos (por ejemplo, incremento del consumo energético) y provocar ciclos de reinicio que afectan directamente a la disponibilidad del dispositivo. Todos los ataques comparten el mismo vector subyacente, la exposición de características de control críticas accesibles mediante escrituras GATT sin cifrado ni autenticación, lo que facilita que cualquier dispositivo dentro del rango radio pueda tomar el control efectivo del periférico.

Inyección de payloads maliciosos y fuzzing de parsers GATT se ejecuta directamente sobre la característica de comandos del periférico BLE, aprovechando que acepta escrituras sin autenticación. Una vez establecida la conexión GATT, el atacante comienza a sustituir los comandos legítimos por secuencias de bytes deliberadamente anómalas: mensajes mucho más largos de lo esperado, cadenas con formatos extraños, caracteres de control, valores numéricos extremos o secuencias de texto que no respetan una codificación válida. Este escenario no solo es relevante para el propio ESP32, sino también como posible punto de entrada hacia las aplicaciones móviles que consumen esos mismos datos: si el dispositivo almacena o refleja los mensajes recibidos y la app asume que siempre están bien formados [20], [21].

IV. Resultados y generación del dataset

Para la obtención de las trazas se configuró en Wireshark la interfaz **nRF Sniffer for Bluetooth LE** en modo *follow* sobre la dirección MAC del periférico objetivo y, a continuación, se inició la captura antes de forzar una nueva conexión entre el ESP32 central y el periférico. De este modo, el sniffer pudo registrar el paquete CONNECT_IND, seguir toda la sesión BLE (incluyendo las operaciones GATT de interés) y almacenar el tráfico resultante en ficheros .pcapng para su posterior análisis.

A partir de las capturas en formato .pcapng, el tráfico se procesó de forma automatizada mediante tshark, extrayendo un conjunto de campos centrado en el análisis de sesiones GATT: identificadores temporales (frame.number, frame.time_epoch), tamaño de los paquetes (frame.len, btle.length), direcciones del central y del periférico (btle.central_bd_addr, btle.peripheral_bd_addr), parámetros del enlace (btle.access_address, btle.data_header.llid) y, de forma destacada, los campos ATT/GATT (btatt.opcode, btatt.handle, btatt.value). Sobre esta salida se aplicó un filtrado que descarta *beacons* de *advertising*, tramas malformadas y paquetes sin contexto de conexión, conservando únicamente el tráfico asociado a sesiones BLE activas. Posteriormente, los *opcodes* ATT se normalizaron a formato entero y se incorporó una característica temporal adicional, el tiempo entre paquetes consecutivos (*inter_arrival_time*), con el objetivo de capturar patrones de comportamiento en la secuencia de mensajes.

El resultado es un dataset en formato CSV con 26 465 instancias y 13 características Tabla II, etiquetadas como tráfico normal o de ataque a partir de los intervalos temporales en los que se ejecutaron los distintos escenarios de explotación. La distribución de clases refleja un desbalance similar al observado en entornos reales: 25 209 registros corresponden a tráfico legítimo (95,25 %) y 1 256 a tráfico bajo ataque (4,75 %). Además, la frecuencia de ciertos *opcodes* difiere entre ambos contextos, ya que algunos ataques generan un número relativamente reducido de tramas pero altamente informativas (por ejemplo, secuencias concentradas de operaciones de escritura), mientras que el tráfico normal incorpora un mayor volumen de peticiones de descubrimiento y lectura. Estas características hacen que el dataset sea adecuado para evaluar técnicas de detección de anomalías y clasificación en escenarios BLE con presencia de clases minoritarias

Disponibilidad del código y datos: todo el código utilizado para el desarrollo del testbed, la implementación de los ataques y la generación del conjunto de datos está disponible en https://github.com/joseAveleira/seguridad_BLE_ataques_GATT.

V. Conclusiones y trabajo futuro

Los resultados de esta investigación demuestran la viabilidad de ataques de inyección y manipulación de comandos en dispositivos BLE carentes de cifrado y autenticación. El banco de pruebas propuesto, basado en microcontroladores ESP32 y en un conjunto de ataques estructurados en fases (reconocimiento, captura pasiva y explotación activa), ha permitido reproducir de forma controlada situaciones en las que un atacante con capacidades limitadas puede tomar el control funcional del dispositivo.

TABLE II
Campos extraídos del tráfico BLE y cobertura en el conjunto de datos.

Campo	Tipo	Descripción	Cobertura
`frame.number`	int	Número de paquete capturado	100 %
`frame.time_epoch`	float	Marca temporal Unix (s)	100 %
`frame.len`	int	Longitud de la trama (bytes)	100 %
`btle.length`	int	Longitud del *payload* BLE	100 %
`btle.central_bd_addr`	string	MAC del dispositivo central (maestro)	100 %
`btle.peripheral_bd_addr`	string	MAC del dispositivo periférico (esclavo)	100 %
`btle.access_address`	hex	*Access Address* del enlace BLE	100 %
`btle.data_header.llid`	int	Tipo de PDU (LLID)	100 %
`btatt.opcode`	int	*Opcode* ATT/GATT (-1 si no aplica)	100 %
`btatt.handle`	int	*Handle* GATT (-1 si no aplica)	100 %
`btatt.value`	hex	Valor ATT (datos escritos/leídos)	0,3 %
`inter_arrival_time`	float	Tiempo entre paquetes (s)	100 %
`type`	string	Clase: “normal” o “attack”	100 %

En este trabajo se ha generado un dataset de tráfico BLE–GATT etiquetado, obtenido a partir de capturas reales en las distintas fases del ataque y procesado mediante herramientas estándar como tshark y Python. Este dataset recoge tanto tráfico normal como tráfico bajo ataque, con un desbalance de clases representativo de escenarios reales, e incluye características temporales y de protocolo útiles para el análisis de patrones y la detección de anomalías. Su estructura y proceso de construcción se han documentado con el objetivo de facilitar la reproducibilidad y su reutilización en futuras evaluaciones de técnicas de monitorización y defensa en entornos IoT basados en BLE.

Como líneas de trabajo futuro, se plantea ampliar el catálogo de ataques considerados (por ejemplo, escenarios de *jamming* lógico a nivel GATT o ataques de *relay* y *MITM* en entornos controlados), así como generar nuevas capturas que permitan equilibrar mejor la proporción entre tráfico normal y malicioso y explorar configuraciones adicionales de servicios y características GATT. Estas extensiones contribuirán a disponer de datasets más variados y representativos, reforzando la evaluación de mecanismos de detección y endurecimiento de sistemas BLE en contextos IoT reales.

AGRADECIMIENTOS

Este proyecto ha sido financiado por el Plan de Recuperación, Transformación y Resiliencia de la Unión Europea (Next Generation) por medio del proyecto "Seguridad del Internet de las Cosas en Entornos Domésticos y Empresariales en el Contexto de la Tecnología 5G-IoT" del INCIBE.

REFERENCES

[1] Timur Yumalin AND R. B. Salikhov AND T. R. Salikhov. *Sustainable production: Leveraging energy-efficient data exchange protocols*. (2024). https://doi.org/10.1051/e3sconf/202454201005

[2] Minoru Fujisawa AND Hiroyuki Yasuda AND Ryosuke Isogai AND Maki Arai AND Y. Yoshida AND Aohan Li AND Song-Ju Kim AND Mikio Hasegawa. *An efficient beaconing of bluetooth low energy by decision making algorithm*. (2024). https://doi.org/10.1007/s44163-024-00122-7

[3] Jian Yang AND Christian Poellabauer AND Pramita Mitra AND Cynthia Neubecker. *Beyond beaconing: Emerging applications and challenges of BLE*. (2019). https://doi.org/10.1016/j.adhoc.2019.102015

[4] Candelaria Hernández-Goya AND Ricardo Aguasca-Colomo AND Cándido Caballero-Gil. *BLE-based secure tracking system proposal*. (2023). https://doi.org/10.1007/s11276-023-03347-z

[5] Xijia Che AND Yi He AND Xuewei Feng AND Kun Sun AND Ke Xu AND Qi Li. *BlueSWAT: A Lightweight State-Aware Security Framework for Bluetooth Low Energy*. (2024). http://arxiv.org/abs/2405.17987

[6] Yue Zhang AND Jian Weng AND Rajib Dey AND Xinwen Fu. *Bluetooth Low Energy (BLE) Security and Privacy*. (2019). https://doi.org/10.1007/978-3-319-32903-1_298-1

[7] Matthias Cäsar AND Tobias Pawelke AND Jan Steffan AND Gabriel Terhorst. *A survey on Bluetooth Low Energy security and privacy*. (2022). https://www.sciencedirect.com/science/article/pii/S1389128621005697

[8] Hadi Givehchian AND Nishant Bhaskar AND Eliana Rodriguez Herrera AND Héctor Rodrigo López Soto AND Christian Dameff AND Dinesh Bharadia AND Aaron Schulman. *Evaluating Physical-Layer BLE Location Tracking Attacks on Mobile Devices*. (2022). https://doi.org/10.1109/sp46214.2022.9833758

[9] Simon Liebl AND Leah Lathrop AND Ulrich Raithel AND Andreas Aßmuth AND Ian Ferguson AND Matthias Söllner. *Analyzing the Attack Surface and Threats of Industrial Internet of Things Devices*. (2024). https://arxiv.org/abs/2405.16318

[10] José Aveleira-Mata AND Diego Narciandi-Rodríguez AND Isaías García-Rodríguez AND Javier Alfonso-Cendón AND Sergio Rubio-Martín AND Héctor Aláiz-Moretón. *Auditoría funcional de juguetes IoT con Bluetooth LE*. (2025). https://doi.org/10.17979/ja-cea.2025.46.12272

[11] Gonzalo Munilla Garrido AND Johannes Sedlmeir AND Ömer Uludağ AND Ilias Soto Alaoui AND André Luckow AND Florian Matthes. *Revealing the landscape of privacy-enhancing technologies in the context of data markets for the IoT: A systematic literature review*. (2022). https://doi.org/10.1016/j.jnca.2022.103465

[12] Chandranshu Gupta AND Gaurav Varshney. *An improved authentication scheme for BLE devices with no I/O capabilities*. (2023). https://doi.org/10.1016/j.comcom.2023.01.001

[13] Candelaria Hernández-Goya AND Pino Caballero-Gil. *Analysis of Lightweight Cryptographic Solutions for Authentication in IoT*. (2013). https://doi.org/10.1007/978-3-642-53862-9_48

[14] Jianjun Ma AND Rabi Shrestha AND Jacob Adelberg AND Chia-Yi Yeh AND Zahed Hossain AND Edward W. Knightly AND Josep Miquel Jornet AND Daniel M. Mittleman. *Security and eavesdropping in terahertz wireless links*. (2018). https://doi.org/10.1038/s41586-018-0609-x

[15] Suranga Seneviratne AND Yining Hu AND Tham Nguyen AND Guohao Lan AND Sara Khalifa AND Kanchana Thilakarathna AND Mahbub Hassan AND Aruna Seneviratne. *A Survey of Wearable Devices and Challenges*. (2017). https://doi.org/10.1109/comst.2017.2731979

[16] Neil R. Cameron. *ESP32 Microcontroller*. (2023). https://doi.org/10.1007/978-1-4842-9376-8_1

[17] Christian Hirsch AND Luca Davoli AND Radu Grosu AND Gianluigi Ferrari. *DynGATT: A dynamic GATT-based data synchronization protocol for BLE networks*. (2023). https://doi.org/10.1016/j.comnet.2023.109560

[18] *INCIBE- Juguetes conectados*. (2024). https://www.incibe.es/sites/default/files/espacios/ed2026/laboratorio/EstudioINCIBE_ciberseguridad_juguetes_conectados.pdf

[19] Romain Cayre AND Florent Galtier AND Guillaume Auriol AND Vincent Nicomette AND Mohamed Kaâniche AND Géraldine Marconato. *InjectaBLE: Injecting malicious traffic into established Bluetooth Low Energy connections*. (2021). https://doi.org/10.1109/dsn48987.2021.00050

[20] Inigo Llaneza Aller AND José Manuel Redondo AND Luis Vinuesa. *Towards Lightweight Mobile Pentesting Tools to Quickly Assess Machine Security Levels*. (2019). https://doi.org/10.1109/tla.2019.8931199

[21] Marco Botta AND Davide Cavagnino. *A Framework for Reversible Data Embedding into Base45 and Other Non-Base64 Encoded Strings*. (2021). https://doi.org/10.3390/app12010241

Detección de Imágenes Generadas por Modelos de Difusión usando Información Espectral y Semántica mediante Mecanismos de Atención

Daniel Povedano Álvarez, Andrés Sánchez Casas, Alexandra Carolina Molina Ospina, Luis Javier García Villalba*

Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {dpovedano, asanch31, alexmoli, javierv}@ucm.es

***Resumen*—El realismo alcanzado por las actuales tecnologías de generación de imágenes sintéticas, impulsadas por modelos de difusión, plantea un desafío crítico para su detección. La creciente similitud visual entre imágenes reales y generadas conlleva riesgos sociales y de seguridad, incluyendo la suplantación de identidad y la difusión de *fake news* con fines maliciosos. Este trabajo presenta el diseño y la evaluación de un modelo basado en características espectrales y semánticas, aprendizaje profundo y mecanismos de atención para la verificación de autenticidad, capaz de identificar los rastros característicos que dejan los generadores. El enfoque propuesto diferencia de forma eficaz entre imágenes reales y sintéticas, alcanzando una precisión del 99 %. Finalmente, se destaca la necesidad de ampliar y diversificar el conjunto de datos con el objetivo de mejorar la robustez y la capacidad de generalización de los modelos, reforzando así su aplicabilidad en escenarios reales de seguridad.**

***Index Terms*—Aprendizaje Profundo, Modelos de Difusión, Detección de Imágenes Sintéticas, Desinformación.**

I. Introducción

El avance acelerado de la Inteligencia Artificial (IA), especialmente en el ámbito de los Modelos de Difusión o *Diffusion Models* (DM), ha marcado una nueva era en la síntesis de imágenes con un realismo sin precedentes. Estas tecnologías, si bien ofrecen un potencial creativo y científico significativo, han introducido simultáneamente serios desafíos en el espectro de la seguridad digital y la ética tecnológica.

La preocupación más urgente radica en el uso malintencionado de estas capacidades generativas para crear contenido falso o manipulado, incluyendo imágenes *deepfake* y material sintético destinado a fines delictivos graves, como la generación de imágenes de menores (Explotación Sexual Infantil (ESI)) y la desinformación masiva. Esta actividad no solo amenaza la integridad de las víctimas, sino que también establece complejos obstáculos legales, éticos y técnicos para las entidades encargadas de su detección y persecución. La urgencia de este problema se evidencia en incidentes recientes a nivel nacional, ilustrando la necesidad de abordar de forma proactiva la ciberdelincuencia facilitada por la IA [1]. [2]. Además, el Anteproyecto de Ley Orgánica Reguladora del Derecho de Rectificación, aprobado en diciembre de 2024, adapta el derecho a rectificación a la era digital, cubriendo plataformas y usuarios influyentes [3]. El objetivo principal de este trabajo es abordar la brecha tecnológica y regulatoria derivada del rápido avance de los modelos generativos, mediante el desarrollo de una herramienta robusta de verificación de autenticidad que permita a las fuerzas de seguridad y organismos competentes identificar con alta fiabilidad el origen sintético del contenido digital. El trabajo se estructura en cinco capítulos que guían al lector desde la fundamentación teórica hasta la validación experimental y las conclusiones. La propuesta se basa en (i) un filtrado espectral aprendible que sustituye la selección heurística de coeficientes DCT por filtros optimizables capaces de capturar artefactos de forma autónoma; (ii) una arquitectura híbrida con mecanismos de autoatención y atención cruzada para modelar dependencias globales entre características físicas (DCT/SRM) y semánticas (OpenCLIP); y (iii) la creación de un dataset propio y diverso de 40.000 imágenes, que incluye múltiples versiones de generadores (DALL·E 2/3, Midjourney 4/5/6 y Stable Diffusion 1.5/2.1), con el fin de mejorar la capacidad de generalización del sistema. A continuación, se presenta la estructura del trabajo:

- En la **Sección I** se presenta el contexto general del estudio y se sitúa al lector en el problema que motiva este trabajo. Además, se describen brevemente los objetivos y la relevancia del enfoque propuesto.
- En la **Sección II** presenta una revisión del estado del arte en la detección de imágenes sintéticas, destacando los enfoques y avances más recientes y relevantes en este campo.
- En la **Sección III** detalla la propuesta metodológica para alcanzar los objetivos del trabajo. Se describe la arquitectura del modelo diseñado, el conjunto de datos utilizado, las técnicas de preprocesamiento aplicadas y los criterios de evaluación definidos para la validación.
- En la **Sección IV** expone el desarrollo experimental completo del trabajo, presentando los resultados obtenidos en cada una de las pruebas y configuraciones realizadas.
- En la **Sección V** cierra el documento recopilando las conclusiones principales derivadas de la investigación, reflexionando sobre las limitaciones encontradas y proponiendo líneas de desarrollo futuro para el modelo.

II. Trabajos relacionados

Este capítulo revisa los métodos recientes para la detección de imágenes sintéticas. Se presentan diversas técnicas, desde el análisis de frecuencias hasta el uso de redes neuronales y herramientas de inversión, analizando sus ventajas y limitaciones. El objetivo es exponer las soluciones actuales y los desafíos pendientes en este campo.

II-A. Análisis de Frecuencias

Los autores de [4] exploraron el uso de transformadas de Fourier para encontrar rastros de modelos generativos. Si bien el rendimiento fue alto en imágenes sin comprimir, con 92.8 % AUC (Area Under the Curve), el rendimiento disminuyó drásticamente al simular condiciones de redes sociales (compresión y redimensionado), cayendo la precisión al 61.5 %. Esto evidencia una baja robustez ante el postprocesamiento común.

II-B. Detección y Localización Jerárquica

En [5] se propuso un sistema multirama que detecta y localiza manipulaciones (incluyendo *inpainting*) usando características de frecuencia. Logró altas métricas (96.1 % *F1-Score* en detección y 97.0 % en localización) en escenarios sintéticos conocidos. Sin embargo, su rendimiento se vio afectado al enfrentar modelos de manipulación no vistos durante el entrenamiento, atribuyéndose a la escasa diversidad de las imágenes reales usadas.

II-C. Detección Robusta

Epstein et al. [6] abordaron la detección en un escenario *online* con modelos en rápida evolución. Usando preprocesamiento realista (desenfoques, marcas de agua), lograron que los clasificadores generalizaran a modelos de arquitectura similar. La combinación de múltiples fuentes de datos (p.ej.,inpainting) mejoró notablemente la precisión (alcanzando un 98.91 %). No obstante, el método requiere reentrenamiento para adaptarse a nuevas arquitecturas y no cubre modelos cerrados.

II-D. Inversión de Imágenes

En *FakeInversion* [7] utilizaron imágenes invertidas de SD (Stable Diffusion) como datos de entrenamiento para identificar imágenes falsas de modelos no vistos. Este enfoque mostró una mejora relativa del 15-27 % en generalización. A pesar de ello, la diversidad de las imágenes reales sigue siendo un factor crucial, ya que el detector puede confundirse si los estilos o contenidos difieren mucho.

II-E. Características Físicas y Semánticas

Yan et al. [8], propusieron AIDE, un detector que incorporaba un módulo de puntuación *Discrete Cosine Transform* (DCT) para capturar las estadísticas de píxeles de bajo nivel, extrayendo parches de alta y baja frecuencia de la imagen; estos luego se procesaban a través de filtros SRM (una técnica originalmente introducida por Fridrich y Kodovsky [9] para caracterizar el patrón de ruido). Por otro lado, para la extracción de características semánticas, utilizaron OpenCLIP, un modelo de código abierto entrenado sobre múltiples datasets. El detector final combinó estadísticas físicas de bajo nivel (filtros DCT y *Spatial Rich Model* (SRM) con representaciones semánticas de alto nivel (Contrastive Language-Image Pre-Training). alcanzando un 86,88 % de media de precisión (*Avg. accuracy*) en el benchmark GenImage y un 92,77 % sobre el benchmark AIGC. Sus limitaciones incluyen una baja interpretabilidad (naturaleza çaja negra") y una reducción del rendimiento ante transformaciones extremas.

II-F. Métodos Híbridos y Comparativos (GANs y Difusión)

Trabajos como [10] y [11] han buscado detectores universales capaces de generalizar múltiples arquitecturas, incluyendo GANs y modelos de difusión, sin necesidad de entrenamiento exhaustivo en todos los generadores. FatFormer [12] combina atención y alineación multimodal para detectar imágenes de ambos paradigmas, mientras que estudios comparativos [13], han identificado características comunes y específicas que pueden ser explotadas para mejorar la detección transversal.

II-G. Resumen de Desafíos

Como se observa en la Tabla I, aunque los métodos han avanzado, la generalización sigue siendo el desafío principal. Los detectores actuales luchan por mantener el rendimiento frente a modelos generativos nuevos, arquitecturas distintas o postprocesamiento común como la compresión.

Tabla I
Resumen de trabajos sobre detección de imágenes generadas por IA

Trabajo	Método	Acc. (%)	Limitaciones
[5]	CNN	97,1	Difícil generalización.
[6]	Preprocesamiento realista y CutMix	98,91	Difícil generalización.
[7]	Inversión + protocolo *SynRIS*	72,23	Bajo rendimiento.
[8]	CLIP + preprocesamiento	86,88	Baja interpretabilidad

III. Detección de imágenes sintéticas

El objetivo principal de este trabajo es desarrollar un sistema automatizado para la detección de imágenes sintéticas (en concreto las generadas por modelos de difusión), utilizando las metodologías y avances más recientes en visión por computador y Aprendizaje Profundo (AP). A través de un análisis exhaustivo de las técnicas actuales, se buscó diseñar una solución eficiente capaz de identificar imágenes generadas por los modelos de difusión, una de las tecnologías más avanzadas en la generación de contenido visual realista. El proceso de desarrollo se estructura en las siguientes fases:

- **Selección de arquitectura base:** En primer lugar, se revisó el estado del arte para identificar las técnicas de detección más actuales. La propuesta consiste en utilizar una arquitectura existente como base o referencia [8], sobre la cual aplicar modificaciones con el objetivo de mejorar su rendimiento.
- **Desarrollo de un modelo propio:** A continuación, se desarrolló un modelo propio tomando como punto de partida la arquitectura base previamente seleccionada. A partir de ella, se añadieron ajustes y mejoras específicas con el fin de obtener los mejores resultados posibles.

- **Construcción del conjunto de datos:** Además, se creó un conjunto de datos propio compuesto por imágenes generadas mediante modelos de difusión, utilizando herramientas como DALL·E, SD y Midjourney.
- **Evaluación del modelo:** El objetivo principal es entrenar varias arquitecturas propias con el fin de explorar diferentes configuraciones y estrategias de mejora. A través de este enfoque comparativo, se busca identificar aquellas variantes que ofrezcan un mejor rendimiento y superar así las limitaciones de la arquitectura original.

III-A. Conjunto de Datos

El conjunto de datos principal está formado por 40.000 imágenes en total. Las imágenes reales fueron extraídas del conjunto de datos MSCOCO [14]. Para las imágenes sintéticas, se han recopilado ejemplos generados por diferentes herramientas con el objetivo de aumentar la diversidad del conjunto de datos y favorecer la capacidad de generalización del modelo. El subconjunto correspondiente a las imágenes sintéticas se resume en la Tabla II y está compuesto por:

- **Dall-E:** Se utilizaron 2000 imágenes generadas con la versión 2, extraídas del artículo [4], y 3310 imágenes generadas con la versión 3, obtenidas del *dataset* disponible en *Hugging Face* [15]. Las imágenes de la segunda versión tienen un tamaño uniforme de 1024x1024 píxeles, mientras que las correspondientes a la tercera versión presentan dimensiones variables.
- **Midjourney:** Se recopilaron 1000 imágenes correspondientes a la versión 4 y 2000 a la versión 5, provenientes del conjunto de datos [16]. Además, se añadieron 4000 imágenes generadas con la versión 6, obtenidas del *dataset* de *Hugging Face* [17]. Todas las imágenes de *Midjourney* tienen dimensiones 1024x1024 píxeles.
- **SD:** Se seleccionaron 3690 imágenes correspondientes a la versión 1.5, extraídas del *dataset* de *Hugging Face* [18], y 4000 imágenes generadas con la versión 2.1, tomadas del conjunto [19]. Las imágenes generadas por la primera versión tienen dimensiones 512x512 y las de la segunda versión 768x768.

Tabla II
DISTRIBUCIÓN DE IMÁGENES SINTÉTICAS POR SOFTWARE, VERSIÓN Y REFERENCIA.

Software	Versión	Nº de imágenes	Referencia
DALL·E	2	2000	[4]
DALL·E	3	3310	[15]
Midjourney	4	1000	[16]
Midjourney	5	2000	[16]
Midjourney	6	4000	[17]
SD	1.5	3690	[18]
SD	2.1	4000	[19]

III-B. Arquitectura

Inicialmente, se tomó como punto de partida la arquitectura propuesta en [8]. Esta arquitectura se basa en dividir la imagen en parches de tamaño uniforme, sobre los cuales se aplica la DCT. De todos los parches, se seleccionaron los dos con mayor y menor coeficientes respectivamente. A continuación, se aplicó un filtro SRM a estos parches seleccionados para extraer patrones de ruido, que posteriormente fueron procesados por una red *ResNet-50* para la extracción de características. Finalmente, las características extraídas de todos los parches se promediaron y se concatenaron con la salida de un modelo CLIP. A partir de esta base, se plantearon modificaciones con el objetivo de optimizar los resultados obtenidos. En contraste con el enfoque original (y otros trabajos), que aplican la DCT sobre parches y preseleccionan de manera heurística determinados coeficientes antes de la *Inverse Discrete Cosine Transform* (IDCT) y la aplicación del SRM, en este trabajo se empleó un proceso alternativo basado en filtrado aprendible. Concretamente, tras la DCT se aplican filtros parametrizados que se optimizan durante el entrenamiento, permitiendo que el modelo aprenda dinámicamente qué bandas de frecuencia son más discriminativas para la detección de *artifacts*. Posteriormente, cada parche filtrado se reconstruye mediante la IDCT para volver al dominio espacial, y todos los parches reconstruidos se combinan para formar la imagen completa, que constituye la entrada a los filtros SRM.

III-B1. Arquitectura Propia: Como se ha comentado anteriormente, para el desarrollo de la arquitectura propia se adoptó un enfoque distinto al del trabajo original, que seleccionaba parches heurísticos de altas y bajas frecuencias antes de aplicar la DCT, la IDCT y los filtros SRM. En nuestro caso, se eliminó dicha preselección manual, permitiendo que el modelo aprenda de forma autónoma qué componentes espectrales son relevantes.

La imagen se divide en parches no solapados de 32×32 píxeles, requisito necesario para garantizar su reconstrucción exacta tras la IDCT. Aunque el método original utilizaba solapamiento para aportar contexto entre parches, en nuestra arquitectura dicho contexto se modela posteriormente mediante mecanismos de autoatención, por lo que deja de ser necesario.

Cada parche se transformó mediante DCT y pasó por filtros aprendibles cuyos pesos se optimizaron durante el entrenamiento, evitando dependencias de particiones fijas del espectro. Tras aplicar la IDCT por parche, se reconstruyó la imagen completa y se procesaron los resultados con filtros SRM, que capturaron patrones de ruido y microtextura relevantes para la detección de manipulaciones. Estas características alimentaron una red CNN *ConvNeXtV2-Femto*, cuyos mapas de activación se refinaron mediante un módulo de autoatención encargado de modelar dependencias globales. Finalmente, se aplicó *sequence pooling* para condensar la información en un único vector para la clasificación.

Este *pipeline* permite procesar la imagen completa sin recurrir a heurísticas espectrales, capturando relaciones globales entre frecuencias y evitando sesgos inducidos por la preselección manual. En paralelo, se extrajeron *embeddings* semánticos utilizando un modelo OpenCLIP, incorporado como representación complementaria. Finalmente, se exploró el uso de un mecanismo de atención cruzada entre ambas representaciones (*cross-attention*), en lugar de realizar una simple concatenación. Los flujos descritos anteriormente se pueden observar en la Figura 1.

IV. EXPERIMENTOS

Los experimentos se llevaron a cabo utilizando diferentes arquitecturas e hiperparámetros con el objetivo de optimizar la clasificación entre imágenes reales y sintéticas. Para ello, se

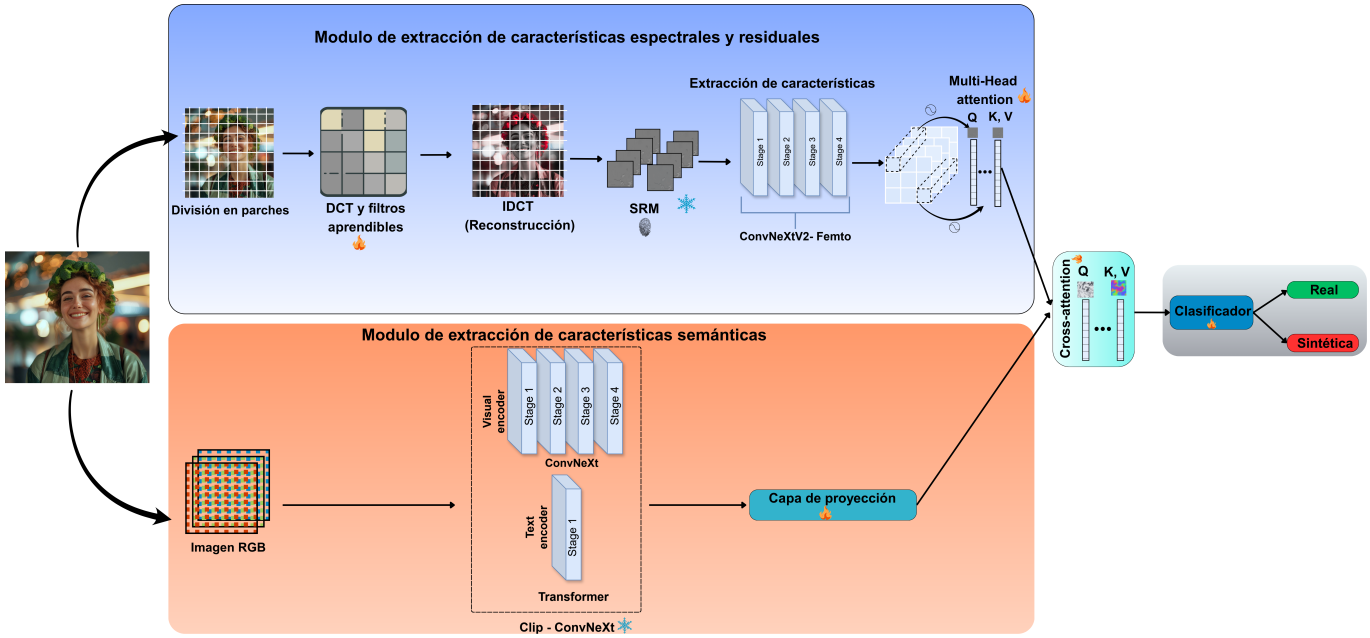


Figura 1. Arquitectura propuesta

estableció una validación cruzada con cinco *folds*, empleando el conjunto de entrenamiento (80 %). El conjunto de prueba (20 %) se reservó para el evaluación final, garantizando así una medida imparcial del desempeño general del modelo.

IV-A. Descripción de los Hiperparámetros

La Tabla III resume los valores utilizados para cada uno de estos hiperparámetros del entrenamiento. Dichos valores se establecieron tras revisar la literatura y realizar una experimentación previa. A continuación, se describen los hiperparámetros mas relevantes empleados en el entrenamiento:

- **Tasa de aprendizaje (*Learning rate*)**: Controla la amplitud de las actualizaciones de los pesos del modelo en cada paso del entrenamiento. Una tasa demasiado baja puede ralentizar el proceso de aprendizaje, mientras que una demasiado alta puede causar inestabilidad o impedir la convergencia del modelo.
- **Tamaño de lote (*Batch size*)**: Indica cuántas muestras se procesan antes de que el modelo actualice sus parámetros.
- **Decaimiento de peso (*Weight decay*)**: Técnica de regularización que ayuda a reducir el riesgo de sobreajuste y mejorar la generalización del modelo.
- **Número de épocas (*Num epochs*)**: Representa la cantidad de veces que el modelo recorre por completo el conjunto de datos de entrenamiento.
- **Parada temprana (*Early stopping*)**: Técnica de regularización que detiene el entrenamiento cuando el rendimiento de validación deja de mejorar.
- **Capas de atención (*Num layers*)**: Capas de autoatención que refinan las características extraídas por la CNN, capturando relaciones contextuales a largo alcance entre distintas regiones de la imagen.
- **Cabezas de atención (*Num heads*)**: Cada cabeza de atención permite al modelo enfocar su atención en diferentes áreas de los mapas de características o entre distintos parches de la imagen. Esto mejora la capacidad del modelo para aprender relaciones complejas, tanto locales como globales, facilitando la detección de patrones finos y estructurados en la imagen.

Tabla III
HIPERPARÁMETROS CLAVE

Nombre	Valor
Tamaño del Lote	8
Cabezas de atención	2
Capas de atención	4
Tasa de Aprendizaje	0,0001
Decaimiento de Peso	0,001
Número de Épocas	25
Parada temprana	3
Optimizador	AdamW
Aumento de datos	compresión, desenfoque, cambios de colores y rotaciones

IV-B. Configuración experimentos

La primera fase de los experimentos consistió en utilizar el conjunto de datos completo definido en la Tabla II, compuesto por 40.000 imágenes distribuidas equitativamente entre clases reales y sintéticas. El objetivo de esta fase era identificar cuál de las cuatro configuraciones de modelos presentaba un mejor rendimiento inicial, para así seleccionar las configuraciones más prometedoras que serían empleadas en las pruebas finales. Para ello, se aplicó una validación cruzada de *(5-fold)*, de manera que cada modelo fue evaluado de forma robusta sobre distintas particiones del conjunto de datos.

La Tabla IV presenta el rendimiento, expresado en términos de *F1-score*, alcanzado por cada configuración experimental durante la validación cruzada.

Como se aprecia en la Tabla IV, los resultados de la primera fase muestran diferencias claras entre las configuraciones evaluadas. El *Mean F1-score* más alto se obtuvo en el Experimento 2 (99,27 %) que incorpora auto-atención sin atención cruzada (empleando por tanto la concatenación

Tabla IV
RESULTADOS PRIMERA FASE DE EXPERIMENTOS.

Exp.	Mean F1-score (%)	Self-Att.	Cross-Att.	DCT	SRM
Base	92,91	–	–	Sí	Sí
1	97,90	No	No	Sí	Sí
2	**99,27**	**Sí**	**No**	**Sí**	**Sí**
3	98,98	No	Sí	Sí	Sí
4	98,87	Sí	Sí	Sí	Sí

Tabla V
REPORTE DE CLASIFICACIÓN PARA EL MODELO CON SELF-ATTENTION SOBRE EL CONJUNTO DE PRUEBA

Clase	Precision (%)	Sensibilidad (%)	F1-score (%)
Real (0)	97,2	99,7	98,4
Sintética (1)	99,6	97,2	98,4
Promedio (macro)	98,4	98,4	98,4
Precisión global		98,41 %	

entre ambas ramas), junto con DCT y SRM, indicando que la atención sobre características individuales de cada parche es especialmente eficaz. La configuración base [8], que carece de mecanismos de atención, obtuvo un desempeño notablemente menor (92,91 %) evidenciando la relevancia de los módulos de atención para mejorar la discriminación entre clases. Los experimentos 3 muestran que la atención cruzada también contribuye a la mejora, aunque en menor medida que solo la auto-atención, y que la combinación de ambos mecanismos (Experimento 4) no supera al Experimento 2 en esta fase inicial. Finalmente, para las pruebas con el conjunto de datos de test, se seleccionaron los Experimentos 2 y 3, ya que fueron los que lograron un mayor rendimiento en términos de *Mean F1-score*. El Experimento 4 se incluyó para evaluar su comportamiento con datos no vistos (test), con el fin de analizar su capacidad de generalización.

IV-C. Pruebas finales

Además de los resultados cuantitativos, fueron analizadas las matrices de confusión y los reportes de clasificación detallados correspondientes a los experimentos 2 y 3 (con mecanismo de auto-atención y atención cruzada respectivamente). En la Figura 2 y la Tabla V se presentan la matriz de confusión y el informe de clasificación , en el que se presentan las métricas de prueba (test) para el experimento 2.

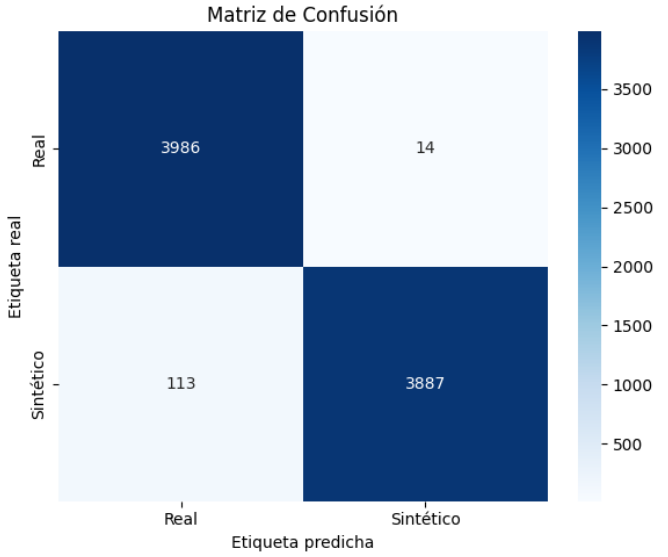


Figura 2. Matriz de confusión Experimento 2.

En el modelo con *Self-Attention*, se observó una ligera tendencia a clasificar imágenes sintéticas como reales, lo que se tradujo en un número más elevado de Falsos Negativos (FN). Tal comportamiento puede apreciarse claramente en la Figura 2, donde el bloque correspondiente a las imágenes sintéticas clasificadas erróneamente como reales muestra una concentración de errores significativamente mayor que su contraparte en el modelo con *Cross-Attention*. A pesar de ello, el sistema demostró una alta capacidad para reconocer contenido auténtico: el 99,6 % de las imágenes reales fueron correctamente identificadas (precisión). Como se recoge en la Tabla V, el *F1-score* macro promedio alcanzó un 98,4 %, reflejando un desempeño sólido y relativamente equilibrado entre ambas clases, aunque con una pequeña asimetría evidente en la detección de imágenes generadas. Por otro lado, los resultados obtenidos con el modelo basado en *Cross-Attention* evidencian una mejora sustancial tanto en precisión como en estabilidad de la clasificación. La Figura 3 muestra una reducción muy notable de FN en comparación con la matriz del modelo anterior, donde el bloque correspondiente a errores es considerablemente más pequeño. En este caso, el porcentaje de acierto en la clase real asciende al 99,57 %, según se indica en la Tabla VI. Además, el *F1-score* macro obtenido (99,3 %) es superior al del modelo con *Self-Attention*, lo que pone de manifiesto una mejora global y más consistente entre ambas clases. Estos resultados sugieren que la interacción explícita entre características, facilitada por el *Cross-Attention*, permite capturar dependencias más finas y relaciones discriminativas que no son explotadas por mecanismos de auto-atención, ya que aquí intervienen dos fuentes de datos diferentes.

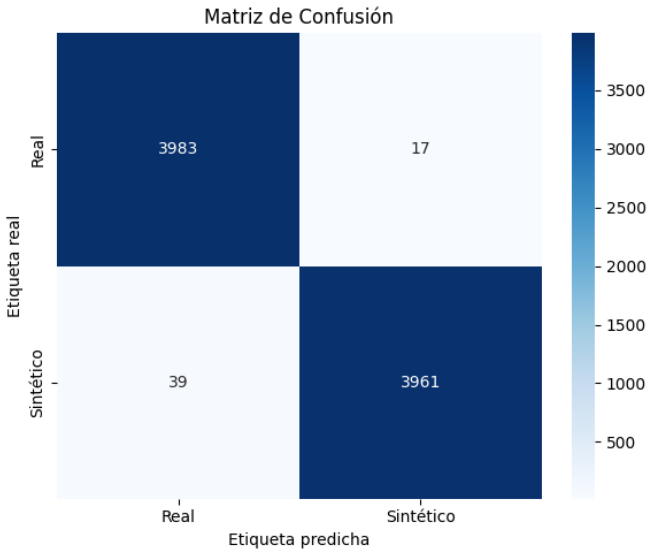


Figura 3. Matriz de confusión Experimento 3.

V. CONCLUSIONES Y TRABAJO FUTURO

Este estudio presenta un modelo robusto para la detección de imágenes sintéticas mediante la integración de CNN,

Tabla VI
REPORTE DE CLASIFICACIÓN PARA EL MODELO CON CROSS-ATTENTION SOBRE EL CONJUNTO DE PRUEBA

Clase	Precision (%)	Sensibilidad (%)	F1-score (%)
Real (0)	99	99,6	99,3
Sintética (1)	99,6	99,0	99,3
Promedio (macro)	99,3	99,3	99,3
Precisión global		99,3 %	

transformaciones DCT/SRM y mecanismos de atención. Los resultados experimentales muestran que ambas variantes propuestas *Self-Attention* y *Cross-Attention* contribuyen a mejorar la capacidad discriminativa del sistema, ya que permiten capturar información global, dependencias espaciales y patrones de alta frecuencia característicos de las imágenes generadas por modelos de difusión. Aunque el modelo correspondiente al Experimento 2 obtuvo las mejores métricas en validación cruzada, en el conjunto de prueba final el enfoque basado en *Cross-Attention* redujo de forma significativa los falsos negativos. Esto sugiere que la interacción explícita entre representaciones, facilitada por la atención cruzada, ofrece un beneficio adicional respecto a la simple concatenación de características. No obstante, el modelo con *Self-Attention* alcanzó el mayor *F1-score* promedio (99,83 %) y una mayor consistencia entre los distintos *folds*, consolidándose como la alternativa más estable y fiable desde el punto de vista de la generalización. El análisis de las matrices de confusión muestra que el modelo con *Self-Attention* tiende a confundir imágenes sintéticas con reales, mientras que la *Cross-Attention* mejora la separación entre clases y reduce las predicciones ambiguas, evidenciando su utilidad en escenarios con fronteras difusas entre contenido real y sintético.

De cara al trabajo futuro, se proponen varias líneas de acción para ampliar el alcance de la arquitectura presentada. En primer lugar, se plantea aumentar la diversidad y el tamaño del conjunto de datos, incorporando bases de referencia del estado del arte para su evaluación en escenarios más exigentes. En segundo lugar, se explorará el uso de *Vision-Language Models* como alternativa o complemento a la arquitectura híbrida actual, con el fin de capturar representaciones multimodales más ricas. Finalmente, se considerará la integración de métodos de interpretación del modelo, como mapas de activación, para analizar los factores espaciales y de frecuencia que influyen en la decisión final.

AGRADECIMIENTOS

Este trabajo se ha realizado con los fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra "Ciberseguridad para la Innovación y la Protección Digital" INCIBE-UCM. Además, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, proyecto CIRMA-CM (TEC-2024/COM-404).

El contenido de este artículo no refleja la opinión oficial de la Unión Europea. Los autores son los únicos responsables de la información y las opiniones expresadas en él.

REFERENCIAS

[1] Redacción HuffPost, "Condenado un hombre que compartía en redes pornografía infantil generada con ia y fotos de allegados," 2025, https://www.huffingtonpost.es/sociedad/condenado-hombre-compartia-redes-pornografia-infantil-generada-ia-fotos-allegados.html. Accedido el 9 de mayo de 2025.

[2] A. Montoro and I. Modrón, "El gobierno aprueba la ley que castiga el contenido sexual de menores creado con ia y el 'grooming'," 2025, accedido el 9 de mayo de 2025. [Online]. Available: https://www.rtve.es/noticias/20250325/ley-proteccion-menores-entornos-digitales-reformas-legales/16506012.shtml

[3] G. Lerma, "El gobierno obligará a los usuarios de redes sociales a rectificar cuando publiquen noticias falsas o inexactas," 2024, https://cadenaser.com/nacional/2024/12/17/el-gobierno-obligara-a-los-usuarios-de-redes-sociales-a-rectificar-cuando-publiquen-noticias-falsas-o-inexactas-cadena-ser/.

[4] R. Corvi, D. Cozzolino, G. Zingarini, G. Poggi, K. Nagano, and L. Verdoliva, "On the detection of synthetic images generated by diffusion models," 2022. [Online]. Available: http://arxiv.org/abs/2211.00680

[5] X. Guo, X. Liu, Z. Ren, S. Grosz, I. Masi, and X. Liu, "Hierarchical fine-grained image forgery detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2023/papers/Guo_Hierarchical_Fine-Grained_Image_Forgery_Detection_and_Localization_CVPR_2023_paper.pdf

[6] D. C. Epstein, I. Jain, O. Wang, and R. Zhang, "Online detection of ai-generated images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2023. [Online]. Available: https://openaccess.thecvf.com/content/ICCV2023W/DFAD/papers/Epstein_Online_Detection_of_AI-Generated_Images_ICCVW_2023_paper.pdf

[7] G. Cazenavette, A. Sud, T. Leung, and B. Usman, "Fakeinversion: learning to detect images from unseen text-to-image models by inverting stable diffusion," in *arXiv preprint arXiv:2406.08603*, 2024. [Online]. Available: https://arxiv.org/abs/2406.08603

[8] S. Yan, O. Li, J. Cai, Y. Hao, X. Jiang, Y. Hu, and W. Xie, "A sanity check for ai-generated image detection," in *International Conference on Representation Learning*, Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu, Eds., vol. 2025, 2025, pp. 70 702–70 720. [Online]. Available: https://proceedings.iclr.cc/paper_files/paper/2025/file/b0303773962ea1b5394c3a83cc7dd066-Paper-Conference.pdf

[9] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[10] U. Ojha, Y. Li, and Y. J. Lee, "Towards universal fake image detectors that generalize across generative models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 24 480–24 489.

[11] Z. Liu, H. Wang, Y. Kang, and S. Wang, "Mixture of low-rank experts for transferable ai-generated image detection," *arXiv preprint arXiv:2404.04883*, 2024.

[12] H. Liu, Z. Tan, C. Tan, Y. Wei, J. Wang, and Y. Zhao, "Forgery-aware adaptive transformer for generalizable synthetic image detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 10 770–10 780.

[13] A. Q. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," in *International conference on machine learning*. PMLR, 2021, pp. 8162–8171.

[14] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: common objects in context," in *European Conference on Computer Vision (ECCV) 2014*, 2014, pp. 740–755. [Online]. Available: https://arxiv.org/abs/1405.0312

[15] Ehristoforu, "Dall·e 3 images dataset," 2024, accessed: 2025-04-27. [Online]. Available: https://huggingface.co/datasets/ehristoforu/dalle-3-images

[16] J. Ricker, D. Lukovnikov, and A. Fischer, "Aeroblade: Training-free detection of latent diffusion images using autoencoder reconstruction error (dataset)," 2024, accessed: 2025-04-27. [Online]. Available: https://doi.org/10.5281/zenodo.10997235

[17] I. Drokin, "Midjourney v6 llava dataset," 2024, accessed: 2025-04-27. [Online]. Available: https://huggingface.co/datasets/brivangl/midjourney-v6-llava

[18] Yuwan0, "Lexica stable diffusion v1.5 dataset," 2023, accessed: 2025-04-27. [Online]. Available: https://huggingface.co/datasets/yuwan0/lexica-stable-diffusion-v1-5

[19] gryffindor-ISWS, "Stable diffusion 2.1 with images dataset," 2023, accessed: 2025-04-27. [Online]. Available: https://huggingface.co/datasets/gryffindor-ISWS/stable-diffusion-2-1-with-images

Un Marco Multicapa de Ciberengaño con IA y Orquestación Zero-touch Multidominio

Pedro Beltrán-López, Pantaleone Nespoli, and Manuel Gil Pérez
CyberDataLab, Universidad de Murcia, 30100 Murcia, España
Email: {pedro.beltranl, pantaleone.nespoli, mgilperez}@um.es

***Resumen*—Este es un resumen extendido de [1]. Las limitaciones de los enfoques tradicionales ante ciberamenazas avanzadas hacen necesario un marco más activo y adaptable. En respuesta, se presenta un marco unificado de ciberengaño (CYDEC) que integra modelos de engaño, monitorización, motores de decisión y orquestación automatizada para cubrir coherentemente prevención, detección, reacción y análisis forense. El diseño modular opera con flujos híbridos en y fuera de tiempo real, incorporando IA para defensas adaptativas (p. ej., rotación de señuelos, *moving target defense* y aislamiento bajo demanda). La telemetría obtenida durante el *engagement* retroalimenta el sistema para mejorar precisión y resiliencia.**

***Index Terms*—Ciberengaño, defensas adaptativas, modelos de engaño, orquestación automatizada**

I. Introducción

En los últimos años, la creciente sofisticación y persistencia de las ciberamenazas ha superado las capacidades de los mecanismos de seguridad tradicionales, exigiendo arquitecturas no solo resilientes, sino también adaptativas e inteligentes [2]. Aunque las tecnologías basadas en ciberengaño han ganado relevancia como medio para distraer a los atacantes y recopilar evidencias, la mayoría de los enfoques actuales permanecen fragmentados, centrándose en capas específicas del sistema o en tácticas aisladas como la detección o la prevención. Esta fragmentación limita su aplicabilidad en escenarios complejos como redes 5G, infraestructuras críticas (ICS) o entornos militares, donde los adversarios explotan múltiples vectores de ataque de forma simultánea.

Además, muchas soluciones carecen de mecanismos efectivos para integrar fuentes de datos heterogéneas, automatizar la toma de decisiones o adaptarse dinámicamente en tiempo real. Como resultado, las defensas tienden a volverse estáticas y predecibles frente a adversarios que emplean automatización, inteligencia artificial y técnicas avanzadas de evasión. En este contexto, resulta necesario un marco unificado que incorpore el ciberengaño de forma transversal y aproveche la IA para reforzar de manera continua los ciclos de defensa.

El marco presentado en este trabajo sigue esta visión [1]. Basado en los principios del Ciberengaño (CYDEC), integra modelos de engaño, mecanismos de monitorización, motores de decisión y componentes de orquestación en una arquitectura cohesiva. A diferencia de aproximaciones previas, abarca de forma automatizada las cuatro tácticas defensivas (prevención, detección, reacción y análisis forense) y se adapta a distintos contextos operativos mediante un diseño modular y orquestación Zero-touch.

En concreto, este artículo contribuye con las siguientes aportaciones:

- Una arquitectura multicapa basada en CYDEC que integra módulos de datos, defensa, decisión y orquestación, aplicable a entornos como 5G, ICS y dominios militares.
- La incorporación de modelos de engaño impulsados por IA capaces de operar en múltiples dimensiones del sistema, habilitando defensas adaptativas y mejora continua.
- Un flujo operativo híbrido que combina reacciones automáticas en tiempo real con procesos periódicos de aprendizaje y optimización.

El resto del artículo se organiza de la siguiente manera. La Sección II revisa el estado del arte, la Sección III describe el marco propuesto y un caso de uso, la Sección IV discute los principales hallazgos y la Sección V presenta las conclusiones.

II. Estado del Arte en Frameworks de Ciberengaño

La literatura ha propuesto frameworks orientados a distintos dominios con diferencias en automatización, escalabilidad y aplicabilidad.

Entre los enfoques iniciales destaca *D-Net* [3], que redirige tráfico desde una red comprometida hacia una red señuelo mediante SDN, permitiendo observación y contención sin interrupción. En paralelo, el *Active Deception Framework* (ADF) [4] ofrece un entorno extensible para desarrollar técnicas de engaño, con monitorización y orquestación apoyadas en SDN.

En IoT, algunos trabajos emplean honeynets o autenticación para redirigir tráfico sospechoso y estudiar al atacante [5], mientras que otros mitigan DDoS mediante reconfiguración dinámica de red [6]. Propuestas más recientes incluyen *DOLOS* [7], que combina MTD y engaño, y *ACDF* [8], que integra honeypots con aprendizaje profundo para generar artefactos más realistas.

MITRE Engage [9] estructura la interacción con el adversario y su alineación con ATT&CK. No obstante, muchos marcos se centran en dimensiones o tácticas concretas (p. ej., red o detección) y no cubren de forma integral prevención, detección, reacción y forense, ni todas las dimensiones (datos, red, sistema y software) con automatización suficiente.

En conjunto, persiste la ausencia de un marco modular que unifique dominios y aproveche IA para optimizar el ciclo defensivo; estas carencias motivan la propuesta presentada.

III. Marco de Ciberengaño Propuesto

Esta sección presenta el marco propuesto basado en CYDEC. La Figura 1 muestra una vista abstracta del marco de diseño resultante, que consiste en varias capas capaces de implementar y mantener mecanismos de seguridad basados en CYDEC. Este marco aprovecha diversas estrategias y mecanismos CYDEC en entornos genéricos, lo que lo hace versátil

para aplicaciones en redes 5G, ICS y entornos militares, entre otros.

Cabe señalar que el marco propuesto, a diferencia de otros, utiliza mecanismos de la IA para mejorar la defensa en las cuatro tácticas de defensa (prevención, detección, reacción y forense).

III-A. Capas del marco de ciberengaño

En la Figura 1, los nombres de las capas que componen el marco están representados como cuadros azules, mientras que en color gris cálido se representan internamente los componentes que contiene cada capa.

DATOS: Esta capa gestiona la adquisición, preprocesamiento y persistencia de información crítica para la defensa cibernética. Incluye mecanismos de ingesta continua de eventos desde monitores de red, host y engaño, aplicando técnicas de *stream processing* y normalización semántica. Los datos se etiquetan mediante esquemas inspirados en el modelo MITRE ATT&CK y se almacenan en bases distribuidas seguras optimizadas para consultas en tiempo real.

- Cola: Una estructura de datos usada para almacenar y procesar eventos de seguridad de manera ordenada y eficiente.
- Preprocesamiento: Este componente consiste en el procesamiento inicial de los datos antes del análisis para mejorar su calidad o extraer características relevantes para la seguridad.
- Fuentes Externas: Los datos externos siempre son necesarios en cualquier marco de defensa. La información recolectada se unirá con información externa de otras fuentes de datos para crear modelos robustos y eficientes.
- Etiquetado: Consiste en asignar etiquetas a los datos para identificar su importancia, sensibilidad u otras características relevantes para la seguridad del sistema.
- Base de Datos: Un almacén de datos usado para diversas funciones de seguridad como registro de eventos, gestión de identidad y accesos, etc.
- Modelo de Engaño: Este componente diseña e implementa modelos de IA que soportan técnicas de engaño en las dimensiones de datos, sistema, aplicación y red [1]. Para dotar al sistema de capacidad autónoma, formalizamos la decisión de engaño como un Proceso de Decisión de Markov Parcialmente Observable (*POMDP*) definido por $(\mathcal{S}, \mathcal{A}, T, R, \gamma)$. El objetivo del agente de IA es maximizar la función de recompensa acumulada esperada $J(\theta)$, que equilibra la obtención de inteligencia frente al riesgo operativo. La función objetivo se define en la Ecuación (1):

$$J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \big(R(s_t, a_t) - \lambda \cdot \text{Risk}(s_t, a_t)\big)\right] + \beta \mathcal{H}(\pi_\theta) \tag{1}$$

where $\pi_\theta(a|s)$ corresponds to the stochastic policy parametrized by a deep neural network, determining the probability of deception actions; $R(s_t, a_t)$ represents the base reward derived from engagement success and intelligence gain; $\text{Risk}(s_t, a_t)$ quantifies the penalty associated with operational impact or detection probability; λ is a weighting coefficient for risk sensitivity (crucial for critical environments); γ is the discount factor; and $\beta\mathcal{H}(\pi_\theta)$ is the entropy regularization term to encourage exploration.

El agente aprende estas políticas mediante algoritmos de *Deep Reinforcement Learning* (como PPO para acciones continuas o DQN para discretas). En paralelo, el proceso se modela como un *juego de Stackelberg* defensor–atacante, resolviendo por *best-response* para seleccionar tácticas que maximizan la utilidad esperada bajo restricciones operativas.

Las dimensiones de aplicación de este modelo son las siguientes:

- *Datos:* Modelos que aprenden patrones de acceso o manipulación de datos, simulando artefactos falsos (*decoy data*) o generando datos sintéticos creíbles mediante modelos generativos (e.g., VAEs, GANs).
- *Sistema:* Agentes que simulan comportamientos del sistema operativo ante acciones sospechosas, adaptando las señales según la política de RL aprendida.
- *Aplicación:* Modelos que replican la lógica o interacción de aplicaciones reales para atraer al atacante; pueden usar NLP o LLMs para generar respuestas contextualmente coherentes.
- *Red:* Modelos que simulan tráfico, latencia o patrones de comunicación a nivel de red (usando RL o aprendizaje adversarial) con el objetivo de confundir o ralentizar la exploración del atacante.

MONITORES: Supervisa continuamente la salud y seguridad del sistema. Obtiene información en tiempo real del sistema y las redes, y la transfiere a la capa de datos para un preprocesamiento y análisis, con el fin de detectar actividad sospechosa en la organización.

- Monitor de Engaño: Supervisa las técnicas de engaño implementadas para advertir de una amenaza.
- Monitor de Host: Supervisa los hosts en busca de posibles amenazas.
- Monitor de Red: Supervisa el estado de la red para detectar comportamientos sospechosos y advertir de posibles amenazas.

DEFENSA: La capa de defensa realiza la detección de amenazas y alerta a la capa de Decisión para reaccionar ante ellas.

- Motor de Detección: Componente responsable de detectar posibles amenazas o intrusiones en el sistema. Utiliza técnicas de análisis de seguridad para identificar patrones maliciosos o sospechosos y generar alertas para un análisis posterior. La información analizada será proporcionada por la capa de datos, que la habrá preprocesado para un análisis más eficiente.
  - Conocimiento: Base de conocimiento que utiliza el motor de detección para identificar patrones de actividad maliciosa o sospechosa.
  - Comportamiento: Análisis de comportamiento del sistema y de los usuarios para detectar anomalías que indiquen un posible ataque en curso.

DECISIÓN: La capa de decisión es responsable de analizar y evaluar los eventos de seguridad para determinar las acciones adecuadas, es decir, protege directamente el sistema. Estas funciones incluyen identificar y priorizar las amenazas,

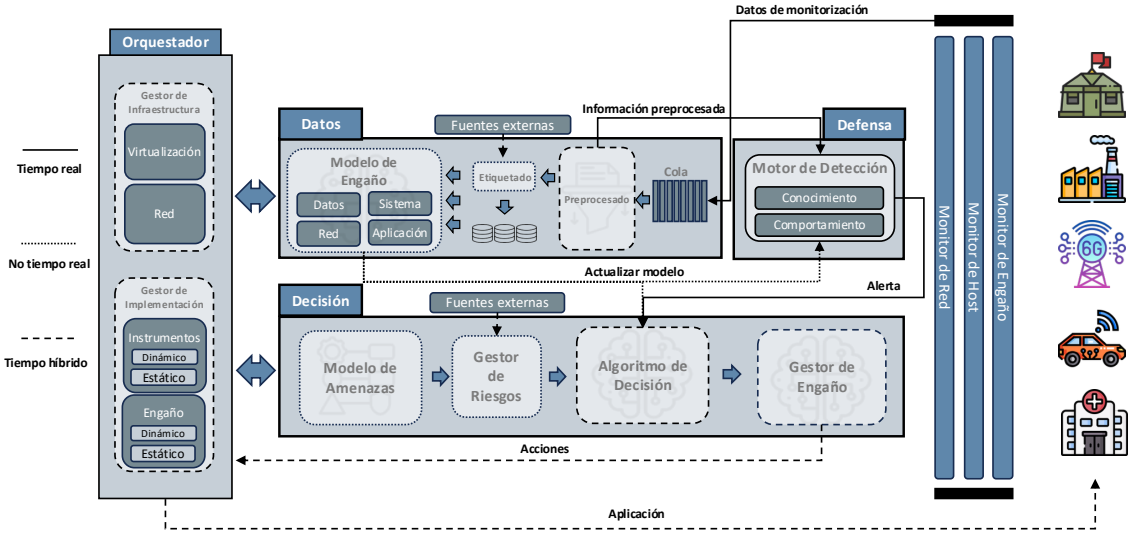


Figura 1. Marco propuesto para realizar tareas de prevención, detección, respuesta y análisis forense usando CYDEC.

evaluar los riesgos y tomar las decisiones automáticas o recomendaciones para mitigar los riesgos identificados. Las decisiones estarán basadas en CYDEC, seleccionando qué técnicas usar en un activo en un momento dado.

- Modelo de Amenazas: Describe las posibles amenazas y vulnerabilidades que podrían afectar al sistema.
- Fuentes Externas: Permiten obtener una correcta evaluación de los riesgos frente a las amenazas que podrían atacar al sistema.
- Gestor de Riesgos: Evalúa el nivel de riesgo asociado a las amenazas identificadas y ayuda a priorizar medidas de seguridad.
- Algoritmo de Decisión: Algoritmo usado para tomar decisiones automáticas ante eventos de seguridad.
- Gestor de Engaño: El Gestor de Engaño de CYDEC diseña y gobierna las tácticas que introducen elementos deliberadamente falsos o controlados en el entorno para confundir y canalizar a posibles atacantes. Apoyado en los modelos de engaño, toma decisiones sobre cuándo y cómo comprometer (engage) al adversario para maximizar la obtención de información dinámica sobre su comportamiento, que luego se analiza para refinar las defensas. Tras definir la táctica y sus parámetros, transmite las especificaciones al Orquestador para su despliegue y supervisión, mientras incorpora la telemetría recibida para ajustar continuamente las estrategias sin afectar a los servicios legítimos.

ORQUESTACIÓN: Esta capa coordina y gestiona operaciones y recursos de la infraestructura extremo a extremo. Automatiza el ciclo de vida de funciones de red y herramientas de seguridad, exponiendo APIs de control y telemetría y usando mensajería asíncrona para desacoplar componentes y reducir latencia. Las decisiones emitidas por los agentes de defensa activa se materializan aquí como cambios concretos sobre la infraestructura y la red.

- Gestor de Infraestructura: Administra los recursos de infraestructura necesarios para el sistema.
  - NFV / Virtualización: Responsable de gestionar el ciclo de vida de VNFs/contenedores que implementan mecanismos de defensa o componentes del sistema (instanciación, *placement*, escalado horizontal/vertical, migración y retiro), aplicando decisiones de los agentes a partir de telemetría en tiempo real y respetando cuotas y afinidades de carga.
  - Red: Configuración y aseguramiento del correcto funcionamiento de las redes mediante un plano de control programable, garantizando coherencia con las políticas y sin degradar el tráfico legítimo.
- Gestor de Despliegue: Coordina el despliegue de herramientas y técnicas de defensa.
  - Instrumentos: Despliega artefactos dinámicos o estáticos, pruebas automáticas de regresión y rollback seguro. Integra telemetría continua para cerrar el bucle de control y ajustar parámetros.
  - Engaño: Despliega y parametriza estrategias de engaño en función de las decisiones y umbrales de riesgo, validando coherencia semántica y desmontando activos tras la captura de evidencia para minimizar la huella y mantener la credibilidad del entorno.

III-B. Flujos de trabajo del marco de ciberengaño

Se han identificado diferentes flujos según si la información que circula por el marco propuesto es en tiempo real o no. Esto permite defenderse de manera reactiva y proactiva frente a ciberamenazas.

- Tiempo real: Procesa los datos en tiempo real sin intervención manual (Zero-touch). El flujo involucra monitores, cola, motor de detección, modelo de decisión, gestor de engaño y orquestador ejecutando acciones inmediatas.
- No en tiempo real: Preprocesa, etiqueta y almacena datos. Posteriormente, se entrenan modelos de IA periódicamente para mejorar la detección, decisión y engaño. Estos modelos permiten medidas preventivas y forenses.
- Híbrido: Opera tanto en tiempo real como fuera de él según las necesidades. Las acciones del Gestor de Engaño y la ejecución del Orquestador pueden ser reactivas

o preventivas.

III-C. Caso de Estudio y Validación del Flujo

Para proporcionar una validación tangible del marco propuesto y aterrizar los conceptos teóricos, describimos a continuación un escenario operativo de *Defensa de Superficie Móvil* ante un ataque de movimiento lateral en un entorno virtualizado 5G. Este boceto ilustra la interacción secuencial de las capas:

1. **Detección Inicial (Monitorización):** El *Monitor de Red* identifica un escaneo de puertos anómalo originado desde una VNF comprometida en el borde de la red. El evento se envía inmediatamente a la *Cola* de la capa de Datos.
2. **Preprocesamiento y Evaluación (Datos):** La capa de Datos normaliza el evento y lo enriquece con inteligencia de amenazas. El *Modelo de Engaño* calcula la puntuación de riesgo actual.
3. **Toma de Decisión (Decisión):** En lugar de bloquear la IP (acción estática), el agente RL utiliza la Ecuación 1. Evalúa que desplegar un señuelo ofrece una recompensa esperada (R) superior al riesgo ($\lambda \cdot \text{Risk}$), dado que permite estudiar las herramientas del atacante. El agente selecciona la acción a_t: "Desplegar servicio Access and Mobility Management Function (AMF) simulado".
4. **Ejecución Zero-Touch (Orquestación):** El *Gestor de Infraestructura* recibe la orden. A través de APIs de Kubernetes, instancia un contenedor ligero que simula la interfaz del AMF. Simultáneamente, reprograma el plano de control SDN para redirigir el tráfico del atacante hacia este nuevo contenedor de forma transparente.
5. **Recolección de Evidencia (Forense):** El atacante interactúa con el señuelo. Toda la telemetría generada se almacena y se utiliza para reentrenar los modelos de detección en el flujo "No en tiempo real", cerrando el ciclo de defensa adaptativa.

Este flujo demuestra cómo la arquitectura permite pasar de una alerta pasiva a una contención activa e inteligente sin intervención humana, validando la coherencia del diseño propuesto.

IV. Discusión

El marco propuesto basado en CYDEC abre una perspectiva renovada sobre el papel del ciberengaño en arquitecturas de defensa modernas. A diferencia de los enfoques tradicionales, que tienden a centrarse en funciones puntuales como la detección o la contención, la propuesta presentada articula de manera integral las cuatro tácticas defensivas en un mismo entorno automatizado. Este enfoque no solo amplía la cobertura frente a ataques multivectoriales, sino que también refuerza la capacidad de aprendizaje continuo a través de la incorporación de modelos de IA.

Uno de los aspectos más relevantes es la capacidad del sistema para operar en flujos híbridos, equilibrando la inmediatez de las respuestas en tiempo real con el reentrenamiento periódico de los modelos de IA. Esta característica le confiere un carácter adaptativo que resulta especialmente valioso frente a adversarios que evolucionan con rapidez. Además, el uso de una arquitectura modular y la integración de orquestación Zero-touch evidencian un avance hacia defensas más autónomas, reduciendo la dependencia de la intervención humana y aumentando la eficiencia en escenarios de gran escala.

Por otro lado, la discusión debe considerar el impacto que un marco de estas características puede tener en la práctica. Su aplicabilidad en entornos como redes 5G, infraestructuras críticas y dominios militares plantea retos de interoperabilidad, pero también una oportunidad para estandarizar el uso de mecanismos de ciberengaño como parte fundamental de la ciberdefensa. En este sentido, este trabajo contribuye a sentar las bases para un ecosistema más robusto, en el que cada interacción hostil puede transformarse en conocimiento accionable y en un refuerzo sistemático de la seguridad.

V. Conclusiones

Este trabajo ha presentado un marco unificado de Ciberengaño (CYDEC) que integra coherentemente prevención, detección, reacción y análisis forense en una única arquitectura. La propuesta combina la versatilidad del ciberengaño con la potencia de la IA, generando defensas dinámicas capaces de adaptarse en tiempo real a la evolución de los ataques. Asimismo, utiliza modelos de engaño fundamentados matemáticamente en procesos POMDP que guían la toma de decisiones para realizar *engagement* con el adversario y obtener información dinámica sobre su comportamiento. Su diseño modular y la orquestación Zero-touch evidencian su aplicabilidad en entornos críticos, donde la automatización y la adaptabilidad son esenciales. Así, cada interacción maliciosa se convierte no solo en una amenaza, sino también en una oportunidad de optimización y aprendizaje. Como trabajo futuro, se plantea profundizar en la validación práctica del marco en escenarios operativos reales ante ataques persistentes y multivectoriales, y enriquecer los modelos de engaño con técnicas avanzadas de IA.

Referencias

[1] P. Beltrán-López, M. Gil Pérez, and P. Nespoli, "Cyber deception: Taxonomy, state of the art, frameworks, trends, and open challenges," *IEEE Communications Surveys & Tutorials*, Early access, 2025.

[2] L. Zhang and V. L. L. Thing, "Three decades of deception techniques in active cyber defense—Retrospect and outlook," *Computers & Security*, vol. 106, art. no. 102288, 2021.

[3] T. Shimanaka *et al.*, "Cyber deception architecture: Covert attack reconnaissance using a safe SDN approach," in *Proc. 52nd Hawaii Int. Conf. Syst. Sci.*, 2019.

[4] M. M. Islam and E. Al-Shaer, "Active deception framework: An extensible development environment for adaptive cyber deception," in *IEEE Secure Dev.*, 2020, pp. 41–48.

[5] J. Pawlick *et al.*, "A game-theoretic taxonomy and survey of defensive deception for cybersecurity and privacy," *ACM Comput. Surv.*, vol. 52, no. 4, pp. 1–28, 2019.

[6] Y. Zhou *et al.*, "An SDN-enabled proactive defense framework for DDoS mitigation in IoT networks," in *IEEE Trans. Inf. Forensic Secur*, 16, 5366-5380, 2021.

[7] M. Almeshekah *et al.*, "DOLOS: A novel architecture for moving target defense," in *IEEE Trans. Inf. Forensic Secur*, 2023.

[8] Q. Gu *et al.*, "Advanced cyber deception framework (acdf): A comprehensive study," in *2024 Int. Conf. Comput., Netw. & Commun. (ICNC)*IEEE, 2024, pp. 203–208.

[9] MITRE, "MITRE Engage: A framework for adversary engagement," Technical Report, 2021.

Microagregación híbrida asistida por modelos de lenguaje

Sara Lanuza-Orna
Departament d'Enginyeria Informàtica i Matemàtiques
Universitat Rovira i Virgili, Tarragona, España
sara.lanuza@urv.cat

Alberto Blanco-Justicia
Departament d'Enginyeria Informàtica i Matemàtiques
Universitat Rovira i Virgili, Tarragona, España
alberto.blanco@urv.cat

***Resumen*—La microagregación es una técnica clásica del control estadístico de la divulgación (SDC) utilizada para garantizar el k-anonimato en microdatos. Sin embargo, los métodos tradicionales se centran en atributos numéricos y no capturan la semántica de los atributos textuales, reduciendo la utilidad de los datos anonimizados. Este trabajo presenta una microagregación híbrida que combina el algoritmo *Maximum Distance to Average Vector* (MDAV) con modelos de lenguaje preentrenados. En concreto, se emplea *Sentence-BERT*, una variante de BERT diseñada para generar representaciones semánticas de frases. Los atributos categóricos y numéricos se transforman conjuntamente en un espacio latente continuo, sobre el cual se aplica el procedimiento de agrupamiento de MDAV. La propuesta preserva las garantías de k-anonimato y mejora la coherencia semántica entre los registros agrupados. Los experimentos con datos mixtos muestran una menor pérdida de información y un coste computacional contenido respecto al MDAV clásico y a enfoques semánticos basados en ontologías.**

***Index Terms*—anonimización de datos, k-anonimato, microagregación, modelos de lenguaje, BERT, privacidad.**

I. Introducción

El creciente acceso a grandes volúmenes de datos de ámbitos sanitarios, gubernamentales y empresariales ha impulsado el desarrollo de métodos de anonimización que permitan compartir información sin comprometer la privacidad de las personas [1]. En este contexto, el *control de la confidencialidad estadística* (*Statistical Disclosure Control*, SDC) agrupa un conjunto de técnicas orientadas a evitar la reidentificación de individuos o la revelación de atributos sensibles mediante transformaciones controladas de los datos [2].

Las estrategias para alcanzar dicha protección suelen situarse entre dos extremos: los enfoques *utility-first* aplican métodos como la generalización, supresión o adición de ruido de forma heurística hasta alcanzar los niveles de protección suficientes para la publicación de los datos; por otro lado, los enfoques *privacy-first* se basan en modelos formales para guiar el proceso de anonimización. Entre dichos modelos, el k-anonimato se ha consolidado como uno de los enfoques más extendidos: garantiza que cada combinación de cuasi-identificadores aparezca en al menos k registros distintos [3], fijando así un riesgo máximo de reidentificación de $1/k$.

La microagregación permite asegurar el k-anonimato agrupando los registros en bloques de tamaño mínimo k y sustituyendo sus valores originales por un representante común, reduciendo el riesgo de reidentificación al tiempo que preserva parcialmente la estructura estadística del conjunto de datos.

Entre las técnicas de microagregación, el algoritmo *Maximum Distance to Average Vector* (MDAV) [4] se emplea como método para generar particiones que satisfacen el k-anonimato. Sin embargo, su aplicación en escenarios actuales presenta limitaciones relevantes. En particular, su formulación clásica opera exclusivamente con atributos numéricos y distancias euclídeas, lo que restringe su capacidad para tratar datos heterogéneos y, especialmente, variables categóricas cuyo contenido semántico no puede capturarse adecuadamente mediante métricas puramente geométricas.

En paralelo, los modelos de lenguaje preentrenados han transformado el procesamiento del lenguaje natural al generar representaciones vectoriales densas que capturan relaciones semánticas entre palabras, frases y conceptos. Entre ellos, BERT [5] destaca como un modelo basado en *transformers* que aprende representaciones contextuales profundas mediante tareas de enmascaramiento y predicción del contexto. Sobre esta arquitectura se construye *Sentence-BERT* [6], una variante entrenada con un esquema siamés para producir *embeddings* semánticos de alta calidad a nivel de frase o de registro completo. Estas representaciones permiten calcular similitudes en espacios vectoriales de forma consistente y eficiente, y resultan especialmente adecuadas en escenarios de anonimización en los que los atributos categóricos codifican información semántica que conviene preservar. Al incorporar esta información semántica en una codificación conjunta de los atributos categóricos y numéricos, es posible obtener datos anonimizados que mantienen relaciones conceptuales significativas sin recurrir a ontologías externas ni jerarquías específicas.

En este trabajo proponemos integrar representaciones semánticas generadas mediante modelos de lenguaje en el proceso de microagregación. Nuestro objetivo es extender MDAV a escenarios con datos mixtos que incluyen tanto atributos numéricos como categóricos, evitando la dependencia de recursos externos como ontologías o jerarquías. Para ello, cada registro completo se codifica de manera conjunta en un espacio vectorial mediante un modelo *Sentence-BERT* preentrenado, de modo que la información semántica y la información numérica quedan integradas en una única representación latente. Sobre estas representaciones se aplica el algoritmo MDAV, obteniendo una microagregación híbrida que mantiene las garantías de k-anonimato, mejora la coherencia interna de los grupos formados y reduce la pérdida de información en comparación con la microagregación tradicional y con métodos semánticos basados en ontologías [7].

II. Conceptos preliminares

II-A. Microagregación y el algoritmo MDAV

La microagregación transforma un conjunto de datos en grupos de al menos k registros, sustituyendo los valores originales por un representante común. Esta estrategia garantiza el k-anonimato, ya que cada registro queda indistinguible de al menos $k-1$ elementos adicionales. Entre los distintos métodos existentes, el algoritmo MDAV [4] destaca por su simplicidad y buena relación entre privacidad y preservación de utilidad.

MDAV opera de forma iterativa siguiendo una estrategia de selección de elementos extremos. En cada paso, se calcula el centroide de los registros aún no agrupados y se identifica el registro que se encuentra a mayor distancia de dicho centro. A continuación, se localiza un segundo registro que maximiza la distancia respecto al primero. Con cada uno de estos dos puntos de referencia se forman grupos tomando sus $k-1$ vecinos más próximos, lo que da lugar a dos bloques k-anónimos por iteración. El procedimiento continúa mientras queden al menos $3k$ registros por procesar. Cuando el número restante se sitúa entre $2k$ y $3k$, los datos se dividen en dos grupos finales; si es inferior a $2k$, se agrupan todos en un único bloque.

El procedimiento completo se resume en el Algoritmo 1.

Algorithm 1 MDAV

Entrada: Conjunto de datos X con n registros y m cuasi-identificadores; parámetro k.
Salida: Una partición de X que satisfaga el k-anonimato.

while $|X| \geq 3k$ **do**
  a. Selección del centroide
  Calcular el centroide $\bar{x}$ de los registros restantes en X.
  Sea x_r el registro más alejado de $\bar{x}$ (distancia euclídea).
  Sea x_s el registro más alejado de x_r.
  b. Formación de dos grupos
  Formar el grupo C_r con x_r y sus $k-1$ vecinos más cercanos.
  Eliminar de X los registros pertenecientes a C_r.
  Formar el grupo C_s con x_s y sus $k-1$ vecinos más cercanos.
  Eliminar de X los registros pertenecientes a C_s.
end while
if $2k \leq |X| < 3k$ **then**
  Formar un grupo con el registro más alejado del centroide actual y sus $k-1$ vecinos más cercanos.
  Asignar los registros restantes a un segundo grupo.
end if
if $0 < |X| < 2k$ **then**
  Formar un único grupo con todos los registros restantes.
end if

II-B. Medida de distorsión para datos mixtos: NCP

La evaluación de la pérdida de información introducida por la microagregación requiere métricas capaces de reflejar el grado de generalización aplicado a los datos. En escenarios con atributos numéricos y categóricos, una medida ampliamente utilizada es el *Normalized Certainty Penalty* (NCP) [8], que permite cuantificar la distorsión de forma homogénea en dominios heterogéneos.

Sea A_j un atributo y sea $A_j(C_z)$ el conjunto de valores que toma dentro del grupo C_z. La penalización asociada a dicho atributo en el grupo se define como:

$$\text{NCP}_{A_j}(C_z) = \begin{cases} \dfrac{\text{range}\,(A_j(C_z))}{\text{range}\,(A_j(T))}, & \text{si } A_j \text{ es numérico,} \\ \dfrac{|\text{dom}(A_j(C_z))|}{|\text{dom}(A_j)|}, & \text{si } A_j \text{ es categórico,} \end{cases} \tag{1}$$

donde el rango de un atributo en un conjunto C_z se define como:

$$\text{range}(A_j(C_z)) = \text{máx}(A_j(C_z)) - \text{mín}(A_j(C_z)). \tag{2}$$

En el caso de atributos numéricos, esta penalización mide la fracción del rango global ocupado por los valores presentes en el grupo. Para atributos categóricos, cuantifica el número de categorías distintas dentro del grupo con respecto al dominio total del atributo.

Una vez calculadas las penalizaciones por atributo y grupo, el NCP total del conjunto anonimizado T' se obtiene mediante el promedio ponderado:

$$\text{NCP}(T') = \sum_{j=1}^{m} w_j \frac{1}{n} \sum_{C_z \in \mathcal{C}} \sum_{\mathbf{x}_i \in C_z} \text{NCP}_{A_j}(C_z), \tag{3}$$

donde w_j representa la importancia relativa del atributo A_j. Valores de NCP próximos a cero indican baja distorsión y, por tanto, mayor utilidad de los datos anonimizados; valores próximos a uno reflejan una pérdida sustancial de información.

III. Trabajo relacionado

La microagregación se ha consolidado como una técnica eficaz para lograr k-anonimato gracias al equilibrio que ofrece entre simplicidad del procedimiento y utilidad de los datos resultantes. El algoritmo MDAV original [4] agrupa los datos usando distancias euclídeas y reemplaza cada grupo por su centroide, minimizando la suma de errores cuadráticos dentro de los grupos. Para mejorar la flexibilidad de la agrupación, V-MDAV permite tamaños de grupo variables entre k y $2k-1$ mediante un factor de ganancia que controla el crecimiento de los grupos y aumenta la homogeneidad intragrupo [9]. Posteriormente, F-MDAV introdujo técnicas de reutilización de normas, precomputación y ordenaciones parciales para reducir la complejidad computacional en entornos de *big data* sin incrementar la distorsión [10], mientras que estrategias de particionado recursivo dividen el conjunto de datos en macroceldas y aplican MDAV de manera local, reduciendo la complejidad de cuadrática a casi lineal para tamaños de datos muy grandes [11].

El tratamiento de datos heterogéneos ha dado lugar a variantes que combinan diferentes tipos de atributos en un marco unificado. MDAV-generic extiende el algoritmo a atributos continuos, ordinales y nominales mediante funciones de agregación específicas para cada tipo (como la media, la mediana o la moda) y procedimientos de normalización adecuados, de forma que se preserva en la medida de lo posible la semántica de cada dominio [12]. Por su parte, las versiones con ponderaciones y sensibilidades adaptativas introducen pesos distintos por atributo para reflejar su importancia relativa y su contribución al riesgo o a la pérdida de utilidad, ajustando la métrica de distancia al contexto de aplicación [13]. En cuanto a atributos categóricos, Martínez et al. propusieron *Semantic-MDAV* [7], que sustituye la distancia euclídea por una medida de similitud semántica basada en ontologías como WordNet y define centroides semánticos que resumen el contenido de cada grupo. Aunque esta aproximación mejora la coherencia de los grupos cuando existen jerarquías semánticas

bien definidas, su dependencia de ontologías externas y la necesidad de tratar manualmente valores no cubiertos por ellas limita su aplicabilidad y conlleva un coste computacional elevado.

IV. Microagregación híbrida propuesta

Nuestra propuesta, denominada *MDAV-SBERT*, integra modelos de lenguaje en el proceso de microagregación para gestionar de manera conjunta atributos numéricos y categóricos. La idea central consiste en codificar cada registro completo en un espacio vectorial común mediante un modelo de lenguaje preentrenado, obteniendo así una representación uniforme apta para el cálculo de similitudes. Sobre este espacio latente se aplica posteriormente el algoritmo MDAV clásico, manteniendo las garantías de k-anonimato y la lógica de agrupamiento basada en la selección de puntos extremos respecto al centroide.

Sea $\mathcal{D}$ un conjunto de n registros con atributos numéricos $\mathbf{x}_u m \in \mathbb{R}^p$ y atributos categóricos $x_t ext \in \mathcal{T}$. El método recibe como entrada el parámetro k y un modelo de lenguaje preentrenado LM (*Sentence-BERT*). A continuación se describen sus fases principales:

1. *Preprocesamiento:* Los valores numéricos y los atributos categóricos con valores vacíos o nulos se sustituyen por un marcador neutro. A continuación, los atributos de cada registro se concatenan para formar una cadena mixta $v_i \in V$. Por ejemplo, para un registro con valores (x_0, x_1, x_2) para los atributos *attr0*, *attr1*, y *attr2*, la cadena mixta resultante sería "`'attr0: ` x_0`, attr1: ` x_1`, attr2: ` x_2`'`". Esta cadena se transforma posteriormente en una representación vectorial empleando el modelo de lenguaje, obteniéndose $S = \mathrm{LM}(V)$ mediante codificación por lotes para reducir el tiempo de ejecución.
   Reducción opcional de la dimensionalidad: El método contempla tres configuraciones para ajustar la dimensión latente empleada por MDAV: (i) *full*, que mantiene íntegramente la dimensión original del embedding; (ii) *sqrt*, que proyecta los vectores a $d = \lceil p^{1/2} \rceil$; (iii) *cube*, que proyecta a $d = \lceil p^{1/3} \rceil$. En todos los casos, p denota el número de atributos de los cuasi-identificadores en el esquema original (antes del preprocesamiento), y d se restringe a no superar la dimensión del embedding generado por LM.
2. *Agrupamiento de los registros:* Mientras existan suficientes elementos por agrupar, se calcula el centroide global en el espacio latente S y se identifica el registro más alejado de dicho centroide. A continuación se selecciona el registro más distante de este primero, y ambos actúan como puntos de referencia para formar grupos de tamaño k añadiendo sus $k-1$ vecinos más próximos. El procedimiento se repite de manera iterativa hasta asignar todos los registros.
3. *Generalización:* Para cada grupo C_j se calcula la media de los atributos numéricos y la moda de los atributos categóricos. Todos los registros del grupo se reemplazan por estos valores representativos, garantizando que los k elementos sean indistinguibles en los cuasi-identificadores.

Algorithm 2 MDAV-SBERT

Entrada: datos $\mathcal{D}$, parámetro k y modelo de lenguaje LM.
Salida: partición k-anónima.
a. Preprocesamiento
Sustituir valores vacíos por un marcador neutro y concatenar los atributos numéricos y categóricos de cada registro en una cadena mixta v_i.
Transformar las cadenas $v_i \in V$ en vectores latentes $S = \mathrm{LM}(V)$ mediante *Sentence-BERT*.
Inicializar el conjunto X a partir de S con todos los registros e índices y precomputar y almacenar normas en el conjunto X para la reutilización en el cálculo de las distancias entre registros.
b. Agrupamiento de los registros
while $|X| \geq 3k$ **do**
    Calcular el centroide global $\bar{z}$ de $\{z_i : i \in X\}$.
    Seleccionar x_r como el registro más alejado de $\bar{s}$.
    Seleccionar x_s como el registro más alejado de x_r.
    Formar el grupo C_r con x_r y sus $k-1$ vecinos más cercanos a x_r; eliminar C_r de X.
    Formar el grupo C_s con x_s y sus $k-1$ vecinos más cercanos a x_s; eliminar C_s de X.
end while

if $2k \leq |X| < 3k$ **then**
    Calcular el centroide $\bar{z}$ de los registros restantes.
    Formar un grupo C con el registro más alejado de $\bar{z}$ y sus $k-1$ vecinos más cercanos; eliminar C de X
else if $0 < |X| < 2k$ **then**
    Formar un único grupo C con todos los registros restantes.
end if

c. Generalización
for cada grupo C **do**
    Sustituir los atributos numéricos por su media y los categóricos por el valor más frecuente.
end for
return Conjunto anonimizado con grupos generalizados.

Además, se incorporan pequeñas variaciones al esquema clásico de MDAV con el fin de adaptarlo mejor al espacio latente generado por el modelo de lenguaje. En primer lugar, la proximidad entre registros y centroides se evalúa mediante la distancia del coseno,

$$d_{Sc}(\mathbf{c}, \mathbf{Y}) = 1 - \frac{\langle \mathbf{Y}, \mathbf{c} \rangle}{\|\mathbf{Y}\| \, \|\mathbf{c}\|}, \tag{4}$$

donde $\langle \mathbf{Y}, \mathbf{c} \rangle$ denota el producto escalar entre el centroide y los vectores restantes del dataset $Y = \{y_1, ..., y_n\}$, mientras que $\|\mathbf{c}\|$ y $\|\mathbf{Y}\|$ corresponden a sus normas euclídeas. Esta medida resulta especialmente adecuada en espacios de alta dimensión, donde el ángulo entre vectores captura relaciones semánticas con mayor fidelidad que la distancia euclídea. Asimismo, para reducir el coste computacional, se reutilizan cálculos intermedios, en particular la norma $\|\mathbf{Y}\|$ y los productos escalares, que permanecen invariantes mientras no cambian los centroides, evitando recalcular desde cero todas las distancias en cada iteración. Finalmente, la selección de los $k-1$ vecinos más próximos se acelera mediante técnicas de ordenación parcial, que permiten identificar los elementos requeridos sin ordenar completamente el conjunto restante, manteniendo el comportamiento del algoritmo original pero con una mejora apreciable en eficiencia.

V. Evaluación experimental

V-A. Metodología experimental

El conjunto de datos *Adult* de la UCI contiene 48 842 registros y 14 atributos demográficos y socioeconómicos. Para la evaluación se definieron dos configuraciones de cuasi-identificadores (QI):

- *QI7*: conjunto compuesto por siete atributos categóricos de tipo demográfico y socioeconómico (*workclass*,

marital-status, *occupation*, *relationship*, *race*, *sex* y *native-country*). Este escenario reproduce un conjunto de QI estrictamente categóricos.

- *QI14*: conjunto formado por la totalidad de los atributos disponibles. Este escenario representa un caso mixto que combina atributos numéricos y categóricos. Solo se evaluó *MDAV-SBERT*, dado que *Semantic-MDAV* no puede operar con atributos numéricos.

V-A1. Variaciones experimentales: En ambos escenarios se analizó el comportamiento de *MDAV-SBERT* variando el tamaño del lote (*batch size*) empleado durante la codificación del modelo preentrenado, con valores de 128, 256, 512 y 1024, así como la dimensionalidad de la proyección latente. Para ello, se evaluaron las siguientes variantes:

- *Semantic-MDAV*: implementación de referencia basada en ontologías léxicas, donde la distancia euclídea se sustituye por una medida semántica (por ejemplo, Wu–Palmer) y los centroides se construyen a partir de WordNet. Este método solo puede aplicarse en escenarios puramente categóricos.
- *MDAV-SBERT (full)*: variante que emplea los embeddings completos generados por *Sentence-BERT*, sin aplicar reducción dimensional.
- *MDAV-SBERT (sqrt)*: variante que proyecta los embeddings a una dimensión latente $d_{\text{sqrt}} = \lceil p^{1/2} \rceil$, donde p es el número de atributos de los cuasi-identificadores en su definición original (antes del preprocesamiento). Si d_{sqrt} supera la dimensión del embedding generado por el modelo de lenguaje, se utiliza esta última como cota superior.
- *MDAV-SBERT (cube)*: variante análoga que utiliza $d_{\text{cube}} = \lceil p^{1/3} \rceil$, lo que induce una compresión más agresiva de la representación latente sin exceder la dimensión original del embedding.

V-A2. Diseño experimental: El parámetro de privacidad k toma valores entre 10 y 100 en incrementos regulares, con el fin de analizar la evolución del compromiso entre privacidad y utilidad. Para cada combinación de método, configuración de cuasi-identificadores, tamaño de lote, esquema de truncamiento y valor de k, se realizaron tres ejecuciones independientes y se reportó la media de las métricas obtenidas.

Se llevaron a cabo tres experimentos complementarios:

1. *Evaluación de utilidad*: mediante el *Normalized Certainty Penalty* (NCP), calculado sobre el conjunto completo de cuasi-identificadores.
2. *Análisis de tiempos por fase*: evaluando el coste computacional asociado a cada una de las tres etapas del proceso:
   - *codificación*: obtención de los embeddings generados por *Sentence-BERT* para los atributos categóricos y numéricos, así como la posible proyección a una dimensión latente reducida;
   - *microagregación*: ejecución del algoritmo MDAV sobre las representaciones resultantes;
   - *generalización*: sustitución de los valores originales por los correspondientes centroides numéricos o valores modales en atributos categóricos.

   Cada gráfica de tiempos por fase muestra el tiempo medio de ejecución de cada etapa, obtenido promediando los valores registrados para todos los niveles de anonimización comprendidos entre $k = 10$ y $k = 100$.
3. *Tiempo medio por k*: análisis detallado de la evolución del coste computacional a medida que aumenta el nivel de anonimización impuesto por el parámetro k.

Todos los experimentos se ejecutaron en un entorno WSL con Ubuntu, equipado con 32 GiB de memoria RAM y ejecutándose sobre un sistema dotado de un procesador AMD Ryzen 9 9900X (12 núcleos, 24 hilos) y una GPU NVIDIA GeForce RTX 2060.

V-B. Resultados y discusión

V-B1. Resultados para QI7: La Fig. 1 muestra la evolución del NCP en función de k para la configuración *QI7*. Se observa que todas las variantes de *MDAV-SBERT* presentan niveles de distorsión notablemente inferiores a los de *Semantic-MDAV*. Mientras que este último supera el 70 % de NCP incluso para valores moderados de k, las variantes basadas en modelos de lenguaje se sitúan en el intervalo aproximado del 20–47.5 %. En conjunto, los resultados indican que las representaciones latentes generadas por *MDAV-SBERT* permiten obtener agrupamientos más consistentes que los derivados de la distancia ontológica empleada en *Semantic-MDAV*.

En cuanto a las variantes de truncamiento, las tres configuraciones (*full*, *sqrt* y *cube*) exhiben una evolución del NCP con forma similar, aunque con desplazamientos sistemáticos en su nivel absoluto. La versión *cube* presenta consistentemente los valores más elevados dentro del grupo de métodos basados en BERT, seguida por *sqrt*, mientras que la configuración *full* obtiene la distorsión más reducida. No obstante, incluso las variantes con mayor reducción dimensional se mantienen claramente por debajo de los valores obtenidos mediante *Semantic-MDAV*.

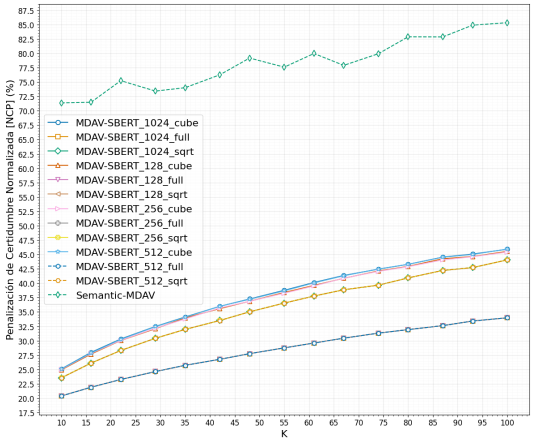


Figura 1. Evolución del NCP en función de k para la configuración *QI7*.

El desglose temporal por fases, representado en la Fig. 2, muestra que la codificación BERT constituye la parte dominante del coste total en todas las configuraciones. La fase de MDAV incrementa su tiempo en las variantes sin reducción dimensional (*full*), consecuencia directa de operar en un espacio latente de mayor dimensión; aun así, su contribución permanece claramente por debajo de la fase de codificación. El tamaño del lote presenta un impacto limitado: los tiempos de codificación son casi constantes en los cuatro valores

considerados y, en la fase de MDAV, únicamente se observa una ligera mejora al aumentar el tamaño de lote en las variantes *full*. El lote de tamaño 128 presenta un tiempo de codificación ligeramente superior.

Este comportamiento sugiere que, para la configuración *QI7*, el coste computacional en la fase de codificación está dominado por componentes que no se benefician de manera significativa del incremento del paralelismo en GPU al aumentar el *batch size*. La penalización observada para el lote de tamaño 128 es coherente con una menor eficiencia computacional asociada a lotes pequeños, mientras que para tamaños de 256 o superiores no se aprecian mejoras significativas, lo que indica un comportamiento estable en esta fase.

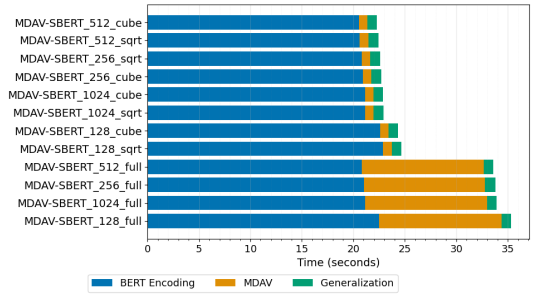


Figura 2. Desglose temporal por fases (codificación, MDAV y generalización) para *QI7*.

La Fig. 3 muestra el tiempo medio total en función de k. *Semantic-MDAV* resulta significativamente menos eficiente que cualquier variante de *MDAV-SBERT*; por ejemplo, para $k = 10$ la relación aproximada es de $\frac{2000\,\text{s}}{25\text{–}30\,\text{s}} \approx 67\text{–}80\times$. Entre las variantes basadas en BERT, las representaciones sin reducción (*full*) presentan tiempos superiores debido a la mayor dimensionalidad del espacio latente, en consonancia con lo observado en la Fig. 2. No obstante, todas las variantes se mantienen en el rango de decenas de segundos y muestran una tendencia decreciente conforme aumenta k, consecuencia del menor número de grupos a formar para niveles de anonimización más elevados.

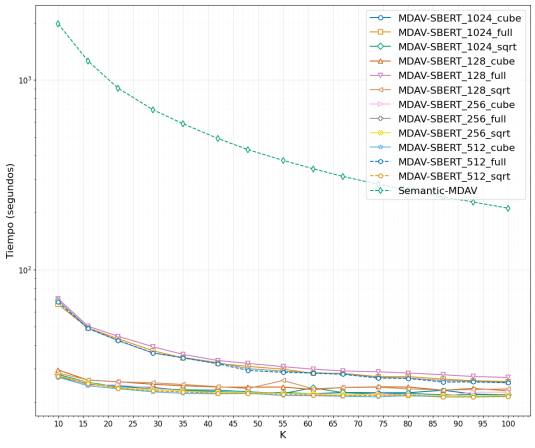


Figura 3. Tiempo medio de ejecución en función de k para *QI7*, con el eje vertical representado en escala logarítmica.

V-B2. Resultados para QI14: La Fig. 4 presenta los valores de NCP para la configuración *QI14*, que incluye todos los atributos categóricos y numéricos. En comparación con *QI7*, los niveles de distorsión abarcan un rango más amplio (aprox. 21–64 %), reflejando el incremento de complejidad al considerar un conjunto más extenso y heterogéneo de cuasi-identificadores. Las diferencias entre las variantes de truncamiento reproducen el patrón previamente observado: *cube* induce mayor pérdida de utilidad, *full* obtiene los valores más bajos de NCP y *sqrt* se sitúa en una posición intermedia.

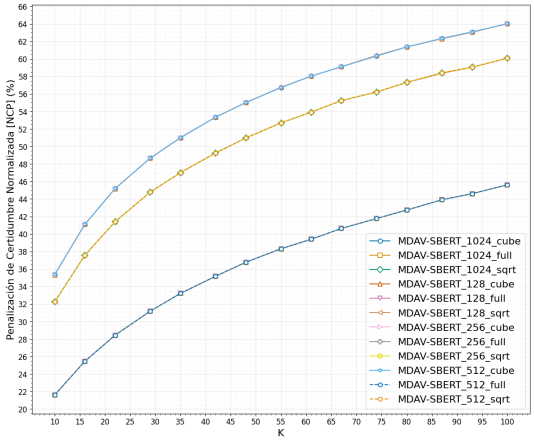


Figura 4. Evolución del NCP en función de k para la configuración *QI14*.

La Fig. 5 muestra el tiempo desglosado por fases. El comportamiento general coincide con el observado en *QI7*: la codificación BERT domina el tiempo total, mientras que la fase de MDAV incrementa su coste en la variante *full* debido a la mayor dimensionalidad. En comparación con *QI7*, los tiempos totales son ligeramente superiores, lo cual es esperable por el mayor número de atributos considerados en *QI14*. El tamaño de lote tiene un impacto reducido y las diferencias apreciables se concentran en las variantes sin reducción. A diferencia de *QI7*, el lote de tamaño 128 no presenta un desfase significativo respecto al resto.

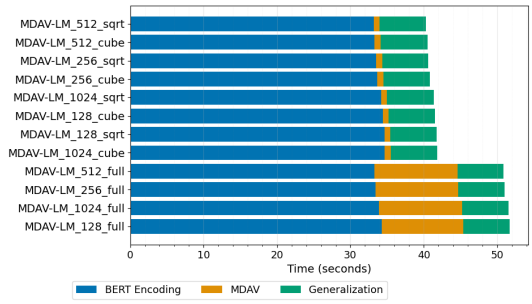


Figura 5. Desglose temporal por fases para *QI14*.

El tiempo medio total en función de k, representado en la Fig. 6, muestra también una tendencia decreciente a medida que aumenta el nivel de anonimización. Las variantes *full* presentan tiempos mayores, mientras que *sqrt* y *cube* reducen de forma apreciable el coste total con un impacto moderado en la utilidad. En todos los casos los tiempos se mantienen en el rango de decenas de segundos, con un comportamiento análogo al observado en *QI7*.

V-B3. Discusión conjunta: Los resultados evidencian que *MDAV-SBERT* ofrece un compromiso favorable entre utilidad y coste computacional en ambas configuraciones analizadas.

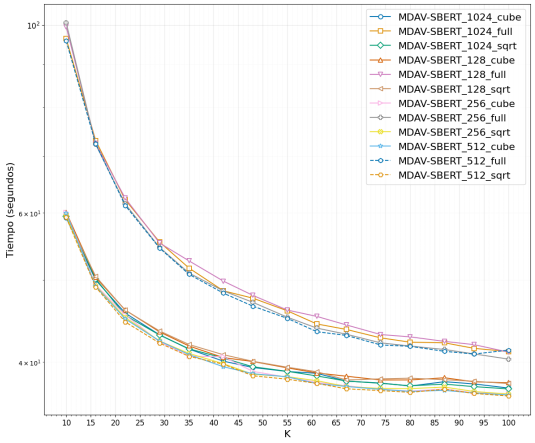


Figura 6. Tiempo medio de ejecución en función de k para *QI14*, con el eje vertical representado en escala logarítmica.

Como cabía esperar, la distorsión medida mediante NCP es mayor en *QI14* que en *QI7*, reflejando la mayor complejidad y heterogeneidad del conjunto de cuasi-identificadores. Sin embargo, en ambos casos las variantes basadas en modelos de lenguaje superan de manera consistente a *Semantic-MDAV*, tanto en términos de utilidad como de eficiencia computacional.

Las diferencias entre las variantes de *MDAV-SBERT* son estables en las dos configuraciones: la versión *full* logra la menor distorsión, seguida por *sqrt*, mientras que *cube* concentra la mayor pérdida de utilidad. No obstante, incluso las configuraciones más agresivas en la reducción dimensional mantienen niveles de NCP muy inferiores a los de *Semantic-MDAV*, lo que indica que las proyecciones latentes compactas preservan suficiente estructura semántica para producir agrupamientos coherentes.

Desde la perspectiva computacional, las configuraciones *sqrt* y *cube* reducen de manera significativa el tiempo total frente a la variante *full*, especialmente en *QI14*, donde la dimensionalidad del embedding original es mayor. En cualquier caso, todas las variantes de *MDAV-SBERT* se mantienen en un rango de ejecución de decenas de segundos, muy inferior al requerido por *Semantic-MDAV*. En conjunto, los resultados demuestran que la integración de modelos de lenguaje en microagregación constituye una alternativa eficaz, flexible y computacionalmente viable para el tratamiento de datos categóricos y mixtos.

VI. Conclusiones

Se ha presentado una técnica de microagregación híbrida basada en modelos de lenguaje que extiende el algoritmo MDAV a conjuntos de datos mixtos. La propuesta codifica conjuntamente los atributos categóricos y numéricos mediante *Sentence-BERT*, utilizando las representaciones obtenidas como espacio latente sobre el que se ejecuta el esquema de microagregación de MDAV con reutilización de distancias. Asimismo, se han incorporado configuraciones de reducción de dimensionalidad basadas en truncamientos dependientes del número de cuasi-identificadores, lo que permite ajustar automáticamente la dimensión latente para equilibrar coste computacional y utilidad.

La evaluación experimental en el conjunto *Adult* para las configuraciones *QI7* y *QI14* muestra que las variantes de *MDAV-SBERT* obtienen valores de *Normalized Certainty Penalty* inferiores y tiempos de ejecución considerablemente más reducidos que *Semantic-MDAV*. En particular, las configuraciones con reducción dimensional mantienen una utilidad muy próxima a la versión sin truncamiento, al tiempo que disminuyen de forma significativa el tiempo de codificación. Estos resultados ponen de manifiesto que la incorporación de modelos de lenguaje preentrenados en microagregación ofrece un compromiso adecuado entre privacidad, utilidad y eficiencia, sin depender de recursos ontológicos externos.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el MCIN/AEI/10.13039/501100011033 y por "ERDF A way of making Europe", en el marco de los proyectos PID2021-123637NB-I00 ("CURLING") y PID2024-156914OB-C42 ("CLEARING-IT"), así como por los fondos NextGenerationEU/PRTR a través de INCIBE (proyecto "HERMES" y Cátedra INCIBE-URV de ciberseguridad). La primera autora cuenta con el apoyo de una beca Martí i Franquès (2025PMF-PIPF-13) de la Universitat Rovira i Virgili.

Referencias

[1] A. Badshah, A. Daud, R. Alharbey, A. Banjar, A. Bukhari, and B. Alshemaimri, "Big data applications: overview, challenges and future," *Artificial Intelligence Review*, vol. 57, no. 11, p. 290, 2024.

[2] A. Hundepool, J. Domingo-Ferrer, L. Franconi, S. Giessing, E. S. Nordholt, K. Spicer, and P.-P. De Wolf, *Statistical disclosure control*. John Wiley & Sons, 2012.

[3] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," 1998.

[4] J. Domingo-Ferrer and J. M. Mateo-Sanz, "Practical data-oriented microaggregation for statistical disclosure control," *IEEE Transactions on Knowledge and data Engineering*, vol. 14, no. 1, pp. 189–201, 2002.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2019.

[6] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3982–3992.

[7] S. Martínez, D. Sánchez, and A. Valls, "Semantic adaptive microaggregation of categorical microdata," *Computers & Security*, vol. 31, no. 5, pp. 653–672, 2012.

[8] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. W.-C. Fu, "Utility-based anonymization using local recoding," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 785–790.

[9] A. Solanas, A. Martínez-Ballesté, and J. Domingo-Ferrer, "V-mdav: A multivariate microaggregation with variable group size," pp. 917–925, 2006.

[10] A. Rodríguez-Hoyos, J. Estrada-Jiménez, D. Rebollo-Monedero, A. M. Mezher, J. Parra-Arnau, and J. Forné, "The fast maximum distance to average vector (f-mdav): An algorithm for k-anonymous microaggregation in big data," *Engineering Applications of Artificial Intelligence*, vol. 90, p. 103531, 2020.

[11] E. Pallarès, D. Rebollo-Monedero, A. Rodríguez-Hoyos, J. Estrada-Jiménez, A. M. Mezher, and J. Forné, "Mathematically optimized, recursive prepartitioning strategies for k-anonymous microaggregation of large-scale datasets," *Expert Systems with Applications*, vol. 144, p. 113086, 2020.

[12] J. Domingo-Ferrer and V. Torra, "Ordinal, continuous and heterogeneous k-anonymity through microaggregation," *Data Mining and Knowledge Discovery*, vol. 11, no. 2, pp. 195–212, 2005.

[13] Z. Zhang, B. Han, H.-C. Chao, F. Sun, L. Uden, and D. Tang, "A new weight and sensitivity based variable maximum distance to average vector algorithm for wearable sensor data privacy protection," *IEEE Access*, vol. 7, pp. 104 045–104 056, 2019.

Patient-centric drug monitoring with certified awareness: a blockchain-enabled IoT platform

Jaume Ramis-Bibiloni
Universitat de les Illes Balears
Crta. Valldemossa Km. 7,5, Palma
jaume.ramis@uib.cat

M. Magdalena Payeras-Capellà
Universitat de les Illes Balears
Crta. Valldemossa Km. 7,5, Palma
mpayeras@uib.cat

Bartomeu Alorda-Ladaria
Universitat de les Illes Balears
Crta. Valldemossa Km. 7,5, Palma
tomeu.alorda@uib.cat

Macià Mut-Puigserver
Universitat de les Illes Balears
Crta. Valldemossa Km. 7,5, Palma
macia.mut@uib.cat

***Abstract*—The conservation of ambient-sensitive pharmaceutical drugs is essential for ensuring their safety and effectiveness, yet proper storage is often compromised once medications reach patients' homes. This paper presents a patient-centric monitoring platform that integrates IoT sensing and blockchain technology to address this challenge. Sensor-equipped devices track temperature, humidity, light, and movement, generating real-time alerts that enable patients to take timely corrective actions. Blockchain-based smart contracts certify patient acknowledgment when a drug becomes non-consumable, ensuring transparency and accountability. By combining cryptographic sensors with an Ethereum-compatible protocol, the system provides secure end-to-end monitoring and supports remote supervision by healthcare providers. Performance evaluation demonstrates the cost-efficiency and feasibility of the proposed approach for reliable drug conservation monitoring.** [1]

***Index Terms*—Blockchain, IoT, Smart contracts, Ambient-sensitive drugs, Drug monitoring, Certified awareness**

I. Introduction

Ambient-sensitive pharmaceutical drugs require strict storage conditions to maintain their safety and effectiveness, yet once dispensed to patients, conservation control is often lost. Recent studies show that improper home storage is widespread: only 2.8% of patients kept thermolabile drugs within the recommended 2–8 °C range, while more than 97% experienced deviations that may compromise drug integrity. This highlights a critical gap in drug monitoring beyond hospital and pharmacy environments.

To address this challenge, we propose a *patient-centric IoT–blockchain platform* that enables real-time monitoring of drug conservation conditions at home. Sensor-equipped devices measure temperature, humidity, light, and movement, transmitting data to a gateway that triggers alerts when deviations occur. Patients can take corrective actions, and significant events are immutably recorded on a blockchain to certify patient awareness, ensuring transparency and accountability.

Unlike prior work focused mainly on supply-chain traceability, our approach targets conservation at the point of use. Building on our previous IoT-based monitoring system, this work introduces cryptographic in-house sensors and a blockchain-based certified awareness mechanism implemented through Ethereum smart contracts. This integration supports secure, verifiable, and end-to-end monitoring of drug quality under patient supervision.

[1]This work is an extended summary of our previously published work [2] that can be downloaded at https://doi.org/10.1007/s12083-025-02120-7.

II. System description

A. System Structure

The proposed platform builds on the architecture introduced in [1] and consists of five main components (Fig. 1): sensor nodes, a gateway, a communication network, a central server, and a blockchain backend. While this work focuses on the blockchain-based protocol, a brief overview of the system is provided below; further hardware details can be found in [1].

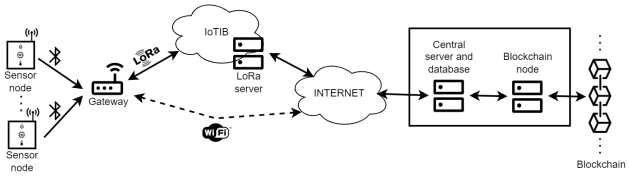


Fig. 1. Overview of the drug conservation monitoring platform.

Sensor node: A custom microcontroller collects temperature, humidity, UV radiation, and vibration data, integrating Elliptic Curve Cryptography (ECC) for secure authentication. The node operates in datalogger or broadcast mode and is physically attached to the drug.

Gateway: The gateway receives sensor data via BLE, supports WiFi or LoRa communication, and uses a secure ATECC608A element for ECC-based authentication. It includes LEDs, a buzzer, and a button to manage alarms and user interactions.

Communication network: Gateways transmit data either through WiFi (when available) or LoRa, which routes information to the central server through the Internet.

Central server and database: An Ubuntu-based architecture processes and stores all data using a set of Dockerized *Node.js* services. The system employs PostgreSQL for storage and a dedicated blockchain interface for smart contract interactions.

Blockchain network: The platform deploys smart contracts on Polygon, an EVM-compatible blockchain, enabling efficient and low-cost certified awareness operations.

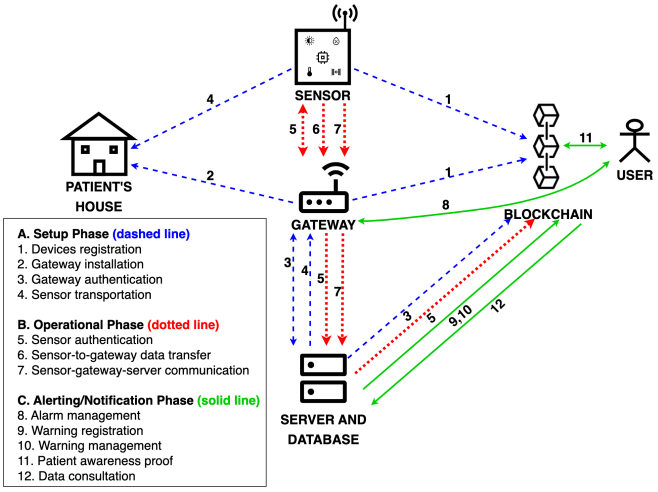


Fig. 2. Flowchart of the drug conservation monitoring process.

B. System Operation

For clarity, we distinguish two concepts used throughout the platform: **(i) Alarms**—sound/light signals generated by the gateway to warn patients of improper storage; **(ii) Warnings**—records stored on the blockchain indicating sustained deviations or connectivity failures.

Figure 2 summarizes the operational workflow, organized into three phases.

A. Setup Phase Devices are registered with identifiers and public keys, and their information is stored both in the database and on the blockchain. A technician installs and configures the gateway at the patient's home (WiFi or LoRa). Gateways authenticate with the server via ECC signatures to obtain renewable certificates. Sensors, provided with the medication, operate in *datalogger* mode during transport and later synchronize with the gateway.

B. Operational Phase Upon detection, sensors authenticate via ECC-based challenge–response, after which their certificates are registered and periodically updated on the blockchain. The sensor uploads logged data to the gateway, which then switches it to *broadcasting* mode for continuous monitoring. Measurements collected every *sensing time* (10 s) are aggregated every *sending time* (10 min) (min/max/avg) and sent to the server, where they are evaluated against conservation thresholds.

C. Alerting and Blockchain Phase Gateways generate local alarms when deviations are detected. The server analyzes stored data to identify five types of warnings (connectivity, temperature, humidity, light, movement), which are created or updated following the protocol in Subsection III-A2. Smart contracts determine whether a drug remains suitable for use or must be discarded. If disposal is required, the patient receives a notification and must acknowledge it; this acknowledgment is immutably recorded on the blockchain as certified awareness. Authorized stakeholders may later consult stored values, device certificates, and warning records through the smart contracts.

III. Blockchain-Based Data Management Protocol

This section summarizes the blockchain component of the proposed platform, which relies on smart contracts to ensure secure, auditable, and decentralized management of device identities, certified awareness events, and monitoring data. Other system elements (hardware, server architecture) lie outside the scope of this conference version and are described in [1].

The blockchain is responsible for: (i) registering and maintaining certificates for gateways and sensor nodes, (ii) recording and processing conservation warnings, including certified patient awareness, and (iii) enabling authorized stakeholders to query relevant drug-monitoring information. Smart contracts are carefully optimized to reduce gas consumption and ensure scalability, efficiency, and interoperability.

The protocol is organized into three subprotocols:

1) **Subprotocol 1: Device and Certificate Management** Handles registration, authentication, and certificate updates for all gateways and sensors.
2) **Subprotocol 2: Alarm and Warning Management with Certified Awareness** Manages warning creation and updates, executes suitability evaluation via smart contracts, and records patient acknowledgment on the blockchain.
3) **Subprotocol 3: Data Visualization** Provides blockchain-based access to stored warnings, device information, and certificate records.

Due to space limitations, only Subprotocol 2 is described in this short version. A complete description of Subprotocols 1 and 3 can be found in [2].

A. Subprotocol 2: Alarm and Warning Management with Awareness Notification

This subprotocol governs the creation, update, and processing of alarms and blockchain-registered warnings related to drug conservation. Five warning types are considered: one for connectivity loss between the gateway and the server, and four linked to monitored environmental parameters (temperature, humidity, light, and movement).

Gateways perform basic local processing and may trigger light or sound *alarms* to notify the patient immediately; however, these alarms are not stored on-chain. In contrast, *warnings* represent instances of sustained improper conservation or connectivity loss and are immutably recorded on the blockchain.

Subprotocol 2 comprises three main functionalities:

- **Alarm and Warning Management Procedure:** Detection of deviations and preliminary assessment of collected sensor data (Subsection III-A1).
- **Warning Registration, Update, and Processing:** Server-side creation and update of on-chain warnings and execution of the smart contract logic to determine conservation status (Subsection III-A2).
- **Warning Notification and Certified Awareness:** If the smart contract determines that a drug is no longer suitable for consumption, a notification is issued to the patient. Their acknowledgment is then recorded on the blockchain as certified awareness (Subsection III-A3).

This subprotocol ensures that deviations are reliably detected, verified, and escalated, while patient acknowledgment of critical warnings is securely documented on the blockchain.

1) Alarm and Warning Management Procedure: After each *sending time* interval, the gateway transmits a summary packet (average, minimum, and maximum values) to the server, which then executes the alarm/warning management logic. If no packet is received for a predefined period, the system registers a *disconnection* warning on the blockchain. Gateways buffer data locally during outages and forward it once connectivity is restored.

Two monitoring procedures are employed: one for light and movement, where warnings are generated automatically when values exceed predefined limits, and another for temperature and humidity, which requires analyzing both measured values and their duration. For these parameters, readings are categorized as *in range*, *out of range*, or *extreme*, based on standard conservation thresholds (e.g., 2–8 °C for temperature).

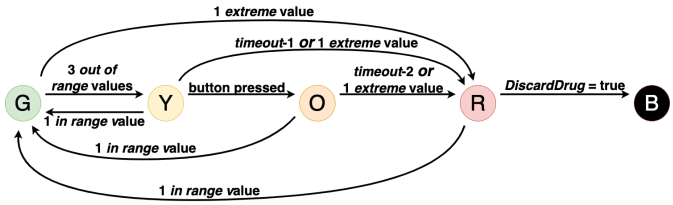


Fig. 3. Alarm/warning states for *temperature and humidity* monitoring.

Figure 3 summarizes the warning and state transitions for temperature and humidity, with each state represented by a gateway LED color. Table I outlines the five states, associated alarms, required patient actions, and blockchain interactions. Minor deviations (*green*, *yellow*, *orange*) do not trigger blockchain events, while persistent or critical deviations (*red*, *black*) result in on-chain warning registration or certified awareness recording.

To prevent indefinite state persistence, timeout mechanisms ensure progression through states based on measured conditions and user acknowledgment. Severe deviations or prolonged exposure escalate the state, while patient acknowledgment (gateway button press) affects transitions between intermediate states. The decision logic is implemented through Algorithm *ProcessWarning*, whose full description is provided in [2].

All five warning types rely on a common data structure w (specified in [2]). For temperature and humidity, warnings may remain *open* while in the *red* state, and are *closed* once the drug returns to a safe condition or becomes non-consumable (transition to *black*). Light, movement, and disconnection warnings do not require updates and are always stored with the *closed* field set to *true*.

2) Warning Registration, Update, and Processing: Warning registration on the blockchain follows the procedure in Algorithm *RegisterWarning* (detailed in [2]). After verifying the validity of the sensor certificate, the system populates the warning structure w and appends it to the sensor's warning list. Temperature and humidity warnings remain open until resolved, whereas light, movement, and disconnection warnings are stored as closed events.

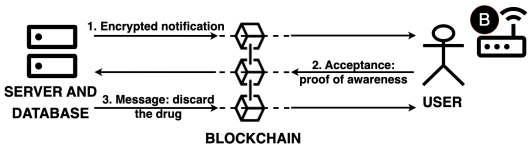


Fig. 4. Notification emission protocol for drug unsuitability.

Algorithm *UpdateWarning* (described in [2]) updates existing temperature and humidity warnings by extending their duration and incorporating new summary values. Updates are only permitted while the warning remains open and the corresponding certificate is valid.

Both registration and update procedures invoke Algorithm *ProcessWarning*, which determines whether the drug remains suitable for consumption. This evaluation considers the drug's conservation parameters and the history of warnings associated with the sensor. For temperature and humidity, processing occurs after every new or updated warning; if an *in-range* value is detected (see Fig. 3), the warning is closed and the state returns to *green*. Otherwise, the system remains in *red* and executes the *DecisionFunction*. Light and movement warnings always trigger the decision analysis.

The *DecisionFunction* examines the full warning history to determine whether the drug must be discarded. This assessment may rely on expert-defined rules or automated algorithms. If the drug is deemed unsafe, a certified notification is issued to the patient (temperature/humidity warnings transition to the *black* state), as detailed in Subsection III-A3.

3) Warning Notification Issuing and Acceptance: When Algorithm *ProcessWarning* determines that a drug is no longer suitable for consumption, the central server generates a notification structure n, whose full description is provided in [2], and sends an encrypted, blockchain-registered alert to the patient. This certified notification ensures that (i) the patient is informed and will not consume a potentially harmful drug, and (ii) pharmacists retain verifiable evidence of awareness, preventing misuse and supporting safe re-prescription when necessary.

Blockchain technology provides an immutable ledger for storing both notification issuance and patient acknowledgment. Our scheme adapts the confidential registered eDelivery protocol of Mut-Puigserver et al. [3] to a two-party setting (server–patient), ensuring confidentiality, integrity, and non-repudiation for both the notification and the acceptance evidence.

The protocol (Fig. 4) operates as follows: (1) the server encrypts the notification using a variant of ElGamal and registers it on the blockchain via a DApp; (2) the patient explicitly accepts the notification through a blockchain transaction, producing non-repudiable proof of receipt; (3) the server finalizes the exchange, after which the patient decrypts the message and is formally made aware that the drug must be discarded.

If a patient does not acknowledge the notification within a predefined interval, a healthcare professional may intervene to ensure proper disposal. The protocol guarantees fairness and confidentiality, as only encrypted data are stored on-chain while awareness evidence remains publicly verifiable.

TABLE I
STATES, ALARMS, AND ACTIONS FOR *temperature and humidity*

State	Gateway Alarm	Description	Required Intervention	Blockchain Registration
Green (G)	Green light. No alarm.	Proper conservation.	No action required.	No registration needed.
Yellow (Y)	Yellow light. One beep per minute.	Initial deviation detected (pre-unsatisfactory conservation).	Patient should verify power supply and ensure the fridge door is closed. Press the gateway button to silence the buzzer.	No registration needed.
Orange (O)	Orange light.	Unsatisfactory conservation. Patient aware (gateway button pressed).	No further action required from the patient.	No registration needed.
Red (R)	Red light.	Persistent unsatisfactory conservation.	No immediate action required from the patient.	Warning registered on the blockchain; status remains *open*. State processed according to Algorithm *ProcessWarning*.
Black (B)	Blinking red light. Two beeps per minute.	Drug is no longer safe for consumption.	Patient must acknowledge the notification and discard the drug. Press the gateway button to stop the buzzer.	Certified notification issued on-chain, ensuring verifiable patient acknowledgment.

For further cryptographic details, readers may consult [3].

IV. IMPLEMENTATION AND PERFORMANCE ANALYSIS

To evaluate the proposed blockchain-based protocol, we developed and deployed three Solidity smart contracts implementing all functionalities described in Section III:

- *Drugs.sol*: Maintains conservation parameters for each drug.
- *Notifications.sol*: Stores encrypted notifications and patient acknowledgment records.
- *Protocol.sol*: Implements the core server logic, structured according to the three subprotocol groups defined in Section III.

This modular design supports clarity, maintainability, and efficient execution across all blockchain operations. The full source code is publicly available in our GitHub repository.[2]

The smart contracts were designed with gas efficiency in mind to ensure suitability for deployment on public Ethereum Virtual Machine (EVM) compatible networks such as Ethereum or Polygon. To quantify performance, we evaluated gas consumption using the `eth-gas-reporter` tool (*https://www.npmjs.com/package/eth-gas-reporter*), integrated into a Hardhat-based automated testing framework. To express these costs in euros, we used Polygon PoS as the reference network, given its low transaction fees and compatibility with the EVM. Costs were computed using the average gas price and the January 2025 exchange rate of 0.33 €/MATIC.

Table II presents the corresponding average deployment and execution costs in euros. Among deployment operations, `Protocol.sol` is the most expensive (average 0.3684 €), although this cost is incurred only once. Regarding repeated invocations, `newWarning()` is the most costly method (average 0.0261 €), but it is expected to run infrequently, as it is triggered only under abnormal conservation conditions. In contrast, functions such as `sensorCertificateUpdate` exhibit consistently low costs, making them suitable for periodic execution.

Overall, the empirical measurements demonstrate that the proposed blockchain-based infrastructure is both feasible and cost-effective for continuous drug conservation monitoring in patient-centric environments, with operational expenses remaining well within acceptable limits for real-world deployment.

[2]https://github.com/secomuib/DrugQualityConservationMonitoring

TABLE II
AVERAGE COSTS

Drugs	0.0749
Notifications	0.0433
Protocol	0.3684
addDrug	0.0083
gatewayRegistration	0.0070
sensorRegistration	0.0067
gatewayCertificateCreation	0.0077
sensorCertificateCreation	0.0124
sensorCertificateUpdate	0.0024
newWarning	0.0261
updateWarning	0.0125

V. CONCLUSIONS

This work introduced a blockchain-enabled IoT platform for patient-centric monitoring of ambient-sensitive drugs after they leave hospital facilities. Unlike traditional supply-chain–oriented solutions, the system supports real-time monitoring of key environmental parameters and employs a certified awareness mechanism to ensure accountability when drug integrity is compromised.

The three implemented smart-contract subprotocols manage device registration, alarm and warning processing, and data visualization, offering secure authentication, automated risk evaluation, and transparent access to conservation records. By combining blockchain immutability with encrypted notifications and off-chain storage, the platform ensures privacy, non-repudiation, and tamper resistance. Performance tests confirm low and stable execution costs on Polygon, demonstrating the protocol's feasibility for practical deployment.

ACKNOWLEDGMENTS

This work was supported by ISC III (DTS21/00089); and by Grant PID2021-122394OB-I00 (BLOBSEC) funded by MCIU/AEI/ 10.13039/501100011033 and by EU.

REFERENCES

[1] Frontera-Bergas, M., Vinaixa-Fernández , M., Oliver-Riera, B., Ramis-Bibiloni, J., Isern, E., Alorda-Ladaria, B.: "A Multi-Sensor IoT Platform for monitoring medicine storage beyond the hospital", in *Internet of Things*, vol. 33, pp. 101711, 2025.

[2] Ramis-Bibiloni, J., Payeras-Capellà, M.M., Alorda-Ladaria, B., Mut-Puigserver, M.: "Patient-centric drug quality conservation monitoring with certified awareness: a blockchain-enabled IoT platform", in *Peer-to-Peer Netw. Appl.*, vol. 18, art. 300, 2025.

[3] M. Mut-Puigserver, M. A. Cabot-Nadal and M. M. Payeras-Capellà, "Removing the Trusted Third Party in a Confidential Multiparty Registered eDelivery Protocol Using Blockchain," in IEEE Access, vol. 8, pp. 106855-106871, 2020.

Seguridad del paciente en IoMT: Una revisión sistemática de la literatura sobre IDS basados en el comportamiento

Jordi Doménech
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya (UPC)
c. Jordi Girona, 1-3
08034, Barcelona, Spain
jordi.domenech.fons@upc.edu

Isabel V. Martin-Faus
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya (UPC)
c. Jordi Girona, 1-3
08034, Barcelona, Spain
isabel.martin-faus@upc.edu

Saber Mhiri
Cybersecurity & Blockchain Research Group
i2CAT Foundation
c. Gran Capità, 2-4
08034, Barcelona, Spain
saber.mhiri@i2cat.net

Josep Pegueroles
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya (UPC)
c. Jordi Girona, 1-3
08034, Barcelona, Spain
josep.pegueroles@upc.edu

***Resumen*—Este artículo es un resumen extendido de la publicación [1]. La incorporación del Internet de las Cosas Médicas (IoMT) en la atención sanitaria ha mejorado la monitorización y el intercambio de datos clínicos, pero también ha incrementado la exposición a amenazas de ciberseguridad que pueden comprometer la seguridad del paciente. Para abordar estos riesgos, los *Behavior-based Intrusion Detection Systems* (BIDS) representan una solución prometedora al detectar ataques mediante el análisis del comportamiento de dispositivos y redes. Esta revisión sistemática de la literatura analiza 81 estudios recientes con el fin de (1) identificar los métodos y técnicas de BIDS aplicados al IoMT, (2) evaluar los ataques detectados y su impacto en la seguridad del paciente y (3) examinar la presencia de medidas de prevención o mitigación. Los resultados revelan una fuerte dependencia de las técnicas de inteligencia artificial, una carencia de datasets específicos de IoMT y una limitada consideración del impacto clínico y de las estrategias de mitigación frente a amenazas. Este trabajo pone de manifiesto la necesidad de soluciones de detección y defensa más completas y centradas en el paciente.**

***Index Terms*—Revisión sistemática de la literatura, internet de las cosas médicas, sistemas de detección de intrusos, ciberseguridad, entorno sanitario, inteligencia artificial.**

I. Introducción

La rápida expansión del Internet de las Cosas Médicas (IoMT, *Internet of Medical Things*) está transformando la atención sanitaria al facilitar la monitorización continua, el diagnóstico remoto y una mayor integración de datos clínicos. En contrapartida, esta conectividad incrementa la superficie de exposición a ciberataques capaces de comprometer, no solo la privacidad del paciente, pero también su integridad física. A diferencia del IoT generalista, los dispositivos IoMT interactúan directamente con funciones críticas de salud, por lo que cualquier alteración maliciosa puede generar consecuencias clínicas graves.

Entre las distintas estrategias de protección, los *Intrusion Detection Systems* (IDS) constituyen un mecanismo esencial para identificar comportamientos anómalos en redes y dispositivos médicos. En particular, los *Behavior-based Intrusion Detection Systems* (BIDS) permiten detectar ataques avanzados y *zero-day* analizando patrones de comportamiento, superando las limitaciones de los sistemas basados únicamente en firmas. A pesar del creciente interés en esta área, el conocimiento existente continua fragmentado: muchos estudios se centran solo en entornos IoT genéricos, sin considerar el impacto directo en la seguridad del paciente o carecen de una evaluación sistemática de las técnicas de detección y prevención de amenazas.

En este contexto, se ha llevado a cabo una Revisión Sistemática de la Literatura (SLR, *Systematic Literature Review*) que ha analizado 81 estudios recientes, con los siguientes objetivos, alineados con las preguntas de investigación (RQ): (1) identificar los BIDS actualmente empleados en entornos IoMT, (2) examinar cómo los ataques detectados pueden afectar la seguridad del paciente y (3) evaluar si los sistemas revisados integran medidas de prevención o mitigación más allá de la simple detección. Al explorar el estado actual de la investigación, identificamos lagunas y proponemos futuras líneas de investigación en este ámbito. En concreto, el SLR aporta las siguientes contribuciones:

- Presenta sistemáticamente el estado actual de la investigación sobre sistemas de detección de intrusiones en entornos IoMT.
- Analiza las medidas de prevención y mitigación comúnmente empleadas.
- Propone líneas futuras de investigación para orientar próximos estudios sobre la detección y prevención de ataques en entornos IoMT.

El presente escrito resume los resultados ya publicados en el artículo [1].

II. Estado del Arte

Durante los últimos años, se han realizado esfuerzos significativos en la investigación de los *Behavior-based Intrusion Detection Systems* (BIDS) dentro del ecosistema IoMT con el objetivo de proteger los entornos sanitarios de las amenazas de ciberseguridad. La presente revisión sistemática además compara las revisiones más destacadas de los últimos cinco años, subrayando así la singularidad del enfoque (Tabla I). Las revisiones anteriores han consolidado el conocimiento sobre sistemas IDS en esta área, ofreciendo perspectivas sobre las fortalezas y debilidades de diferentes metodologías. Algunas a destacar son:

a) Revisiones de seguridad general en entornos IoMT. Mahmood et al. [2] proporcionaron una visión general de los desafíos y soluciones de seguridad en IoMT, destacando la necesidad de protocolos y arquitecturas robustas. Sin embargo, su estudio no cubre los avances recientes en sistemas basados en el comportamiento (BIDS) ni el impacto crítico de las amenazas en la seguridad clínica del paciente.

b) Revisiones centradas en IDS y ML/DL. Rbah et al. [3] revisan el uso de técnicas de Machine Learning (ML) y Deep Learning (DL) en IDS para IoMT, reconociendo su necesidad debido a las restricciones de recursos de estos dispositivos. Clasificaron los IDS por métodos de detección (basados en firmas, anomalías y especificaciones) y destacaron los desafíos, como el alto consumo de energía y la falta de bases de datos con ataques avanzados para entrenar adecuadamente los modelos. Si-Ahmed et al. [4] también examinan cómo los IDS integrados con ML mejoran la seguridad de IoMT. Finalmente, Wazid et al. [5] se centran en los desafíos de seguridad planteados por ataques de malware.

c) Revisiones enfocadas en el impacto en la seguridad del paciente. Otras investigaciones, como las de Aijaz et al. [6] y Kamir y Sarra [7], se enfocan en el impacto de los ciberataques en la seguridad del paciente, ofreciendo taxonomías de ataques y sus efectos. A pesar de enfatizar las amenazas en la salud del paciente, estos trabajos no cubren otros temas relevantes, como los dispositivos específicos involucrados en los ataques.

d) Revisiones Sistemáticas (SLR) anteriores. Solo tres SLR en los últimos cinco años se enfocan en IDS en entornos IoMT. Hernandez-Jaimes et al. [8] revisan la detección de anomalías usando ML, y Weber et al. [9] revisan la detección de ataques en sistemas ciberfísicos médicos (*medical cyber-physical sistems*), cubriendo enfoques, tipos de datos y medidas preventivas. Finalmente, Messinis et al. [10] se enfocan en el papel de la IA y tecnologías como blockchain.

A pesar de ser revisiones exhaustivas, estas no se enfocan en BIDS ni proporcionan conclusiones detalladas sobre las técnicas específicas de detección utilizadas, ni tampoco cómo los ataques detectados afectan la seguridad del paciente, lo cual es una preocupación crítica en los entornos médicos. El presente trabajo se distingue por su enfoque exhaustivo en los BIDS para IoMT, abordando específicamente la necesidad crítica de garantizar la seguridad del paciente en todos los aspectos.

Tabla I
Comparación de revisiones previas.

Ref. (Año)	Revisión sistemática	Temas tratados en la literatura						
		1	*2*	*3*	*4*	*5*	*6*	*7*
[5] (2019)	-	✓	-	-	✓	-	-	-
[6] (2021)	-	-	-	-	✓	✓	-	✓
[3] (2022)	-	✓	-	-	✓	-	-	-
[4] (2023)	-	✓	-	-	✓	-	-	-
[8] (2023)	✓	-	✓	-	✓	-	✓	-
[7] (2023)	-	-	✓	-	✓	✓	-	-
[11] (2023)	-	-	✓	✓	✓	-	-	✓
[9] (2023)	✓	✓	-	✓	✓	-	✓	✓
[10] (2024)	✓	-	✓	-	✓	-	✓	-
[12] (2024)	-	-	✓	✓	✓	-	✓	-
Our review	✓	✓	✓	✓	✓	✓	✓	✓

1: Métodos de detección; 2: Técnicas de detección; 3: Tipos de datos analizados; 4: Análisis de los ataques; 5: Impacto en la seguridad del paciente; 6: Tipos de dispositivos; 7: Medidas de prevención y mitigación.

III. Metodología

Esta Revisión Sistemática de la Literatura (SLR) se ha llevado a cabo siguiendo las directrices para SLRs en Ingeniería del Software proporcionadas por Kitchenham y Charters [13]. De este modo, se siguieron los siguientes pasos:

1. Identificación de Estudios. Se definieron dos grupos principales de palabras clave: uno para BIDS (ej. *anomaly detection*, *abnormal behavior*) y otro para IoMT (ej. *internet of medical things*, *medical cyber–physical systems*). Se utilizaron seis bases de datos académicas relevantes: Scopus, ACM Digital Library, IEEE Xplore, Web of Science, PubMed y Science Direct. La búsqueda inicial identificó 782 publicaciones.
2. Criterios de Selección. Para garantizar la relevancia y la actualidad, se aplicaron criterios de inclusión y exclusión. Se excluyeron los estudios publicados antes del 1 de enero de 2019. Además, otro criterio de exclusión importante fue la omisión de artículos cortos (6 páginas o menos), para asegurar que el análisis se centrara en investigaciones con suficiente profundidad técnica.
3. Síntesis de Datos. Tras un proceso de eliminación de duplicados, revisión de títulos/resúmenes y una evaluación de calidad (donde se excluyeron 5 artículos por baja calidad), se seleccionaron finalmente un total de 81 artículos para la síntesis cualitativa. Scopus fue la base de datos más significativa, albergando 68 de los estudios incluidos.

IV. Resultados

IV-A. RQ1: BIDS actuales en atención sanitaria

- Métodos de detección: Dentro de los BIDS, predominan los sistemas basados en detección de anomalías (*anomaly-based*), siendo 75 de los 81 estudios revisados

(aproximadamente el 92 %), tal y como se ilustra en la Fig. 1. Estos son efectivos para detectar ataques *zero-day*, aunque a menudo presentan altas tasas de falsos positivos. En contraste, la detección basada en especificaciones (*specification-based*) fue menos prevalente (alrededor del 6 %), aunque ofrece tasas de falsos positivos más bajas, lo cual es crítico en entornos sanitarios. En algunos estudios concretos se propone un enfoque híbrido de múltiples capas.

- Técnicas empleadas: Existe una fuerte predilección por los enfoques de Machine Learning (ML), utilizados en 68 estudios. Las técnicas de ML y Deep Learning (DL), como SVM, RF, DT y DNN, son fundamentales para analizar grandes volúmenes de datos e identificar patrones de comportamiento anormales. No obstante, el rendimiento de estos modelos está estrechamente ligado a la calidad del conjunto de datos utilizado, siendo comunes bases de datos generales IoT (como TON_IoT y NSL-KDD), aunque también se emplean algunos específicos IoMT (WUSTL-EHMS-2020 y ECU-IoHT). Otras técnicas consideradas incluyen métodos estadísticos (7 estudios), lenguaje de descripción, sistemas expertos y máquinas de estado finito.
- Tipos de datos analizados: Los tipos de datos analizados son diversos, siendo el flujo de red el más común para detectar ciberataques. Sin embargo, la robustez del IDS se mejora al incorporar otros tipos de datos, como datos de paquetes, biometría del paciente y registros de dispositivos (logs).

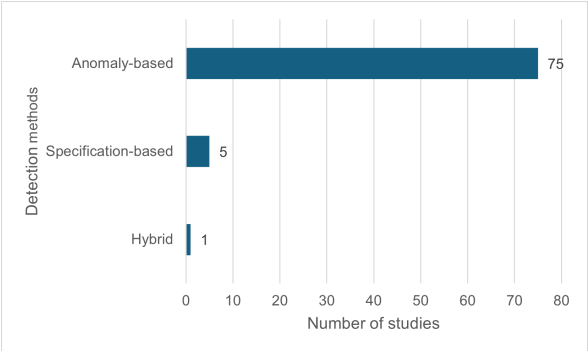


Figura 1. Métodos de detección.

IV-B. RQ2: Impacto en la seguridad del paciente

- Ubicación de los ataques (capas): Los ataques pueden ocurrir en las cuatro capas de la arquitectura IoMT: percepción, red, transporte y aplicación. La mayoría de los estudios priorizan la detección de ataques en la capa de red. Sin embargo, otras capas, como la capa de percepción (que incluye dispositivos médicos y sensores), son puntos críticos de vulnerabilidad que también se abordan.
- Superficies de ataque: La mayoría de los estudios (74 %) se concentran en las superficies de ataque secundarias, abordando vulnerabilidades amplias de la infraestructura de red IoMT. Solo el 26 % se centra en las superficies de ataque primarias, que son los dispositivos médicos que interactúan directamente con el paciente y cuya alteración pone en peligro de forma directa su salud (ej. bombas de insulina, marcapasos).

IV-B1. Tipos de ataques y dispositivos:

- Ataques: Tal y como se observa en la Fig. 2, los ataques más frecuentemente estudiados son los de Denegación de Servicio (DoS) y DoS Distribuido (DDoS) (24.7 %), cuyo objetivo es abrumar los recursos de la red y comprometer la seguridad del paciente. Le siguen los ataques Man-in-the-Middle, spoofing e inyección de datos (24.4 %). Otros ataques incluyen explotación y acceso no autorizado (22.7 %) y malware (7.5 %).
- Dispositivos: Una porción sustancial de estudios (34 %) se enfoca en dispositivos específicos de IoMT, como pulsioxímetros (SpO2) y electrocardiogramas (ECG). Sin embargo, la mayoría de los dispositivos utilizados en la literatura (49 % del total) son sensores ambientales y dispositivos IoT generales (como estaciones meteorológicas o termostatos), lo que amplía la superficie de ataque. Los dispositivos de TI (17 %), como smartphones y servidores, también se utilizan parcialmente.

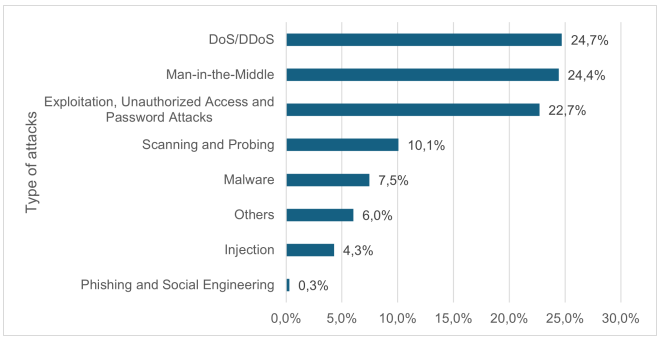


Figura 2. Ataques más utilizados en la literatura.

IV-C. RQ3: Medidas de prevención

El 41 % de los estudios revisados combinan detección con medidas de prevención. Esta proporción evidencia que la mayoría de los trabajos se enfocan exclusivamente en la detección. Las metodologías de prevención implementadas varían en complejidad:

a) Alertas y Notificaciones. El método más común es la generación de alertas para notificar al personal clínico o de TI cuando se detecta una anomalía.
b) Mecanismos Avanzados. Incluyen el bloqueo inmediato de conexiones desde nodos identificados como maliciosos. En enfoques híbridos, se utilizan sistemas de filtrado de tráfico para bloquear actividades sospechosas o la orquestación de red (como SDN) para aislar y reemplazar dispositivos infectados.

V. Discusión

El análisis de los 81 artículos revisados identifica avances significativos en el campo de los BIDS para IoMT, pero también destaca importantes desafíos que requieren atención urgente. De ese modo, cuatro brechas principales de investigación fueron identificadas, las cuales se ilustran en la Fig. 3 y se describen en detalle a continuación:

1. Uso Extensivo de ML con Limitaciones. Existe una gran dependencia de las técnicas de ML, que son

efectivas para detectar patrones complejos, pero requieren recursos computacionales significativos y grandes conjuntos de datos de entrenamiento. El desafío reside en desarrollar sistemas ligeros con alta precisión para entornos IoMT con recursos limitados.

2. Foco en Dispositivos IoT Generales. Muchos estudios se basan en conjuntos de datos y dispositivos IoT genéricos, en lugar de enfocarse en las características y vulnerabilidades particulares de los dispositivos médicos. La escasez de conjuntos de datos específicos de IoMT y soluciones personalizadas para este contexto es una brecha crítica.
3. Priorización de la Protección de Datos sobre la Seguridad del Paciente. La mayoría de los trabajos se centran en proteger los datos de salud (superficie secundaria), en lugar de salvaguardar directamente la salud y la seguridad clínica del paciente mediante la protección de dispositivos críticos (superficie primaria, como bombas de insulina). Es esencial desarrollar soluciones centradas en el paciente.
4. Falta de Estrategias de Prevención y Mitigación Robustas. La mayoría de los estudios se limitan a la detección y a la generación de alertas, sin incorporar medidas integrales y proactivas para prevenir o mitigar la propagación de los ataques. La complejidad de implementar medidas preventivas sin interrumpir el funcionamiento de dispositivos críticos para la vida del paciente es una dificultad clave.

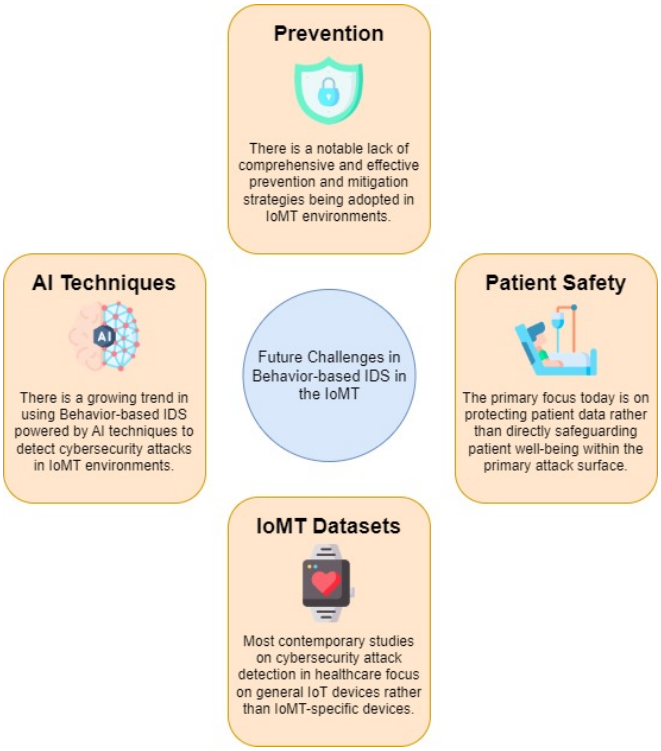


Figura 3. Futuras líneas de investigación.

VI. Conclusiones y líneas futuras

La revisión concluye que, si bien la confianza en las técnicas de ML es creciente, existen retos persistentes, como la necesidad de bases de datos específicas de dispositivos IoMT que capturen la complejidad del tráfico médico y las amenazas de ciberseguridad avanzadas. Las investigaciones futuras deben enfocarse en:

- Desarrollar soluciones de ciberseguridad centradas en el paciente.
- Explorar soluciones de detección *cross-layer* que combinen datos de red y biometría para una defensa integral.
- Diseñar estrategias robustas de prevención y mitigación capaces de generar una respuesta automática a las alertas del IDS, priorizando la salud del paciente.
- Crear una taxonomía detallada que clasifique cómo las amenazas de ciberseguridad impactan directamente en la salud del paciente, proporcionando un marco estructurado para entender los riesgos.

Este trabajo sienta las bases para futuras investigaciones dirigidas a fortalecer la postura de seguridad del sector sanitario y asegurar entornos médicos conectados más resilientes.

Agradecimientos

Esta publicación y sus resultados de investigación cuentan con el apoyo del programa predoctoral AGAUR-FI Ajuts Joan Oró (2024 FI-1 00643) Ajuts Joan Oró (Generalitat de Catalunya) y de la Cátedra de Ciberseguridad CARISMATICA financiada por la UE e INCIBE.

Referencias

[1] J. Doménech, I. V. Martin-Faus, S. Mhiri, and J. Pegueroles, "Ensuring patient safety in iomt: A systematic literature review of behavior-based intrusion detection systems," *Internet of Things*, vol. 28, p. 101420, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660524003615

[2] M. Mahmood, M. I. Khan, Ziauddin, H. Hussain, I. Khan, S. Rahman, M. Shabir, and B. Niazi, "Improving Security Architecture of Internet of Medical Things: A Systematic Literature Review," *IEEE Access*, vol. 11, pp. 107 725–107 753, 2023.

[3] Y. Rbah, M. Mahfoudi, Y. Balboul, M. Fattah, S. Mazer, M. Elbekkali, and B. Bernoussi, "Machine learning and deep learning methods for intrusion detection systems in IoMT: A survey." Institute of Electrical and Electronics Engineers Inc., 2022.

[4] A. Si-Ahmed, M. A. Al-Garadi, and N. Boustia, "Survey of machine learning based intrusion detection methods for internet of medical things," 6 2023.

[5] M. Wazid, A. K. Das, J. J. Rodrigues, S. Shetty, and Y. Park, "IoMT malware detection approaches: Analysis and research challenges," *IEEE Access*, vol. 7, pp. 182 459–182 476, 2019.

[6] M. Aijaz, M. Nazir, and M. N. Anwar, "Classification of security attacks in healthcare and associated cyber-harms." Institute of Electrical and Electronics Engineers Inc., 2021, pp. 166–173.

[7] K. Kamir and C. Sarra, "Machine learning solutions for securing iot-based healthcare: A review." Institute of Electrical and Electronics Engineers Inc., 2023.

[8] M. L. Hernandez-Jaimes, A. Martinez-Cruz, K. A. Ramírez-Gutiérrez, and C. Feregrino-Uribe, "Artificial intelligence for IoMT security: A review of intrusion detection systems, attacks, datasets and cloud–fog–edge architectures," 10 2023.

[9] S. B. Weber, S. Stein, M. Pilgermann, and T. Schrader, "Attack detection for medical cyber-physical systems-a systematic literature review," *IEEE Access*, vol. 11, pp. 41 796–41 815, 2023.

[10] S. Messinis, N. Temenos, N. E. Protonotarios, I. Rallis, D. Kalogeras, and N. Doulamis, "Enhancing internet of medical things security with artificial intelligence: A comprehensive review," 3 2024.

[11] M. A. Khatun, S. F. Memon, C. Eising, and L. L. Dhirani, "Machine learning for healthcare-iot security: A review and risk mitigation," *IEEE Access*, vol. 11, pp. 145 869–145 896, 2023.

[12] E. C. P. Neto, S. Dadkhah, S. Sadeghi, H. Molyneaux, and A. A. Ghorbani, "A review of machine learning (ml)-based iot security in healthcare: A dataset perspective," pp. 61–77, 1 2024.

[13] B. A. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," Keele University and Durham University Joint Report, Tech. Rep. EBSE 2007-001, 2007.

Further Development of the Toolbox for the Design of Non-Binary Pseudorandom Sequence Generators

Slobodan Petrović
Norwegian University of Science and Technology
P.O. Box 191, N-2802 Gjøvik, Norway
slobodan.petrovic@ntnu.no

***Abstract*—A software toolbox for the design of pseudorandom sequence generators contains various components. As a minimum, it includes a source of pseudorandomness, a source of non-linearity, and a tool for testing local randomness. Several systems of this kind exist for the development of pseudorandom sequence generators in the binary environment, but none that could help in the design of such generators in non-binary environments. This paper describes further work on the development of a toolbox for the design of pseudorandom sequence generators intended for application in non-binary (base 3, 5, etc.) symmetric cipher systems (stream and block ciphers). The toolbox includes a generator of primitive polynomials in a Galois field with the characteristic $p > 2$, an Algebraic Normal Form (ANF) calculator of non-binary switching functions, an analyzer of spectral properties of such functions, and a generator of non-binary switching functions based on Latin squares.**

***Index Terms*—Cryptographic engineering, Non-binary logic, Symmetric ciphers**

I. Introduction

Most electronic devices are nowadays realized in binary technology due to its implementation simplicity and relatively low cost. However, this technology has its limitations summarized in a phenomenon referred to as "Moore's curse" [1]. Namely, in applications realized in binary technology, there are problems related to power delivery and dissipation as the density of transistors in a binary integrated circuit increases. There are also difficulties due to the latency and bandwidth difference between on-chip and off-chip data processing. Finally, the development of production technologies is not followed fast enough by the development of software tools needed for the design of binary integrated circuits. To overcome these problems, the industry is investing more and more in the development of non-binary integrated circuits and a new branch of electronics, so called *post-binary electronics*, has an ambition to replace the classical binary electronics in the future. A promising field of application of post-binary electronic circuits is Internet of Things (IoT), where complex schemes (in particular, security-related ones) must be implemented on devices with very limited computational capabilities. On such a platform, non-binary technologies enable realization of complex circuits with a relatively small number of active elements (typically transistors), and the problems related to Moore's curse can be solved.

There are few results regarding practical design tools used in the development of security solutions, in particular symmetric ciphers, for a non-binary environment. Even though some general tools, like Matlab [2] or Sage [3] can be used for this purpose, they are in general not capable of helping during the whole design process, from finding the convenient source of (pseudo)randomness, through introducing the adequate level of non-linearity in the system, to final analysis and testing of the generated data, including the local randomness testing. This motivated us to develop our own toolbox for the purpose of symmetric cipher system design in a non-binary environment. In [4] and [5], we gave an overview of the toolbox contents and presented some results that illustrate its application in the development of some simple symmetric ciphers. In this paper, we present additional tools included in the toolbox and illustrate their application.

The paper is organized as follows: in Section II, we enumerate and explain the basic elements of our toolbox. In Section III, we present some results of application of the toolbox in the design of stream ciphers. Section IV concludes the paper.

II. The toolbox contents

The toolbox for the design of symmetric ciphers in a non-binary environment is planned to contain a module for the design of stream ciphers and a module for the design of block ciphers. The stream cipher design module contains a component for the design of a source of pseudorandom sequences and a component for introducing additional non-linearity in the designed pseudorandom sequence generator scheme. Currently, the component for the design of a source of pseudorandomness contains a set of tools for the definition and the simulation of a Linear Feedback Shift Register (LFSR) in a field of the characteristic p, where $p \in \{2, 3, 5\}$. For introducing (additional) non-linearity in the system, we include the tool for selecting an adequate non-linear switching function in the Galois field of the chosen characteristic, either as a non-linear filter or a non-linear combiner of output sequences of several LFSRs.

The principal component of the block cipher design module, which is planned to be included in the toolbox in the future, is the S-box generator and analyzer. It will generate vectorial switching functions in Galois fields of the characteristic p and test their properties by means of spectral analysis, using the Fourier transform.

A. Generating and simulating LFSRs

The LFSR theory is well developed and most results hold in a Galois field of any characteristic. However, the simplifications of the theoretical results that are possible in a Galois field of characteristic $p = 2$ do not hold if $p > 2$ due to the following facts:

1. The sum modulo p for $p > 2$ is not an involution.

2. Factorization of integers of the form $p^n - 1$ is needed for all n since $p^n - 1$ always has a factor 2 if $p > 2$. Thus, to obtain a primitive characteristic polynomial of an LFSR in a non-binary field, we need a factorization of an integer of the form $p^n - 1$, for any LFSR length n, in order to check the conditions of the Alanen-Knuth-Herlestam theorem, see [6][7], using which we determine whether the given polynomial in GF(p) is primitive.

Our toolbox checks the conditions of the Alanen-Knuth-Herlestam theorem for a randomly generated polynomial with the coefficients modulo $p \in \{2, 3, 5\}$ for LFSR lengths $n \leq 257$. The factorizations of the integers of the form $p^n - 1$ needed for this purpose have been collected from various sources, see [4].

In our toolbox, it is possible to generate primitive polynomials in a Galois field with the number of non-zero terms given in advance, which is not possible with other similar tools. This feature is very important for practical applications since in the implementations an LFSR is run by moving the feedback positions and not by moving the contents of the cells of the register. Such an implementation is usually faster than an implementation using the definition of the LFSR since the number of feedback connections is much smaller than the length of the LFSR.

The LFSR-related component of the toolbox also contains a converter from the characteristic polynomial coefficients in GF(p) to the feedback connections of the LFSR. In a binary environment, such a converter is not needed since the so-called feedback polynomial that corresponds to the feedback connections of the LFSR is reciprocal to the characteristic polynomial. For non-binary environments, we use the method of obtaining the feedback connections of the LFSR described in [8]. From the monic primitive polynomial $f(x) = x^n + c_{n-1}x^{n-1} + \cdots + c_1 x + c_0$, we obtain the corresponding linear recurrence $s_n = -c_{n-1}s_{n-1} - \cdots - c_1 s_1 - c_0 s_0$. The negative numbers are usually converted to the positive ones by adding p. In such a way, we get the feedback connections of the LFSR.

Example: Let $p = 3$ and let the characteristic polynomial of the LFSR be $f(x) = x^3 + 2x + 1$, which is primitive in GF(3). The length of the LFSR is $n = 3$ and we can write $f(x) = x^n + 2x^{n-2} + x^{n-3}$. The corresponding linear recurring relation is $s_n = -2s_{n-2} - s_{n-3}$, which we transform to a relation with positive coefficients by adding $p = 3$ to the negative coefficients. The linear recurring relation then gets the form $s_n = s_{n-2} + 2s_{n-3}$.

Indeed, if we find the characteristic polynomial $f(x) = \det(x\mathbf{I} - \mathbf{C})$ of the transition matrix $\mathbf{C}$ of this LFSR

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix},$$

we obtain $f(x) = x^3 + 2x + 1$. The LFSR is shown in Fig. 1. □

The simulation of the obtained LFSR on a binary platform is also present in the LFSR-related component of our toolbox. It moves the feedback connections to the direction opposite to the direction of moving the contents of the LFSR cells.

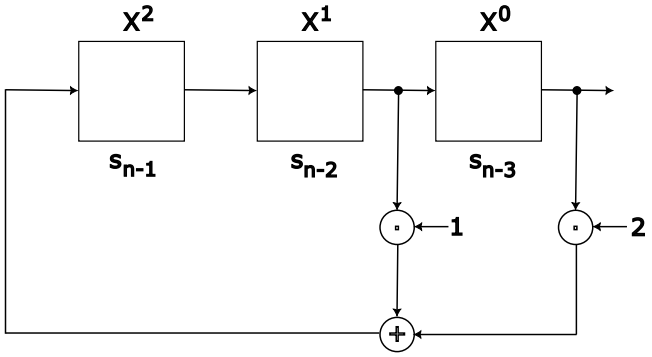


Fig. 1: The LFSR with the primitive characteristic polynomial $f(x) = x^3 + 2x + 1$ in GF(3).

B. Finding an adequate switching function

A pseudorandom sequence generator designed using our toolbox contains either several LFSRs with primitive characteristic polynomials, whose output sequences are combined in a non-linear switching function or a single LFSR, whose stage output sequences are combined in such a function, which is then called a filter function. To prevent algebraic attacks,we need a highly non-linear combining switching function that takes the LFSRs' outputs as its inputs and outputs a pseudorandom sequence, whose linear complexity is high. At the same time, the pseudorandomness (i.e., statistical) properties of the input sequences must be preserved. Therefore, the combining (or filtering) switching function must be balanced. If we use a combiner switching function, then it must be correlation-immune to prevent correlation attacks.

We assume that a switching function is given by its value table F. Given a switching function in GF(p), its balancedness is easily checked by inspecting its value table. To determine its non-linearity properties, we need

1. The Algebraic Normal Form (ANF), to determine the algebraic degree.
2. The Vilenkin-Chrestenson spectrum (sometimes also called Walsh or Walsh-Hadamard spectrum, although these terms are more often used in the binary context), to determine its distance to affine functions.

The way the ANF and the Vilenkin-Chrestenson spectrum are computed in our toolbox is described in [4] and [5]. We have also included the FFT-like fast computation algorithms. In binary, this algorithm is described in [9]. In a non-binary field, we extended the theoretical reasoning from [9] and implemented it in the toolbox.

To determine the correlation properties of the given non-binary switching function, we also need the Vilenkin-Chrestenson spectrum. Here it is worth mentioning the well-known result by Liu et al. [10] related to the correlation immunity of non-binary switching functions, which motivates their use in cryptography: it is possible to obtain a non-linear switching function of n variables, whose correlation immunity is equal to $n - 1$. An example of such a function is given in [10] for $p = 5$ and $n = 2$.

In most cases, only balanced functions are adequate to be used in cryptographic applications since the good statistical properties of the input sequences must be preserved. So-called bent functions, whose non-linearity is maximal, are

never balanced and cannot be used in pseudorandom sequence generators directly. The number of balanced functions of n variables in GF(p) is given by the following expression:

$$N_b = \prod_{i=0}^{p-2} \left(\begin{array}{c} p^n - ip^{n-1} \\ p^{n-1} \end{array} \right). \quad (1)$$

There are much more balanced switching functions in Galois fields with higher characteristics than in GF(2). For example, using the equation (1) we obtain, for $n = 4$, that there are 12870 balanced binary switching functions and 227873431500 balanced ternary functions. A relatively small fraction of all these balanced functions are non-linear and correlation-immune of certain order. Our toolbox can be used to search for such functions.

A separate component of our toolbox is related to Latin squares, which have some desirable properties when used as combiners in cryptography. They realize non-linear functions with the property that each output symbol can be correlated only with the linear combinations of input symbols that contain variables from both address branches (rows and columns). Our toolbox contains a Latin square generator in the field of characteristic p. It starts from a sum of inputs (the row address and the column address) and then shuffles the rows and columns (it is known that the Latin square properties are not lost in this process) using random permutations. Finally, the output symbol is chosen by the user and this symbol determines the concrete switching function extracted from the Latin square.

Example: A Latin square in a non-binary field generated by means of our toolbox is given below. Here, $p = 3$, $n = 2$ (which means that the Latin square is of dimension 9×9 and the switching functions realized using it have 4 input variables).

$$\begin{array}{ccccccccc} 8 & 3 & 6 & 1 & 2 & 5 & 7 & 0 & 4 \\ 3 & 7 & 1 & 5 & 6 & 0 & 2 & 4 & 8 \\ 6 & 1 & 4 & 8 & 0 & 3 & 5 & 7 & 2 \\ 1 & 5 & 8 & 3 & 4 & 7 & 0 & 2 & 6 \\ 2 & 6 & 0 & 4 & 5 & 8 & 1 & 3 & 7 \\ 0 & 4 & 7 & 2 & 3 & 6 & 8 & 1 & 5 \\ 7 & 2 & 5 & 0 & 1 & 4 & 6 & 8 & 3 \\ 4 & 8 & 2 & 6 & 7 & 1 & 3 & 5 & 0 \\ 5 & 0 & 3 & 7 & 8 & 2 & 4 & 6 & 1 \end{array}$$

If we chose the most significant symbol from the output of this Latin square, then the chosen switching function value vector is

$$\begin{aligned} F \;=\; & (2,0,0,1,2,2,1,0,1,0,1,1,2,0,0,2,1,2,0,1, \\ & 1,2,0,0,2,1,2,1,2,2,0,1,1,0,2,0,2,0,0,1, \\ & 2,2,1,0,1,0,1,1,2,0,0,2,1,2,1,2,2,0,1,1, \\ & 0,2,0,1,2,2,0,1,1,0,2,0,2,0,0,1,2,2,1,0,1). \end{aligned}$$

The input variables x_1 and x_2 address the row of the Latin square and the input variables x_3 and x_4 address its column. Our ANF computation tool gives that $f(x_1, x_2, x_3, x_4) = 2 + x_4^2 + x_3x_4 + 2x_3^2 + 2x_3^2x_4 + x_2^2 + 2x_1x_2^2 + 2x_1^2 + x_1^2x_2$. The algebraic degree of this function is $d = 3$, which makes it highly non-linear. The spectral analysis performed by means of the toolbox reports the following properties of this function:

1. The function is balanced.
2. The function is not affine.
3. The function is not bent.
4. The function is not plateaued (i.e., the number of non-zero absolute values in its spectrum is greater than 2).
5. The function is not a constant.
6. Correlation immunity of the function is 1, which means that it is correlated only with the linear combinations involving 2 or more input variables. In addition, each such linear combination contains variables from both address branches, which is very useful since this property eliminates the possibility of a divide-and-conquer correlation attack directed against only one address branch (row or column).

In addition, our tool gives all the linear combinations of the inputs correlated with the output of the given switching function. The number of such linear combinations is equal to the number of non-zero components in the Vilenkin-Chrestenson spectrum of the function. In this particular case, there are 42 such linear combinations, which are all listed by our spectral analysis tool.

□

III. Applications of the toolbox

We illustrate the application of our toolbox on the design of a small pseudorandom sequence generator, which consists of two LFSRs with primitive characteristic polynomials in GF(5), whose output sequences are combined in a non-linear switching function given in [10]. We check the non-linearity and correlation-immunity properties of this combining function.

We first choose the appropriate lengths of our LFSRs. To this end, we look at the factorizations of the integers of the form $5^n - 1$, whose list for $n \leq 257$ is included in our toolbox, and choose those containing a small number of large prime factors, apart from a small power of 2, which is always present in these factorizations. We choose $n_1 = 127$ and $n_2 = 149$ for the lengths of the LFSR1 and LFSR2, respectively. The chosen lengths n_1 and n_2 ensure long periods of the output sequences of our LFSRs ($5^{127} - 1$ and $5^{149} - 1$) and small numbers of prime factors of $5^{127} - 1$ and $5^{149} - 1$ make these LFSRs resistant to the attacks that exploit known factorizations of these numbers.

The primitive polynomial generator from our toolbox gives two primitive polynomials with coefficients in GF(5), whose numbers of feedback connections are small (in our case, 3 feedback connections for each LFSR). This ensures fast implementation by moving the feedback connections in the opposite direction of moving the contents of the LFSRs. The generated primitive polynomials are: $f_1(x) = x^{127} + x^{69} + 3x^{20} + 3$ and $f_2(x) = x^{149} + 2x^{109} + 2x^{81} + 3$.

After obtaining the primitive polynomials, we use the tool that gives us the feedback connections given these polynomials, since the generated polynomials are the *characteristic*

polynomials of our LFSRs. The linear recurring relations that correspond to our primitive characteristic polynomials are:

For LFSR1

$$\begin{aligned} s_n &= -s_{n-58} - 3s_{n-107} - 3s_{n-127} = \\ &= 4s_{n-58} + 2s_{n-107} + 2s_{n-127}. \end{aligned}$$

For LFSR2

$$\begin{aligned} s_n &= -2s_{n-40} - 2s_{n-68} - 3s_{n-149} = \\ &= 3s_{n-40} + 3s_{n-68} + 2s_{n-149}. \end{aligned}$$

This means that the feedback connections of the LFSR1 are 58, 107, and 127 and the feedback connections of the LFSR2 are 40, 68, and 149 counting from the left and starting with the stage 1.

As the combining switching function, we use the function from [10]. Its value vector is given below:

$$\begin{aligned} F &= (0,1,3,2,4,1,2,4,3,0,2,3 \\ &\quad 0,4,1,3,4,1,0,2,4,0,2,1,3). \end{aligned}$$

Our ANF computation tool gives us the ANF of this function: $f(x_1, x_2) = x_2^3 + x_1$ (as stated in [10]). The algebraic degree of the function is $d = 3$, which is highly non-linear, and the spectral analysis tool gives us the following properties of the function:

1. The function is balanced.
2. The function is not affine.
3. The function is not bent.
4. The function is not plateaued.
5. The function is not a constant.
6. Correlation immunity of the function is 1.

From these results, we see that the function has the maximum possible correlation-immunity $(n-1)$, in spite of its high algebraic degree, which cannot be achieved in a binary or a ternary environment. As such, this simple combiner is adequate for ensuring resistance to correlation attacks against the generator that we are designing. Our tool also gives all the linear combinations of the input variables, with which the function is correlated. They are: $x_1 + x_2$, $x_1 + 2x_2$, $x_1 + 3x_2$, and $x_1 + 4x_2$.

This completes the design of our pseudorandom sequence generator that operates in GF(5). The scheme of the generator is presented in Fig. 2.

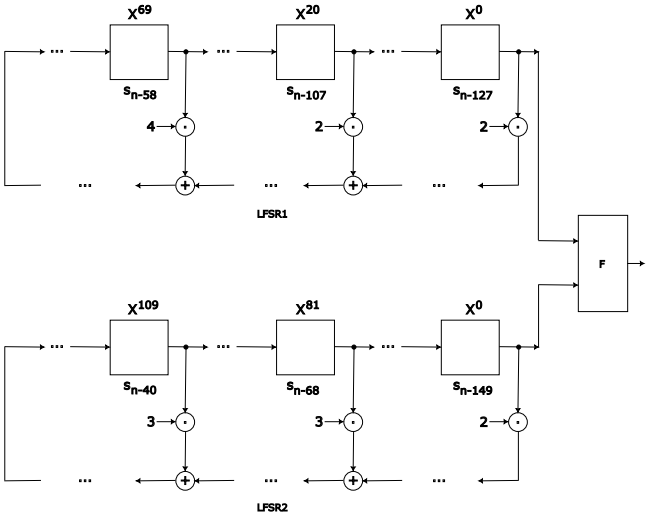


Fig. 2: A small pseudorandom sequence generator in GF(5).

IV. Conclusion

In this paper, we have presented the results of further development of the toolbox that helps in the design of symmetric ciphers in non-binary environments. This toolbox now includes the generator of primitive polynomials of the given degree and weight, the switching function analysis tool that computes the algebraic normal form of the given function and performs the spectral analysis of its characteristics based on the Vilenkin-Chrestenson spectrum, as well as a switching function generator that extracts a function from a Latin square. The Galois fields of the characteristics 2, 3, and 5 are currently supported. Further work will be focused on an S-box generator for block ciphers and sequence complexity analyzer and local randomness testing tool.

References

[1] Bos S.: "Beyond 0 and 1: A mixed radix design and verification workflow for modern ternary computers", PhD thesis, University of South-Eastern Norway, Kongsberg, May 2024.

[2] Matlab, https://www.mathworks.com

[3] Sage, https://www.sagemath.org

[4] Petrović S.: "A toolbox for the design of non-binary pseudorandom sequence generators for IoT", in *Proceedings of the XVIII Spanish Conference on Cryptology and Information Security (RECSI 2024)*, León, Spain, pp. 291–296, October 2024.

[5] Petrović S.: "Towards a framework for the design of nonlinear combiners in ternary logic", in *Proceedings of the Norwegian Information Security Conference (NISK 2024)*, Bergen, Norway, November 2024.

[6] Alanen J.D. and Knuth D.E.: "Tables of finite fields", *The Indian Journal of Statistics, Series A*, Vol. 26, No. 4, December 1964.

[7] Herlestam T.: "On using prime polynomials in crypto generators", in *Proceedings of Eurocrypt 1982*, pp. 207–216, Springer-Verlag, 1982.

[8] McAndrew A.: "Introduction to Cryptography with Open-Source Software", CRC Press, 2011.

[9] Joux A.: "Algorithmic Cryptanalysis", CRC Press, 2010.

[10] Liu M., Lu P., and Mullen G.: "Correlation-immune functions over finite fields", *IEEE Transactions on Information Theory*, Vol. 44, No. 3, May 1998, pp. 1273–1276.

NOVADEF: Una plataforma de monitorización de seguridad ubicua para actividades de perfilado de atacantes basadas en el engaño

Alberto García Pérez, Pedro Beltrán-López, José María Jorquera Valero,
Antonio López Martínez, Manuel Gil Pérez, Gregorio Martínez Pérez
CyberDataLab, Universidad de Murcia, España
{alberto.garciap, pedro.beltranl, josemaria.jorquera, antonio.lopez41, mgilperez, gregorio}@um.es

***Resumen*—La creciente expansión del ciberespacio alimenta la mayor exposición de infraestructuras críticas a nuevas amenazas persistentes y complejas. Aunque las organizaciones suelen emplear diversas herramientas de seguridad, la falta de integración entre herramientas de seguridad fragmenta la defensa y dificulta la detección y respuesta. Además, las configuraciones estáticas y la ausencia de mecanismos dinámicos limitan las habilidades para contrarrestar ataques de forma inminente. Por ello, este artículo introduce una plataforma de monitorización programable que permite desplegar y configurar a través de microservicios diversas herramientas de seguridad en base a políticas dinámicas. A través de un ciclo continuo, la plataforma pretende orquestar el despliegue, la configuración en tiempo real y automatizar contramedidas en base al perfil del atacante. Dicha plataforma hace uso de un servicio de perfilado de ataques y atacantes para identificar comportamientos maliciosos mediante técnicas de engaño. Finalmente, los experimentos muestran las capacidades de la plataforma para identificar ataques dentro del OWASP Top 10 junto con su rendimiento.**

***Index Terms*—Seguridad dinámica, ciberengaño, monitorización ubicua, perfilado de atacantes**

I. Introducción

La acelerada transformación digital ha expandido el ciberespacio, incrementando la frecuencia, sofisticación y persistencia de los ciberataques. Estos afectan la confidencialidad, integridad y continuidad de sistemas críticos como hospitales, industria y suministros nacionales [1]. Para mitigar los riesgos y ciberamenazas, las organizaciones suelen utilizar un diverso abanico de herramientas de seguridad para detectar, registrar información y responder a anomalías, p. ej., sistemas de detección de intrusos (IDS), plataformas de detección de *endpoints*, soluciones de gestión de información y eventos de seguridad (SIEM), etc. A pesar de la proliferación de tecnologías, la complejidad operativa de gestionar entornos de seguridad heterogéneos permanece como un gran reto [2]. Por lo tanto, los equipos de seguridad afrontan un panorama fragmentado que dificulta la supervisión eficaz, ralentiza los tiempos de respuesta en escenarios de amenazas dinámicos y genera una sobrecarga de datos y alertas difícil de interpretar.

Una limitación crítica de las soluciones actuales para la seguridad defensiva es la falta de integración y adaptabilidad en tiempo real. Muchos de los sistemas confían en configuraciones estáticas, las cuales son incapaces de lidiar con amenazas cambiantes [2]. Además, la naturaleza de configurar manualmente las herramientas de seguridad y políticas reduce la capacidad de reacción rápida a patrones de ataques emergentes. Esta rigidez impide el establecimiento de un ciclo continuo de retroalimentación entre la supervisión, el análisis y la respuesta, una característica esencial para garantizar la ciberresiliencia en tiempo real [3]. Para hacer frente a estas limitaciones, es necesario diseñar soluciones capaces de seleccionar, configurar y orquestar de forma autónoma los mecanismos de seguridad en función de los requerimientos contextuales del entorno y amenazas cambiantes.

En este sentido, este trabajo propone NOVADEF (pervasive mo**N**itoring platform for decepti**O**n-dri**V**en attacker profiling activities and course of **A**ction deployment in cyber **DEF**ense scenarios) una plataforma adaptativa de monitorización de la seguridad a nivel de infraestructura, servicio y red que permite identificar tanto perfiles de atacantes como tipos de ataques a través de técnicas de engaño e inteligencia artificial. Para ello, la plataforma proporciona una arquitectura de ciberseguridad dinámica a través del modelo de toma de decisiones estratégico: Observar, Orientar, Decidir y Actuar (OODA). En particular, la solución propuesta presenta un servicio de monitorización programable que permite seleccionar dinámicamente, a partir de políticas de seguridad establecidas por el usuario, tanto las herramientas que mejor se ajustan a los requisitos de seguridad como su configuración necesaria. Seguidamente, toda la información recopilada a través de múltiples fuentes de datos es normalizada y analizada para identificar patrones de ataques y construir perfiles dinámicos de atacantes que enriquecen la detección con atribución. Finalmente, la plataforma contempla la selección de estrategias de mitigación para determinar el curso de acciones a partir del perfil del atacante, el tipo de ataque y el activo afectado. Dicha decisión conllevará el reajuste automático, si es necesario, de las herramientas de monitorización, la ejecución automática de las contramedidas y el engaño activo del atacante a través de entornos controlados.

La estructura del trabajo es la siguiente. La Sección II expone una revisión de la literatura sobre monitorización de seguridad automática y técnica de engaño para el perfilado de atacantes. La Sección III presenta el flujo de pasos llevados a cabo por NOVADEF así como una arquitectura de alto nivel. Por último, la Sección IV expone un ataque del OWASP Top 10 y evalúa el rendimiento de la propuesta, mientras que la Sección V resume los principales hallazgos y vías futuras.

II. Estado del arte

Esta sección revisa trabajos recientes en dos frentes: (i) plataformas de monitorización programables y (ii) técnicas

de perfilado de atacantes para enriquecer la detección.

II-A. Plataformas de monitorización programables

La monitorización es el componente esencial para el análisis y tratamiento de datos en cualquier framework de seguridad. Diversas propuestas han abordado este reto con enfoques específicos. Calderon *et al.* [4] presentan una arquitectura distribuida basada en el stack ELK y Kafka para entornos IoT. Aunque modular, se limita a métricas de salud y red específicas de IoT, careciendo de la profundidad necesaria para análisis exhaustivos de ataques complejos. En el ámbito de infraestructuras críticas, Costa *et al.* [5] proponen LOMOS, un sistema que emplea IA y PLN para procesar eventos en el *Cloud Continuum*. Sin embargo, su enfoque se restringe a eventos de seguridad, omitiendo métricas de salud y red. Por su parte, SecConEC [6] se orienta a contenedores IoT para priorizar riesgos por impacto económico, aunque su visibilidad de red es parcial al centrarse en configuraciones y uso de recursos. Finalmente, iMonitor [7] destaca en entornos industriales (IIoT) por su baja latencia en datos de series temporales, pero ignora la interconexión de red y eventos de seguridad, vinculando las incidencias únicamente al estado del hardware. A diferencia de estos trabajos, NOVADEF integra de forma holística las tres capas de datos (salud, red y seguridad) en el *Cloud Continuum*, cubriendo desde la normalización y detección hasta la notificación y visualización de incidentes.

II-B. Perfilado de atacantes

El auge de las amenazas avanzadas impulsa enfoques capaces de caracterizar y perfilar actores maliciosos más allá de la detección tradicional mediante aprendizaje automático y grafos de conocimiento. Entre los trabajos destacados, ThreatInsight [8] propone detección y atribución temprana de APTs mediante IPs señuelo y grafos de inteligencia, aunque su dependencia de fuentes externas limita su autonomía. APT-MMF [9] introduce una fusión multimodal para la atribución de actores con alta precisión, si bien se limita a información CTI estática. Asimismo, Böge *et al.* [10] presentan un modelo híbrido Transformer–CNN para atribución basada en comportamiento, pero su dependencia de registros locales restringe su aplicación en entornos de red. Por último, Kotenko *et al.* [11] emplean redes LSTM sobre metadatos de red para predecir perfiles, aunque sin integrar una representación semántica. NOVADEF aborda estas limitaciones al unificar la captura de flujos (MP), el análisis, la inferencia semántica y el almacenamiento ontológico en Neo4j, generando perfiles dinámicos y actualizables para la futura automatización de decisiones.

II-C. Comparativa con sistemas EDR y SIEM

A diferencia de los sistemas de detección en endpoints (EDR) o SIEM convencionales, NOVADEF aporta capacidades disruptivas mediante una monitorización ubicua y programable que unifica tres capas (infraestructura, servicio y red). Mientras que los SIEM actúan frecuentemente como receptores pasivos de logs con dificultades para adaptarse a entornos dinámicos y granulares [12], la plataforma de monitorización programable de NOVADEF utiliza microservicios y reglas Sigma para reconfigurar dinámicamente qué y cómo se monitoriza según el contexto de la amenaza, superando la rigidez de las herramientas tradicionales. Además, NOVADEF integra ciberengaño activo para extraer inteligencia del adversario en tiempo real, empleando ontologías para traducir alertas técnicas en perfiles humanos que evalúan motivación y *skills*. Este enfoque de defensa activa [13] permite una respuesta adaptativa mediante el ciclo OODA que trasciende el análisis de firmas y patrones estáticos propio de los sistemas convencionales.

III. Metodología y arquitectura

La metodología de NOVADEF se apoya en el ciclo OODA como marco de defensa adaptativo para incluir capacidades de automatización, IA e inteligencia semántica a través de mecanismos de adaptación dinámica y retroalimentación. Esta sección explica tanto el flujo de etapas de NOVADEF como su arquitectura principal.

III-A. Flujo de etapas metodológicas

Como se ha comentado anteriormente, en la Figura 1 se observa el mapeo de NOVADEF al ciclo OODA: la plataforma recoge y normaliza telemetría (Observar), la enriquece y correlaciona semánticamente (Orientar), estima ataque y perfil del adversario con un motor de políticas y modelos de IA (Decidir), y orquesta contramedidas y engaños en entornos controlados (Actuar). Los resultados retroalimentan modelos y configuraciones para el siguiente ciclo. A continuación, se definen con detalle las acciones que NOVADEF realiza en cada fase del ciclo.

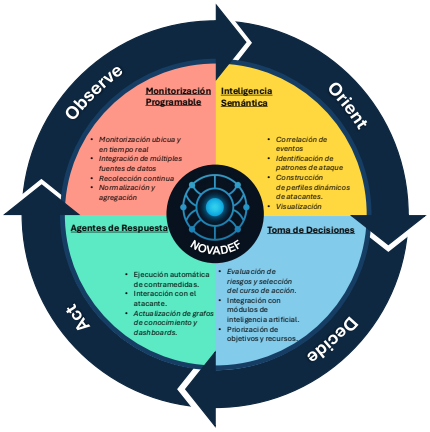


Figura 1. Modelo metodológico de NOVADEF basado en el ciclo OODA

- **Observe:** Fase de observación implementada por la Monitorización Programable (MP). Permite una monitorización ubicua y en tiempo real, integrando fuentes de datos de infraestructura, servicio y red. Los procesos de recolección, normalización y agregación ofrecen una visión coherente y automatizada del entorno, eliminando la fragmentación de la información y adaptándose a nuevos elementos a monitorizar sin reconfiguración manual.
- **Orient:** Fase de correlación e Inteligencia Semántica. Analiza los datos recolectados mediante técnicas de aprendizaje automático para identificar patrones de ataque y generar perfiles dinámicos de atacantes. Estos resultados actualizan el modelo de amenaza, materializado en el grafo semántico, e incorporan una estimación de riesgo por entidad/actividad, que más adelante se utiliza

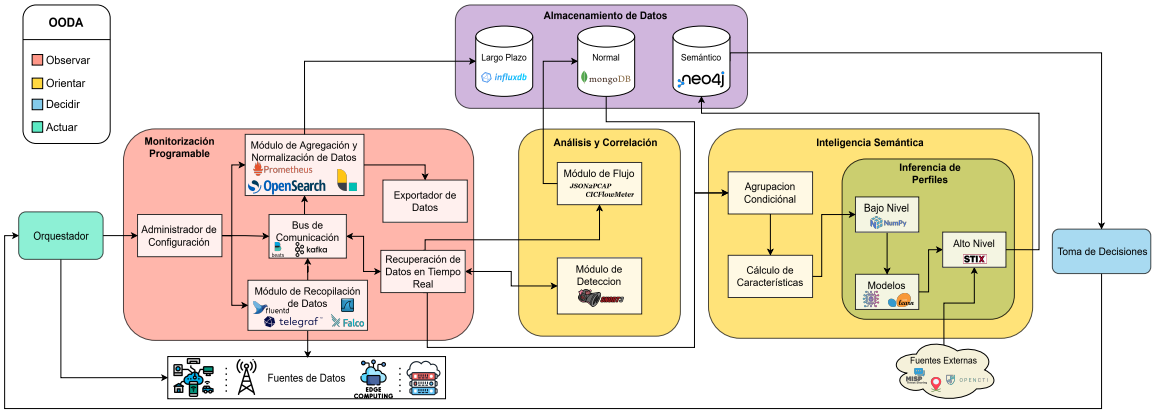


Figura 2. Arquitectura general y modular de la plataforma NOVADEF

en *Decide* para priorizar el curso de acción. Así se mejora la precisión y la adaptabilidad del sistema.

- **Decide:** Fase de evaluación automática de riesgos y selección del curso de acción (CoA) óptimo. Utiliza modelos de IA para priorizar objetivos y recursos según el impacto y la probabilidad de cada evento, reduciendo los tiempos de respuesta y los errores humanos.
- **Act:** Fase de ejecución automática de contramedidas mediante agentes de respuesta inteligentes. Estos interactúan con el atacante mediante técnicas de engaño y actualizan en tiempo real los grafos de conocimiento, cerrando el ciclo de retroalimentación y permitiendo el aprendizaje continuo del sistema.

En síntesis, la metodología basada en el ciclo OODA dota a NOVADEF de una estructura operativa, iterativa, inteligente y autónoma, en la que cada fase alimenta a la siguiente dentro de un proceso continuo de observación, orientación, decisión y actuación. Este enfoque permite transformar la monitorización tradicional en un sistema adaptativo y autoevolutivo, capaz de responder en tiempo real a entornos cambiantes y amenazas emergentes. Al integrar IA, la automatización de decisiones y ejecución autónoma de contramedidas, NOVADEF aborda las limitaciones detectadas en el estado del arte, estableciendo una metodología sólida para la defensa dinámica. De esta manera, la plataforma no solo detecta y responde, sino que aprende y se reconfigura de forma proactiva, consolidando un modelo de seguridad inteligente que encarna plenamente los principios del ciclo OODA aplicado a la ciberseguridad moderna.

III-B. Arquitectura de la plataforma

La arquitectura de la plataforma NOVADEF (ver Figura 2) se estructura en torno a un conjunto de módulos funcionales que implementan las distintas fases del ciclo OODA, descrito previamente en la metodología de la Sección III-A. Cada uno de estos componentes cumple un papel y color específico dentro del proceso de defensa adaptativa, garantizando la adquisición, procesamiento, análisis e interpretación de la información en tiempo real. Como se muestra en la Figura 2, la arquitectura integra de forma modular los bloques de Monitorización Programable, Análisis y Correlación, Inteligencia Semántica, Almacenamiento de Datos, Toma de Decisiones y Orquestación. Esta organización permite un flujo de información continuo, en el que los datos capturados son progresivamente transformados en conocimiento semántico y decisiones operativas. El diseño modular y escalable de NOVADEF facilita su adaptación a diferentes entornos y su futura ampliación con nuevas capacidades de automatización y respuesta inteligente.

*III-B1. **Monitorización Programable**:* La MP es responsable de la fase *Observe* (color rojo), realizando la monitorización continua de diferentes niveles en el espectro del Cloud Continuum. Su objetivo es el de unificar y coordinar la recolección de información para exponerla a otros componentes de análisis. Trabaja sobre tres capas de datos relacionadas con la infraestructura, los servicios y la red. Está sustentada bajo una estructura modular que permite su expansión mediante el añadido de nuevos módulos. Además de esto, todos sus módulos han sido contenerizados para un mejor despliegue dentro de un entorno totalmente controlado y de forma agnóstica al dispositivo objetivo. La configuración es dinámica, (re)programable y estandarizada gracias al uso de reglas Sigma, de esta forma se puede adaptar a las necesidades requeridas en cualquier momento.

Como se muestra en la Figura 2, la MP es gestionada por un *Orquestador* de seguridad que tiene la competencia de interpretar los requerimientos del entorno y adaptar la política de seguridad. Genera reglas Sigma para la configuración de los módulos a través del *Administrador de Configuración*, que es el encargado de configurar las herramientas internas de los módulos. El despliegue también es llevado a cabo por el *Orquestador*. Es necesario que el *Módulo de Recopilación de Datos* sea desplegado en una máquina objetivo, pudiendo elegir de forma individual qué capa de datos monitorizar y, por ende, qué tipo de datos extraer.

Una vez realizada la configuración y despliegue, el *Módulo de Recopilación de Datos* se encarga de generar un identificador único del dispositivo, recolectar métricas de salud con Telegraf (capa de infraestructura), registros de seguridad con Fluentd y Falco (capa de servicio), y finalmente, trazas de red con Tshark (capa de red). Toda la información es transmitida constantemente a través del *Bus de Comunicación* gracias a Filebeat que es el encargado de leer, comprimir y mandar a Kafka los datos previamente recolectados.

En este punto, la información puede ser consumida en bruto haciendo uso del módulo de *Recuperación de Datos en Tiempo Real*. Adicionalmente, se ha incluido el *Módulo*

de Agregación y Normalización de Datos con Prometheus, Opensearch y Logstash para preprocesar los datos en bruto y estandarizar el formato con Elastic Common Schema, dando acceso a los datos, mediante el identificador de dispositivo, a otros componentes que los necesiten a través del *Exportador de Datos*. Además, un histórico es almacenado en una base de datos de series temporales (InfluxDB) llamada *Largo Plazo*.

III-B2. ***Análisis y Correlación****:* Este módulo forma parte de la fase *Orient* (color amarillo), y transforma telemetría de red bruta en conocimiento procesable, detecta anomalías y relaciona eventos y entidades. El proceso se inicia en el *Módulo de Detección*, que identifica patrones sospechosos basados en reglas a partir de las fuentes de datos (indicadores de compromiso y rasgos temporales/estadísticos) y emite *alertas* de amenazas/ataques que se remiten al módulo de *Inteligencia Semántica* a través del módulo de *Recuperación de Datos en Tiempo Real*. La actividad de red pasa al *Módulo de Flujo*, donde se estructuran *flujos bidireccionales*: JSON2PCAP convierte registros JSON a PCAP y CICFlowMeter analiza y unifica métricas de red (entidad, volumen, frecuencia, duración y comportamiento de paquetes). Los flujos se guardan en MongoDB para su distribución. Estos datos son correlacionados por tiempo, origen/destino y tipo de amenaza, unificando incidentes. Posteriormente, el nivel de *Inteligencia Semántica* recoge dichos flujos enriquecidos de MongoDB.

III-B3. ***Inteligencia Semántica****:* Este módulo también forma parte de la fase *Orient* (color amarillo), y constituye uno de los pilares funcionales de la arquitectura NOVADEF. Su propósito principal es transformar datos procesados y normalizados de red, convirtiendo la información técnica de *Análisis y Correlación* en inteligencia útil para interpretación y decisión. Está estrechamente vinculado a tres submódulos: *Agrupación Condicional*, *Cálculo de Características* e *Inferencia de Perfiles*, que operan de forma encadenada sobre las evidencias recibidas [14].

En *Agrupación Condicional* los flujos de red se organizan por criterios como tipo de amenaza, fuente/destino, ventana temporal y relaciones con otros sucesos, generando conjuntos coherentes de contexto. El *Cálculo de Características* extrae métricas cuantitativas y cualitativas (frecuencias, distribuciones, correlaciones, indicadores de comportamiento). A partir de estas señales, *Inferencia de Perfiles* hace uso de modelos de IA (definidos en [14]) previamente entrenados para la generación de características semánticas y aplica reglas, umbrales y relaciones semánticas para inferir perfiles de actores a nivel de red, comportamientos esperables, clases de amenaza y niveles de riesgo, produciendo una representación semántica enriquecida de la situación. Cabe destacar que los detalles sobre la arquitectura de estos modelos, el proceso de entrenamiento y la validación de los atributos de perfilado (como *skills*, motivación o afiliación) se encuentran en el trabajo referenciado [14], donde se evaluaron diversos algoritmos de aprendizaje automático resultando óptimo el uso de *Random Forest* (RF) debido a la alta explicabilidad que este proporciona para la justificación de dichas características.

Los resultados se envían al *Almacenamiento de Datos* en formato estructurado/semántico para consulta posterior y correlación con nuevas evidencias, constituyendo una base de conocimiento dinámica que refuerza la detección, el análisis forense y la respuesta ante incidentes. Funcionalmente, este módulo integra inteligencia de amenazas externa (como MISP u OpenCTI) mediante el uso de estándares como STIX y protocolos TAXII. Esta integración permite que el motor de inferencia no solo dependa de la telemetría local, sino que pueda combinar patrones detectados con indicadores de compromiso (IoC) y tácticas, técnicas y procedimientos (TTPs) conocidos globalmente. De este modo, NOVADEF puede ser desplegado de forma complementaria a plataformas de *Threat Intelligence* existentes, utilizando su conocimiento semántico para validar y enriquecer la atribución de perfiles generada en el grafo de Neo4j.

III-B4. ***Almacenamiento de datos****:* Este módulo (representado en color lila) constituye la capa persistente del sistema y garantiza la conservación, estructuración y disponibilidad de la información generada a lo largo del ciclo. Al ser un elemento usado por todos los módulos, no se incluye dentro de una fase específica del OODA. Se divide en tres dominios diferenciados según la Figura 2:

- Normal: basado en MongoDB, que aloja flujos de red.
- Semántico: implementado en Neo4j, donde se consolidan las relaciones ontológicas generadas por el módulo de *Inteligencia Semántica* para mantener un grafo de conocimiento consultable que permita descubrir correlaciones, dependencias y realizar inferencias sobre entidades y eventos.
- Largo Plazo: sustentado en InfluxDB, diseñado para el almacenamiento histórico y análisis temporal de métricas y eventos.

Esta segmentación refleja la necesidad de manejar datos heterogéneos (estructurados, semiestructurados y no estructurados). El repositorio semántico permite ejecutar consultas para descubrir correlaciones y dependencias entre entidades, mientras que la capa temporal facilita análisis longitudinales y la detección de patrones persistentes. La coexistencia de bases de datos de naturaleza distinta fomenta una visión multidimensional del conocimiento, en la que los indicadores técnicos se enriquecen con metadatos semánticos y contextuales, asegurando coherencia entre la detección operativa y la inferencia táctica.

III-B5. ***Toma de Decisiones****:* Este módulo forma parte de la fase *Decide* (color azul), y constituye la capa responsable de interpretar la inteligencia generada por los módulos previos y convertirla en acciones operativas o estratégicas. Aunque aún no está implementado, su desarrollo está previsto para próximas fases siguiendo una metodología incremental, ya que depende del correcto funcionamiento y madurez de *Observe* y *Orient*. Conceptualmente, representa la etapa final del ciclo de análisis, orientada a evaluar los resultados de los módulos de *Análisis y Correlación* e *Inteligencia Semántica* para determinar la respuesta más adecuada ante una amenaza. En futuras versiones, se prevé que automatice tareas como la activación de contramedidas, la reconfiguración de políticas de seguridad o la adaptación dinámica de los entornos de monitorización y engaño.

III-B6. ***Orquestación****:* Este módulo forma parte de la fase *Act* (color verde), y actúa como núcleo de configuración y control del sistema, garantizando la coordinación bajo un

marco común. Además de centralizar parámetros, activar módulos y supervisar el estado, seleccionará y ajustará cursos de acción mediante modelos de teoría de juegos (a partir del perfil dinámico del atacante inferido en el grafo semántico) y orquestará agentes de engaño (decoys, honeytokens, servicios señuelo) para comprometer al adversario y recolectar inteligencia, actualizando en tiempo real creencias y políticas. Aunque aún no está implementado, su desarrollo está previsto para próximas fases siguiendo una metodología incremental, ya que depende del correcto funcionamiento y madurez de *Observe* y *Orient*.

IV. Evaluación experimental

Una vez descrita la arquitectura y finalidad de NOVADEF, se procede a validar su funcionamiento mediante un experimento controlado diseñado para comprobar las capacidades de monitorización, detección y análisis de la plataforma [15]. El objetivo es evaluar cómo NOVADEF ingiere trazas, caracteriza un incidente y genera inteligencia semántica y grafos de conocimiento a partir de un ataque realista de prueba.

IV-A. Configuración del experimento

El experimento se lleva a cabo sobre un testbed real y controlado, configurado para garantizar la seguridad y la reproducibilidad del escenario. El testbed cuenta con 8 núcleos, 32GB de memoria RAM y 100GB de almacenamiento, todo ello sobre un sistema operativo Ubuntu-server-24.04.03. En este entorno se han desplegado los siguientes componentes:

- NOVADEF: sistema bajo evaluación que integra los módulos de monitorización, recolección, correlación de eventos, inteligencia semántica y perfilado de atacante.
- Dispositivo objetivo: se ha desplegado un honeypot de alta interacción basado en *Snare/Tanner*, configurado para emular un servidor web con una aplicación deliberadamente vulnerable a ataques de SQL Injection. Este actúa como señuelo y fuente de eventos para NOVADEF.
- Máquina atacante: equipo desde el cual se lanza manualmente el ataque de SQL Injection siguiendo un conjunto de payloads y técnicas predefinidas del *OWASP Top 10*, generando el tráfico malicioso que NOVADEF debe detectar y analizar.

El ataque *SQL Injection* se ejecuta manualmente desde la máquina atacante contra el dispositivo objetivo, permitiendo observar en tiempo real cómo NOVADEF identifica el comportamiento anómalo, correlaciona evidencias, genera inteligencia semántica y construye el grafo de conocimiento (entidades, eventos y actores). La plataforma actúa como único sistema de observación y análisis, registrando todas las trazas y produciendo los artefactos para su evaluación. Dado el estado actual de desarrollo, la fase de decisión y actuación no se activa, con el objetivo de evaluar detección, correlación, inteligencia semántica, perfilado del atacante y construcción del grafo para demostrar el correcto funcionamiento de las fases *Observe* y *Orient*.

IV-B. Análisis y resultados

Tras llevar a cabo la ejecución del experimento descrito en la Sección IV-A (OWASP TOP 10 - SQL Injection, Brute Force y DDoS), con NOVADEF como único sistema de observación y análisis, la plataforma identificó el comportamiento anómalo multivectorial, correlacionó las evidencias y materializó el incidente como un grafo de conocimiento consultable de mayor complejidad.

El grafo resultante (ver Figura 3) sintetiza el incidente integrando múltiples hilos de ataque a través de nodos clave (IPs de origen, IPs de destino, técnicas de ataque como *Password Spraying* o *Direct Network Flood*, vectores de explotación y herramientas como *Hydra*) y las aristas *ORIGINATES_FROM*, *TARGETS*, *USES*, *USES_TOOL*, *IN_PHASE* y *PREFERS*. Estas relaciones codifican de forma granular el origen, el objetivo, la técnica específica, la fase táctica y las preferencias de los distintos actores involucrados.

La lectura del grafo permite reconstruir un hilo causal único y consistente: se identifican diversos perfiles (nodos *hacker* y *script_kiddie*) que, partiendo de una infraestructura común (10.0.0.10), ejecutan acciones simultáneas de fuerza bruta para el acceso a credenciales, explotación de vulnerabilidades y ataques de denegación de servicio (DDoS) mediante inundación de red, impactando sobre los activos de destino (10.0.0.1, 10.0.0.2 y 10.0.0.3).

Además, los nodos principales del grafo encapsulan información adicional más allá de lo visible (nivel de riesgo del atacante, motivación, actitud, afiliación, perfil, skills, conocimiento, grado de automatización, *firstSeen/lastActivity*, ámbito/país, etc.), lo que habilita una priorización y consultas dirigidas sobre el propio grafo ante una amenaza multivectorial. En el incidente analizado se han inferido perfiles diferenciados: desde un atacante caracterizado como "hacker" con *skills* bajas y uso de herramientas específicas (*Hydra*), hasta un perfil de "script_kiddie" con alta automatización y actitud destructiva; estos atributos están representados como propiedades de los nodos actor y se utilizan para priorizar respuestas, ajustar utilidades en los modelos de decisión y seleccionar agentes/acciones de engaño apropiadas para cada vector de ataque.

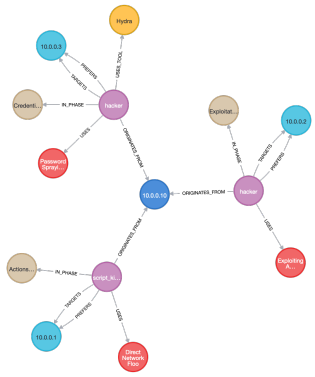


Figura 3. Grafo de conocimiento generado por NOVADEF

Los resultados recogidos en la Tabla I muestran un desempeño sólido de NOVADEF en términos de rendimiento y módulos.

En el módulo *MP*, la Cobertura de Telemetría Avanzada (CTA) alcanza valores superiores al 95 %, reflejando una alta profundidad de observabilidad. Esta métrica cuantifica el porcentaje de tipos de datos sobre las capas monitorizadas, indicando la completitud de la telemetría recolectada. La

Integridad de Payload (IPY) supera el (99,9 %), garantizando que la información capturada es completamente utilizable y libre de corrupción; mide la proporción de cargas útiles recibidas sin pérdida ni alteración. Matemáticamente, se define como $IPY = (P_{val}/P_{tot}) \times 100$, donde P_{val} representa los paquetes cuyo *checksum* y contenido coinciden íntegramente en el destino frente al total de paquetes capturados en el origen (P_{tot}), asegurando la ausencia de truncamientos.

En el módulo de *Análisis y Correlación*, tanto la Integridad de Flujos (IFL) como la Precisión de Deduplicación (PDD) mantienen niveles por encima del 99,5 %. La IFL evalúa la consistencia temporal y estructural de los flujos reconstruidos a partir de la telemetría. Se formaliza mediante el ratio $IFL = F_{comp}/F_{total}$, donde F_{comp} representa los flujos bidireccionales con correlación completa de *flags* (SYN/FIN) y números de secuencia coherentes, y F_{total} es el total de flujos detectados, validando la ausencia de huecos de información en las sesiones. Por su parte, la PDD mide la exactitud del proceso de deduplicación, evitando tanto la fusión excesiva como la fragmentación.

Finalmente, en el módulo de *Inteligencia Semántica*, la Integridad de Agrupación (IAG) asegura homogeneidad antes del proceso inferencial; mide la coherencia interna de los grupos semánticos generados. El F1-score ($\geq$ 0,85) [14] resume un excelente equilibrio entre precisión y recobrado en el mapeo de técnicas, reflejando la capacidad del sistema para identificar correctamente patrones y relaciones relevantes.

Tabla I
MÉTRICAS TEMPORALES Y DE CALIDAD POR MÓDULO EN NOVADEF

Categoría / Módulo	Métrica	Resultado
Tiempos globales		
	Ataque → Neo4j	3,632s
	Ataque → Monitorización	12ms
	Monitorización → Detección	3s
	Detección → Perfil	500ms
	Perfil → Neo4j	120ms
MP		
	Cobertura de telemetría	$\geq 95\,\%$
	Integridad de payload	$\geq 99,9\,\%$
Análisis y Correlación		
	Integridad de flujos	$\geq 99,5\,\%$
	Precisión de deduplicación	$\geq 99,5\,\%$
Inteligencia Semántica		
	Integridad de agrupación	$\geq 99,0\,\%$
	F1-score	$\geq 0,85$

En cuanto al comportamiento temporal, también reflejado en la Tabla I, el tiempo total desde "Ataque" hasta "Neo4j" se explica principalmente por el diseño de procesamiento entre Monitorización y Detección. La transferencia de información entre los módulos internos y el procesamiento de la herramienta de detección de alertas que trabaja en streaming y la herramienta de correlación de flujos generan una pequeña demora. Se ha mantenido una política de *mini-batching* en los flujos para establecer el inicio y fin de un flujo de red al completar PCAPs con 50KB o un rango de tiempo de medio segundo. Aun así, los tiempos parciales restantes muestran una transmisión y análisis altamente eficientes, con valores del orden de milisegundos entre módulos consecutivos.

V. CONCLUSIONES Y VÍAS FUTURAS

Este trabajo muestra el diseño y desarrollo inicial de una plataforma de ciberseguridad dinámica encargada de monitorizar, analizar, decidir y reaccionar en tiempo real a través de técnicas de engaño. Dicha plataforma observa comportamientos inusuales a nivel de infraestructura, servicio y red para crear perfiles de atacantes y patrones de ataque. Los resultados obtenidos demuestran que NOVADEF es capaz de monitorizar, identificar y perfilar atacantes en 4 segundos con un F1-score $\geq 0,85$ para ataques en el OWASP Top 10.

Como siguientes pasos, NOVADEF extenderá el espectro de ataques a ser perfilados con su correspondiente técnica(s) de engaño y las fuentes de datos consideradas. Además, la plataforma incluirá nuevas funcionalidades reflejadas en el diseño como la toma de decisiones basada en CoAs, el perfil del atacante y su tipo de ataque, así como la inclusión de un orquestador para automatizar el despliegue de los microservicios, su reconfiguración y la aplicación de contramedidas.

AGRADECIMIENTOS

Este trabajo ha sido financiado por la Comisión Europea a través del proyecto ROBUST-6G (Acuerdo de subvención n.º 101139068) del programa Horizonte Europa/JU SNS.

REFERENCIAS

[1] M. Lehto, "Cyber-attacks against critical infrastructure," in Computational Methods in Applied Sciences, 2022, pp. 3–42.
[2] A. Dehghantanha, A. Yazdinejad, and R. M. Parizi, "Autonomous cybersecurity: Evolving challenges, emerging opportunities, and future research trajectories," in Workshop on Autonomous Cybersecurity, 2024, pp. 1–10.
[3] R. Ross, V. Pillitteri, R. Graubart, D. Bodeau, and R. McQuaid, "NIST Special Publication 800-160, Volume 2: Developing Cyber-Resilient Systems: A Systems Security Engineering Approach," Tech. Rep., 2022.
[4] G. Calderon, G. del Campo, E. Saavedra, and A. Santamaría, "Monitoring framework for the performance evaluation of an IoT platform with Elasticsearch and Apache Kafka," Information Systems Frontiers, vol. 26, pp. 2373–2389, 2024.
[5] J. P. Costa *et al.*, "LOMOS: An AI-based runtime security monitoring system fit for the Cloud Continuum," in European Conference on Parallel Processing, 2024, pp. 5–14.
[6] Q. Deng, M. Goudarzi, A. Shaghaghi, M. Sarvi, and R. Buyya, "A secure framework for containerized IoT applications in integrated edge-cloud computing environments," Future Generation Computer Systems, vol. 174, p. 108010, 2026.
[7] Z. Liang, L. Jia, B. Zheng, and H. Wang, "iMonitor: A real-time monitoring platform for industrial internet of things," in International Conference on Database Systems for Advanced Applications, 2024, pp. 462–465.
[8] Z. Wang *et al.*, "ThreatInsight: Innovating early threat detection through threat-intelligence-driven analysis and attribution," IEEE Transactions on Knowledge and Data Engineering, vol. 36, no. 12, pp. 9388–9402, 2024.
[9] N. Xiao, B. Lang, T. Wang, and Y. Chen, "APT-MMF: An advanced persistent threat actor attribution method based on multimodal and multilevel feature fusion," Computers & Security, vol. 144, p. 103960, 2024.
[10] E. Böge, M. B. Ertan, H. Alptekin, and O. Çetin, "Unveiling cyber threat actors: A hybrid deep learning approach for behavior-based attribution," Digital Threats: Research and Practice, vol. 6, no. 1, pp. 1–20, 2025.
[11] I. Kotenko, E. Fedorchenko, E. Novikova, and A. Jha, "Cyber attacker profiling for risk analysis based on machine learning," Sensors, vol. 23, no. 4, p. 2028, 2023.
[12] S. Bhatt, S. K. Raghuwanshi, and A. Bhardwaj, "Siem for cloud security," International Journal of Computer Applications, vol. 101, no. 1, 2014.
[13] D. Fraunholz, S. D. Anton, C. Lipps, D. Reti, and H. D. Schotten, "Adaptive cyber deception: Targets, strategies and challenges," International Journal of Cyber Warfare and Terrorism, vol. 8, no. 2, pp. 1–15, 2018.
[14] P. Beltran, M. Gil Pérez, and P. Nespoli, "Antifragile cyber deception: Profiling threat actors through adaptive intelligence and semantic inference," Available at SSRN 5598038, 2025.
[15] CyberDataLab, "NOVADEF," https://github.com/CyberDataLab/NOVADEF, 2025.

Generadores Pseudoaleatorios sobre $\mathbb{F}_{p^n}$ con Realimentación Dinámica Inducida por Curvas Elípticas

Guillermo Cotrina Cuenca
Universidad Complutense de Madrid
Campus de Somosaguas
28223 Pozuelo de Alarcón MADRID
guicotri@ucm.es

Alberto Peinado
Instituto de investigación en Telecomunicación (TELMA)
Universidad de Malaga
Bulevar Louis Pasteur, 35 29071 MÁLAGA
apd@uma.es

***Resumen*—En este trabajo se propone un generador pseudoaleatorio basado en registros de desplazamiento con realimentación lineal definidos sobre cuerpos finitos extendidos $\mathbb{F}_{p^n}$, en los que la realimentación se deriva de curvas elípticas $E(\mathbb{F}_{p^n})$ a partir de puntos consecutivos kG y $(k+1)G$. Para ello, se concatenan las coordenadas de los puntos y se extrae una ventana central de coeficientes base-p, reinterpretada como un elemento del cuerpo. El flujo de la secuencia final se obtiene mediante una binarización por traza mejorada. Los resultados experimentales muestran equilibrio estadístico, baja autocorrelación y elevada complejidad lineal, indicando que la combinación entre LFSR y parcheo elíptico ofrece secuencias aptas para aplicaciones criptográficas.**

***Index Terms*—generadores pseudoaleatorios, LFSR sobre cuerpos finitos, curvas elípticas sobre $\mathbb{F}_{p^n}$.**

I. Introducción

La generación de secuencias pseudoaleatorias de alta calidad es esencial en criptografía aplicada, comunicaciones seguras y sistemas embebidos. Los *registros de desplazamiento con realimentación lineal* (LFSR) son especialmente atractivos por su eficiencia y sencillez de implementación; sin embargo, su naturaleza lineal los hace vulnerables a ataques algebraicos cuando se observa una cantidad suficiente de salida, como ilustra el algoritmo de Berlekamp–Massey. Esto motiva la necesidad de incorporar mecanismos de no linealidad que aumenten la resistencia criptográfica sin comprometer el rendimiento.

Los generadores basados en *curvas elípticas* (EC-PRNG) han mostrado un comportamiento estadístico competitivo debido a la complejidad asociada al logaritmo discreto elíptico. Diversas propuestas emplean secuencias derivadas de puntos sucesivos kG [2], [1], [6], obteniendo salidas con baja correlación y adecuadas para aplicaciones ligeras [5], [8]. No obstante, también se conocen debilidades relevantes en ciertos diseños elípticos: por ejemplo, algunos modelos de tipo congruencial definidos sobre curvas han sido atacados mediante técnicas de retículas [3], y existen resultados que muestran cómo inferir estados o secuencias generadas por determinados esquemas a partir de filtraciones parciales [4]. Asimismo, el caso Dual EC-DRBG constituye un ejemplo paradigmático de vulnerabilidad estructural cuando existen relaciones internas no auditables entre parámetros públicos y el estado [7]. En conjunto, estas observaciones subrayan que, aun cuando la aritmética elíptica aporta complejidad algebraica, el diseño del extractor y la forma de acoplar el estado interno son determinantes para evitar predictibilidad y filtraciones.

En este trabajo proponemos un enfoque **híbrido** que combina la estructura bien conocida de un LFSR sobre $\mathbb{F}_{p^n}$ con una fuente adicional de no linealidad obtenida a partir de puntos consecutivos sobre una curva elíptica definida en el mismo cuerpo. La información extraída de las coordenadas de kG y $(k+1)G$ se condensa mediante una *ventana central* de coeficientes base-p, que actúa como perturbación dinámica del polinomio de conexión del LFSR. De esta forma, se mantiene la eficiencia del núcleo lineal, pero se rompe la linealidad global de la evolución, dificultando ataques basados en reconstrucción algebraica bajo el supuesto de una conexión estacionaria.

A partir de la secuencia sobre el cuerpo se obtiene un flujo binario mediante una *binarización por traza mejorada*, diseñada para reducir sesgos y reforzar la mezcla local. Los resultados experimentales muestran equilibrio estadístico, baja autocorrelación y complejidades lineales elevadas en las proyecciones observadas, lo que indica que el acoplamiento entre LFSR y realimentación elíptica constituye una estrategia prometedora para la generación de secuencias pseudoaleatorias, a falta de una evaluación estadística y criptanalítica más amplia.

II. Construcción del Modelo Generador

El generador propuesto se construye como una arquitectura híbrida que integra, de manera controlada, dos componentes algebraicos de naturaleza complementaria:

1. un LFSR definido sobre un cuerpo finito $\mathbb{F}_{p^n}$, y
2. un mecanismo de perturbación no lineal derivado de puntos consecutivos de una curva elíptica $E(\mathbb{F}_{p^n})$.

La interacción entre ambos componentes modifica dinámicamente la evolución del LFSR, rompiendo su linealidad global y dando lugar a una secuencia cuya estructura ya no está gobernada por una única recurrencia lineal fija sobre $\mathbb{F}_{p^n}$.

El diseño se apoya en la siguiente observación fundamental: los LFSR presentan una excelente eficiencia computacional y simplicidad de implementación, pero su comportamiento es completamente predecible una vez observada una cantidad suficiente de salida cuando el polinomio de conexión es fijo; por el contrario, las sucesiones derivadas de la aritmética de

curvas elípticas exhiben una complejidad algebraica elevada ligada a la dificultad del problema del logaritmo discreto elíptico, aunque a un coste computacional sensiblemente mayor.

El objetivo del modelo es combinar lo mejor de ambos enfoques. El LFSR actúa como núcleo eficiente de generación, mientras que una cantidad acotada de información procedente de una sucesión elíptica se inyecta en cada iteración para introducir dependencia temporal no lineal. Esta inyección se realiza de forma interna, modulando dinámicamente uno de los coeficientes del polinomio de realimentación, de modo que se preserva la estabilidad operativa del registro sin exponer directamente información del grupo elíptico en la salida.

En las subsecciones siguientes se describen con detalle los componentes algebraicos implicados, el mecanismo preciso de reducción de la información elíptica y su integración en la dinámica del LFSR, así como la estructura multicapa de las secuencias producidas por el generador y los extractores binarios considerados.

II-A. Estructura algebraica de $\mathbb{F}_{p^n}$

Sea p un número primo y sea $f(x) \in \mathbb{F}_p[x]$ un polinomio irreducible de grado n. El cuerpo finito de orden p^n se construye como el cociente

$$\mathbb{F}_{p^n} = \mathbb{F}_p[x]/\langle f(x)\rangle.$$

Denotamos por ξ la clase de x en el cociente. Entonces el conjunto

$$\mathcal{B} = \{1, \xi, \xi^2, \ldots, \xi^{n-1}\}$$

forma una base vectorial de $\mathbb{F}_{p^n}$ sobre $\mathbb{F}_p$.

Todo elemento $a \in \mathbb{F}_{p^n}$ admite una representación única respecto a esta base,

$$a = \sum_{i=0}^{n-1} a_i\, \xi^i, \qquad a_i \in \mathbb{F}_p,$$

lo que permite identificar de manera natural los elementos del cuerpo con vectores de longitud n sobre el cuerpo primo.

Con este fin introducimos las aplicaciones lineales

$$\phi : \mathbb{F}_{p^n} \longrightarrow \mathbb{F}_p^n, \qquad \phi(a) = (a_0, \ldots, a_{n-1}),$$
$$\psi : \mathbb{F}_p^n \longrightarrow \mathbb{F}_{p^n}, \quad \psi(u_0, \ldots, u_{n-1}) = \sum_{i=0}^{n-1} u_i\, \xi^i.$$

Por construcción, ϕ y ψ son isomorfismos de espacios vectoriales inversos entre sí, y preservan todas las relaciones lineales sobre $\mathbb{F}_p$.

Las operaciones aritméticas en $\mathbb{F}_{p^n}$ se implementan mediante aritmética polinómica con reducción módulo $f(x)$. En particular, el inverso multiplicativo de un elemento no nulo $a \in \mathbb{F}_{p^n}$ puede calcularse como $a^{-1} = a^{p^n-2}$, empleando exponenciación rápida junto con reducciones modulares.

II-A0a. Ventaja estructural.: La representación explícita en coeficientes sobre $\mathbb{F}_p$ permite descomponer cualquier secuencia $\{o_t\} \subset \mathbb{F}_{p^n}$ producida por el LFSR en n secuencias coordinadas

$$\mathcal{C}_i = \{a_{t,i}\}_{t\geq 0} \subset \mathbb{F}_p, \qquad 0 \leq i \leq n-1,$$

lo que resulta especialmente útil para el análisis fino de propiedades estadísticas, correlaciones cruzadas y complejidad lineal de proyecciones del generador.

II-B. Curva elíptica y propiedades de la sucesión kG

Sea E una curva elíptica no singular definida sobre el cuerpo finito $\mathbb{F}_{p^n}$, dada por una ecuación de Weierstrass corta

$$E:\ y^2 = x^3 + ax + b, \qquad a, b \in \mathbb{F}_{p^n},$$

con discriminante $\Delta \neq 0$. El conjunto de puntos $E(\mathbb{F}_{p^n})$, junto con el punto al infinito, forma un grupo abeliano finito bajo la ley de adición elíptica.

Fijamos un punto $G \in E(\mathbb{F}_{p^n})$ de orden grande (preferentemente primo o con cofactor pequeño) y un desplazamiento inicial $t_0 \in \mathbb{Z}$. Definimos la sucesión de puntos

$$P_t = kG, \qquad Q_t = (k+1)G, \qquad k = t_0 + t, \quad t \geq 0.$$

En la práctica, la sucesión se genera de forma incremental mediante una única suma elíptica por iteración,

$$Q_t = P_t + G, \qquad P_{t+1} = Q_t,$$

lo que permite una implementación eficiente.

Cada punto se escribe en coordenadas afines como

$$P_t = (x_t, y_t), \qquad Q_t = (x'_t, y'_t),$$

con $x_t, y_t, x'_t, y'_t \in \mathbb{F}_{p^n}$.

Las aplicaciones

$$k \longmapsto x(kG), \qquad k \longmapsto y(kG)$$

no son funciones polinómicas ni racionales simples en k, y su estructura algebraica está relacionada con la dificultad del problema del logaritmo discreto elíptico (ECDLP). Para curvas adecuadamente seleccionadas, se observa empíricamente que las coordenadas de los puntos sucesivos presentan una dispersión amplia en $\mathbb{F}_{p^n}$, lo que las convierte en una fuente eficaz de complejidad algebraica para modular la dinámica del registro.

Mediante la representación del cuerpo introducida en la Subsección II-A, cada coordenada se expande en coeficientes sobre el cuerpo primo,

$$\phi(x_t), \phi(y_t), \phi(x'_t), \phi(y'_t) \in \mathbb{F}_p^n.$$

En consecuencia, cada punto elíptico aporta $2n$ coeficientes en $\mathbb{F}_p$, y el par de puntos consecutivos (P_t, Q_t) proporciona un total de $4n$ coeficientes estructuralmente relacionados.

Estos $4n$ coeficientes constituyen la materia prima a partir de la cual se construye, en la subsección siguiente, el elemento $\theta_t \in \mathbb{F}_{p^n}$ que modula dinámicamente la evolución del LFSR. El objetivo no es asumir independencia estadística estricta entre dichos coeficientes, sino explotar su complejidad algebraica conjunta para inducir dependencia temporal no lineal en el generador.

II-C. LFSR sobre $\mathbb{F}_{p^n}$: estructura y limitaciones

Un LFSR de longitud m sobre el cuerpo $\mathbb{F}_{p^n}$ está definido por un estado

$$S_t = (s_t^{(0)}, \ldots, s_t^{(m-1)}) \in (\mathbb{F}_{p^n})^m,$$

y un polinomio de conexión

$$C(x) = c_0 + c_1 x + \cdots + c_{m-1} x^{m-1}, \qquad c_i \in \mathbb{F}_{p^n}.$$

La evolución del registro viene dada por

$$s_{t+1}^{(0)} = \sum_{i=0}^{m-1} c_i \, s_t^{(i)}, \qquad s_{t+1}^{(j)} = s_t^{(j-1)}, \quad 1 \le j \le m-1.$$

La salida del registro se toma como

$$o_t = s_t^{(m-1)} \in \mathbb{F}_{p^n}.$$

II-D. Transformación elíptica y reducción $4n \to n$

A partir de los puntos elípticos consecutivos

$$P_t = (x_t, y_t), \qquad Q_t = (x'_t, y'_t) \in E(\mathbb{F}_{p^n}),$$

consideramos sus expansiones en coeficientes respecto a la base $\mathcal{B}$:

$$A_t = \phi(x_t) = (a_{t,0}, \ldots, a_{t,n-1}),$$

$$B_t = \phi(y_t) = (b_{t,0}, \ldots, b_{t,n-1}),$$

$$C_t = \phi(x'_t) = (c_{t,0}, \ldots, c_{t,n-1}),$$

$$D_t = \phi(y'_t) = (d_{t,0}, \ldots, d_{t,n-1}).$$

Esta indexación (de 0 a $n-1$) se mantiene uniformemente en toda la sección.

II-D0a. Paso 1: interleaving de coeficientes.: Construimos un vector $U_t \in \mathbb{F}_p^{4n}$ intercalando los cuatro bloques por coordenada:

$$U_t = (a_{t,0}, b_{t,0}, c_{t,0}, d_{t,0}, \; a_{t,1}, b_{t,1}, c_{t,1}, d_{t,1}, \; \ldots,$$

$$, \; \ldots, a_{t,n-1}, b_{t,n-1}, c_{t,n-1}, d_{t,n-1}),$$

que denotamos como $U_t = (u_{t,0}, \ldots, u_{t,4n-1})$.

II-D0b. Paso 2: ventana central de longitud n.: Dado que U_t tiene longitud $4n$, su zona central se sitúa alrededor del índice $2n$. Definimos una ventana contigua de longitud exacta n mediante

$$W_t = (u_{t,\, 2n-\lfloor n/2 \rfloor}, \; u_{t,\, 2n-\lfloor n/2 \rfloor+1}, \; \ldots,$$

$$, \; \ldots, \; u_{t,\, 2n-\lfloor n/2 \rfloor+n-1}) \in \mathbb{F}_p^n.$$

Este paso implementa de forma explícita la reducción $\mathbb{F}_p^{4n} \longrightarrow \mathbb{F}_p^n$, mezclando coeficientes procedentes de las cuatro coordenadas elípticas, con una selección centrada que evita privilegiar sistemáticamente los primeros o los últimos bloques.

II-D0c. Paso 3: reconstrucción en $\mathbb{F}_{p^n}$.: Finalmente, interpretamos W_t como un elemento del cuerpo mediante

$$\theta_t = \psi(W_t) = \sum_{i=0}^{n-1} W_{t,i} \, \xi^i \in \mathbb{F}_{p^n},$$

donde $W_t = (W_{t,0}, \ldots, W_{t,n-1})$.

II-D0d. Observación.: El interleaving previo garantiza que, en general, cada componente de θ_t depende de coeficientes alternados de (x_t, y_t, x'_t, y'_t), evitando una dependencia sistemática de una única coordenada.

II-E. Actualización dinámica del LFSR

En el generador propuesto, los coeficientes

$$c_0, \ldots, c_{m-2} \in \mathbb{F}_{p^n}$$

se mantienen constantes, mientras que el coeficiente de mayor grado se hace dependiente del tiempo mediante $c_{m-1}(t) = \theta_t$. La ecuación de realimentación pasa a ser

$$s_{t+1}^{(0)} = \sum_{i=0}^{m-2} c_i \, s_t^{(i)} + \theta_t \, s_t^{(m-1)}, \qquad s_{t+1}^{(j)} = s_t^{(j-1)}.$$

La evolución global puede escribirse como $S_{t+1} = A_t S_t$, donde A_t es la matriz *companion* asociada al LFSR cuyo último coeficiente depende de θ_t.

II-F. Extracción de bits: extractores considerados

La salida del LFSR en cada iteración es un elemento

$$o_t = \sum_{i=0}^{n-1} a_{t,i} \, \xi^i \in \mathbb{F}_{p^n}, \qquad a_{t,i} \in \mathbb{F}_p.$$

A partir de $\{o_t\}$ se pueden definir distintos extractores binarios. A continuación se formalizan, de manera uniforme, los extractores empleados o discutidos en este trabajo.

II-F0a. Convención de paridad en $\mathbb{F}_p$.: Como $a_{t,i} \in \mathbb{F}_p$, fijamos la identificación canónica $\iota : \mathbb{F}_p \to \{0, 1, \ldots, p-1\}$, y definimos la paridad de un coeficiente como $\iota(a_{t,i}) \bmod 2$. Con esta convención, el *bit del coeficiente superior* queda definido por

$$h_t = \iota(a_{t,n-1}) \bmod 2 \in \{0, 1\}.$$

Extractor A: carácter cuadrático (dos capas):

II-F0b. 1) Carácter cuadrático.: Para p impar, sea $\chi : \mathbb{F}_{p^n} \to \{-1, 0, +1\}$ el carácter cuadrático definido por

$$\chi(x) = \begin{cases} +1, & x \neq 0 \text{ es residuo cuadrático en } \mathbb{F}_{p^n}, \\ -1, & x \text{ es no residuo}, \\ 0, & x = 0. \end{cases}$$

Operativamente, para $q = p^n$, puede evaluarse como $\chi(x) = x^{(q-1)/2}$ (con la convención $\chi(0) = 0$).

II-F0c. 2) Combinación con el coeficiente superior.: Definimos el bit extraído como

$$b_t^{(\chi,\mathrm{imp})} = \mathbb{1}\{\chi(o_t) = +1\} \oplus h_t.$$

Esta combinación introduce no linealidad adicional al mezclar una propiedad global del elemento o_t (residuocidad cuadrática) con un coeficiente local de su representación.

Extractor B: traza (y variante mejorada): Dado que en la evaluación experimental se emplean extractores basados en la traza (Sección III), formalizamos también este esquema.

II-F0d. 1) Traza de extensión.: Sea $\mathrm{Tr}_{\mathbb{F}_{p^n}/\mathbb{F}_p} : \mathbb{F}_{p^n} \to \mathbb{F}_p$ la traza

$$\mathrm{Tr}(a) = a + a^p + a^{p^2} + \cdots + a^{p^{n-1}}.$$

II-F0e. 2) Binarización de la traza y variante mejorada.: En característica impar, una binarización natural se define especificando un subconjunto $T \subset \mathbb{F}_p$ que se mapea a 1 (y su complemento a 0), con posible descarte de ciertos valores si se desea un flujo irregular. En particular, en el caso experimental $p = 3$ (Sección III) se considera el mapeo condicionado por $\mathrm{Tr}(o_t) \in \{1, 2\}$ y el descarte de $\mathrm{Tr}(o_t) = 0$, y la variante mejorada mezcla adicionalmente h_t:

$$b_t^{(\mathrm{Tr})} = \mathbb{1}\{\mathrm{Tr}(o_t) = 1\} \quad \text{(con descarte si procede)},$$

$$b_t^{(\mathrm{Tr,imp})} = \mathbb{1}\{\mathrm{Tr}(o_t) = 1\} \oplus h_t \quad \text{(con el mismo criterio)}.$$

Esta familia se incluye aquí para fijar notación y asegurar consistencia entre la definición del modelo y su evaluación posterior.

II-G. Secuencias diezmadas y análisis estructural

La expansión de coeficientes induce n secuencias sobre $\mathbb{F}_p$,

$$\mathcal{C}_i = \{a_{t,i}\}_{t \geq 0}, \qquad 0 \leq i \leq n-1.$$

Para estudiar proyecciones binarias ligeras de dichas coordenadas (especialmente útil cuando p es pequeño), definimos sus versiones binarizadas como

$$\beta_{t,i} = \mathbb{1}\{\iota(a_{t,i}) = 1\} \in \{0, 1\}.$$

(En el caso $p = 3$, esto equivale a distinguir el símbolo 1 frente a $\{0, 2\}$, tal y como se emplea en los experimentos.)

Estas secuencias permiten estudiar correlaciones cruzadas, complejidad lineal individual y dependencias algebraicas entre distintas proyecciones del estado interno. Aunque dichas proyecciones contienen información parcial del estado, la dependencia temporal no lineal inducida por θ_t impide, en general, una recomposición directa de la secuencia original $\{o_t\}$ a partir de ellas.

II-H. Secuencia final y estructura multicapa

El generador produce simultáneamente las siguientes familias de secuencias:

$$\begin{aligned}
\mathcal{A} &= \{o_t\} \subset \mathbb{F}_{p^n}, \\
\mathcal{B}_{\chi,\mathrm{imp}} &= \{b_t^{(\chi,\mathrm{imp})}\} \subset \{0,1\}, \\
\mathcal{B}_{\mathrm{Tr}} &= \{b_t^{(\mathrm{Tr})}\} \subset \{0,1\} \text{ (si se emplea)}, \\
\mathcal{B}_{\mathrm{Tr,imp}} &= \{b_t^{(\mathrm{Tr,imp})}\} \subset \{0,1\} \text{ (si se emplea)}, \\
\mathcal{C}_i &= \{a_{t,i}\} \subset \mathbb{F}_p, \qquad 0 \leq i \leq n-1, \\
\mathcal{D}_i &= \{\beta_{t,i}\} \subset \{0,1\}, \qquad 0 \leq i \leq n-1.
\end{aligned}$$

Esta estructura multicapa proporciona múltiples vistas algebraicas del sistema, útiles para el análisis de su comportamiento frente a ataques lineales, algebraicos y estadísticos.

III. Resultados experimentales

En esta sección evaluamos el comportamiento estadístico de las secuencias generadas por el modelo descrito en la Sección II. En la instancia experimental considerada, la salida del registro toma valores en $o_t \in \mathbb{F}_{3^5}$.

A partir de la secuencia $\{o_t\}$ se construyeron dos flujos binarios mediante extractores basados en la traza: (i) **traza pura** y (ii) **traza mejorada**, esta última incorporando además la paridad del coeficiente superior en la representación polinómica del elemento (véase subsección II-F).

Tabla I: Indicadores estadísticos — **Traza pura** (primeros 33,436 bits)

Magnitud	Valor	Comentario
Monobit (Freq.)	$p \approx 0{,}3756$	Equilibrio adecuado
Runs (rachas)	$p \approx 0{,}2189$	Alternancia correcta
Autocorrelación (lag 1)	$-0{,}0068$	Dependencia muy baja

El análisis estadístico se realizó sobre los primeros 33,436 *bits útiles* producidos por cada extractor. Los resultados se interpretan como indicadores básicos (frecuencia, rachas y autocorrelación) comparables con criterios habituales de aceptación en tests estándar de aleatoriedad (p. ej., NIST SP 800-22). En particular, los p-values reportados deben entenderse como medidas exploratorias sobre una única realización representativa del generador.

III-A. Binarización por traza (método 1)

La traza de extensión

$$\mathrm{Tr}_{\mathbb{F}_{3^5}/\mathbb{F}_3}(a) = a + a^3 + a^{3^2} + a^{3^3} + a^{3^4}$$

es una aplicación $\mathbb{F}_3$–lineal cuyo rango es $\mathbb{F}_3 = \{0, 1, 2\}$. Cuando la entrada se comporta de manera pseudo-uniforme, la traza tiende a distribuirse aproximadamente de forma equilibrada entre sus tres valores.

Para obtener un flujo binario se aplicó el mapeo $1 \mapsto 1$, $2 \mapsto 0$, y se descartaron aquellas iteraciones en las que $\mathrm{Tr}(o_t) = 0$. Por tanto, el bitstream resultante es un flujo irregular (por *decimación* dependiente de la traza): la longitud reportada corresponde a los bits efectivamente emitidos. Bajo una hipótesis ideal de equiprobabilidad de la traza, la tasa de emisión esperada sería aproximadamente $2/3$; en la práctica, esta tasa puede desviarse ligeramente y constituye un parámetro observable del extractor (por lo que conviene tener en cuenta su posible impacto en modelos donde el adversario observe no sólo el bitstream, sino también el patrón temporal de emisión).

Los indicadores estadísticos se muestran en la Tabla I. En particular, el estadístico de autocorrelación reportado para *lag 1* se entiende como correlación entre bits consecutivos *emitidos* (esto es, tras la decimación).

III-B. Binarización por traza mejorada

Con el fin de reducir sesgos residuales a escala local, definimos el bitstream mejorado como

$$b_t^{(\mathrm{imp})} = \mathbb{1}\{\mathrm{Tr}(o_t) = 1\} \ \oplus \ \big(\iota(a_{t,4}) \bmod 2\big),$$

donde $o_t = \sum_{i=0}^{4} a_{t,i}\xi^i$ con $a_{t,i} \in \mathbb{F}_3$, $a_{t,4}$ es el coeficiente asociado a ξ^4 (coeficiente superior en $\mathbb{F}_{3^5}$ bajo base polinómica), e $\iota : \mathbb{F}_3 \to \{0, 1, 2\}$ denota la identificación canónica usada para definir la paridad. En esta evaluación, el criterio de emisión sigue siendo el mismo que en el método 1 (esto es, la decimación determinada por la traza), y la operación $\oplus$ se realiza en $\mathbb{F}_2$.

Esta mezcla introduce una perturbación ligera sobre el bit extraído de la traza, sin alterar la eficiencia del extractor: combina una propiedad global del elemento (su traza) con

Tabla II: Indicadores estadísticos — **Traza mejorada** (primeros 33,436 bits)

Magnitud	Valor	Comentario
Monobit (Freq.)	$p \approx 0{,}0279$	Dentro del umbral ($p \geq 0{,}01$)
Runs (rachas)	$p \approx 0{,}0564$	Alternancia estable
Autocorrelación (lag 1)	$-0{,}0105$	Correlación aún menor

un coeficiente local de su representación, lo que tiende a amortiguar sesgos de corto alcance sin modificar el núcleo generador sobre $\mathbb{F}_{3^5}$. Los indicadores estadísticos se recogen en la Tabla II.

III-C. Secuencias diezmadas y complejidad lineal

Sea

$$o_t = \sum_{i=0}^{4} a_{t,i}\xi^i, \qquad a_{t,i} \in \mathbb{F}_3.$$

Cada coordenada induce una secuencia ternaria

$$\mathcal{C}_i = \{a_{t,i}\}_{t\geq 0}, \qquad 0 \leq i \leq 4.$$

Para estudiar su estructura lineal bajo una observación binaria ligera (en el espíritu de un atacante que sólo observa una proyección muy limitada), definimos

$$\beta_{t,i} = \begin{cases} 1, & a_{t,i} = 1, \\ 0, & a_{t,i} \in \{0, 2\}. \end{cases}$$

Esta binarización preserva parcialmente la información de la coordenada, pero introduce no linealidad respecto de la dinámica original sobre $\mathbb{F}_{3^5}$.

La complejidad lineal de las secuencias $\{\beta_{t,i}\}$ generadas en el experimento es $L_i \approx 2500$, valor muy superior a la longitud del registro ($m = 5$).

III-D. Visualización y análisis cualitativo

Las figuras siguientes complementan los indicadores numéricos:

- El histograma del flujo por traza pura muestra dispersión global sin sesgos evidentes a simple vista.
- Las curvas de autocorrelación sugieren que la traza mejorada reduce dependencia de corto alcance en los bits emitidos.
- La distribución de longitudes de rachas es compatible con el comportamiento esperado en flujos aproximadamente equilibrados.
- Las correlaciones cruzadas entre proyecciones diezmadas son pequeñas en magnitud, lo que sugiere dependencia limitada entre coordenadas observadas bajo las proyecciones consideradas.

Las correlaciones entre proyecciones son pequeñas y la complejidad lineal estimada es elevada en las binarizaciones consideradas. Estos resultados apoyan empíricamente que la perturbación elíptica introduce dependencia temporal y rompe la linealidad fija típica de un LFSR clásico, sin que se observen artefactos estadísticos inmediatos en las métricas evaluadas.

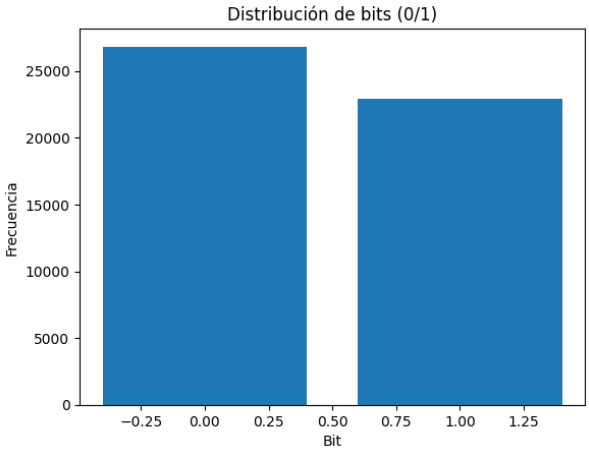


Figura 1: Histograma de bits (traza pura).

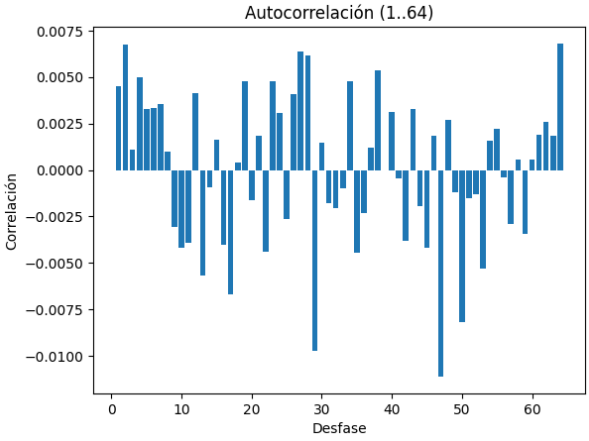


Figura 2: Autocorrelación: traza pura vs. traza mejorada.

III-E. Complejidad computacional del generador

Analizamos el coste por iteración del generador en función de los parámetros (p, n, m), donde p es la característica, n el grado de extensión y m la longitud del LFSR. Representamos los elementos de $\mathbb{F}_{p^n}$ en base polinómica $\{1, \xi, \ldots, \xi^{n-1}\}$, de modo que una suma cuesta $O(n)$ y una multiplicación cuesta $O(n^2)$ mediante aritmética polinómica con reducción módulo $f(x)$ (sin entrar en optimizaciones específicas como Karatsuba/Toom-Cook para n grande).

La actualización del registro requiere m multiplicaciones y m sumas en $\mathbb{F}_{p^n}$. En el modelo de coste adoptado, esto implica $T_{\mathrm{LFSR}}(n, m) = O(mn^2)$, donde el término dominante corresponde a las multiplicaciones en el cuerpo.

Cada iteración actualiza un punto mediante una suma elíptica $Q_t = P_t + G, \quad P_{t+1} = Q_t$. En coordenadas afines, una suma típica requiere una inversión en $\mathbb{F}_{p^n}$ además de varias multiplicaciones. Denotando por $I(n)$ el coste de una inversión (según el método de inversión elegido y la representación), se obtiene $T_{\mathrm{EC}}(n) = O(I(n) + n^2)$. En implementaciones prácticas puede emplearse coordenadas proyectivas para reducir la frecuencia de inversiones, sustituyéndolas por un número adicional de multiplicaciones en $\mathbb{F}_{p^n}$; esta decisión afecta a constantes pero no altera el esquema de descomposición del coste.

La expansión de coeficientes mediante ϕ produce cuatro

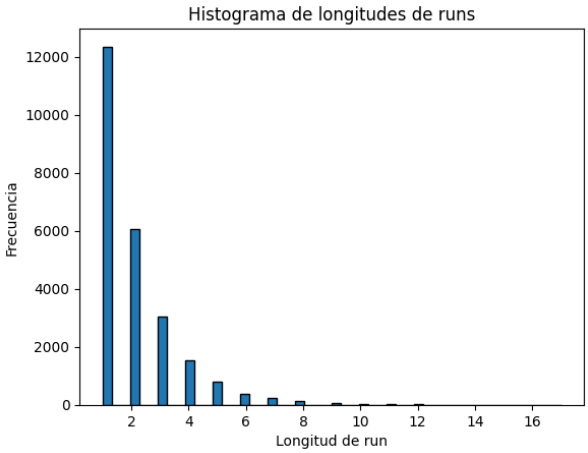


Figura 3: Distribución de longitudes de rachas.

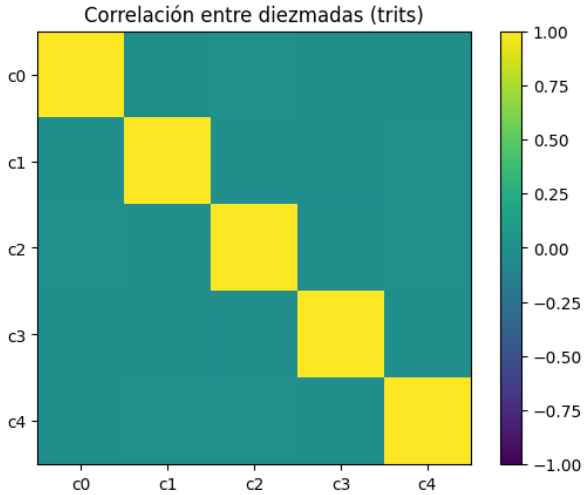


Figura 4: Correlación entre diezmadas (trit-valuadas).

bloques en $\mathbb{F}_p^n$. El interleaving genera un vector de longitud $4n$ y la extracción de una ventana central produce $W_t \in \mathbb{F}_p^n$. Estas operaciones son lineales en n: $T_\theta(n) = O(n)$. La reconstrucción $\theta_t = \psi(W_t)$ es igualmente $O(n)$.

La asignación $c_{m-1}(t) \leftarrow \theta_t$ es de coste constante $T_{\text{update}} = O(1)$, mientras que el coste de los extractores está dominado por potencias de Frobenius (en característica p) y sumas en el cuerpo. En particular, para $q = p^n$,

$$\mathrm{Tr}(a) = a + a^p + a^{p^2} + \cdots + a^{p^{n-1}},$$

y su evaluación puede implementarse eficientemente en aritmética de campo. En términos asintóticos, el coste por iteración queda acotado por $T_{\mathrm{Tr}}(n) = \widetilde{O}(n^2)$, dependiendo de la representación y de la implementación de las potencias de Frobenius (que, en ciertas bases, pueden ser especialmente eficientes).

Sumando componentes, el coste por iteración queda dominado por la aritmética elíptica y la actualización del LFSR: $T_{\text{total}}(n, m) = O(mn^2) + O(T_{\text{EC}}(n)) + O(n)$. Para extensiones pequeñas (como $n = 5$), el coste es compatible con implementaciones ligeras, especialmente si se genera la sucesión elíptica de forma incremental y se elige una representación eficiente del cuerpo.

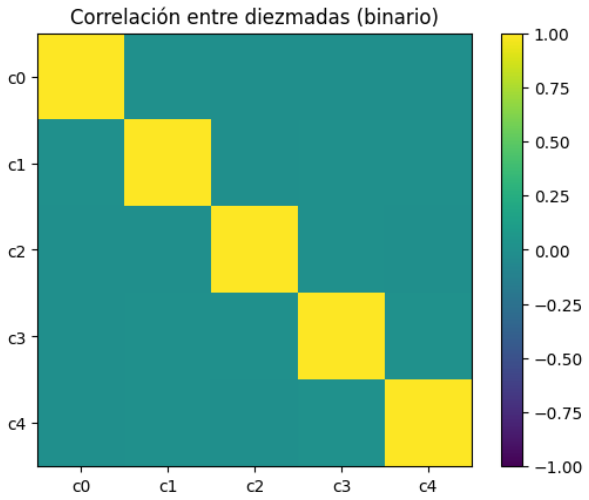


Figura 5: Correlación entre diezmadas (binarias).

IV. Conclusiones

El generador presentado conserva la simplicidad y eficiencia estructural del LFSR, al tiempo que introduce dependencia temporal no lineal mediante el parámetro θ_t, cuya obtención se basa en las coordenadas de (P_t, Q_t) y en una reducción explícita de $4n$ a n coeficientes mediante un esquema de interleaving y ventana central.

En la instancia experimental considerada sobre $\mathbb{F}_{3^5}$, los extractores basados en la traza producen flujos binarios con buen equilibrio, baja autocorrelación y ausencia de patrones evidentes en las métricas estadísticas básicas evaluadas. Asimismo, el análisis de las proyecciones diezmadas muestra complejidades lineales elevadas y correlaciones cruzadas reducidas, lo que sugiere que la perturbación elíptica degrada eficazmente la estructura lineal observable del LFSR. En conjunto, los resultados empíricos apoyan que el modelo ofrece un compromiso adecuado entre coste computacional, interpretabilidad algebraica y calidad estadística del flujo generado, sin introducir artefactos inmediatos en los tests considerados.

Referencias

[1] S. H. AbdElHaleem, S. K. Abd-El-Hafiz y A. G. Radwan, "A generalized framework for elliptic curves based PRNG and its utilization in image encryption," *Scientific Reports*, vol. 12, p. 13278, 2022.

[2] G. Gong, T. A. Berson y D. R. Stinson, "Elliptic Curve Pseudorandom Sequence Generators," en *Proc. Selected Areas in Cryptography (SAC'99)*, pp. 34–48, 1999.

[3] J. Gutiérrez, "Attacking the linear congruential generator on elliptic curves via lattice techniques," *Cryptography and Communications*, vol. 14, pp. 505–525, 2022.

[4] T. Mefenza y D. Vergnaud, "Inferring sequences produced by elliptic curve generators using Coppersmith's methods," *Theoretical Computer Science*, vol. 816, pp. 20–42, 2020.

[5] T. Haider *et al.*, "A novel pseudo-random number generator based on elliptic curve genetic algorithm for image encryption," *Expert Systems with Applications*, p. 122446, 2024.

[6] O. Reyad, "Random Bit Generator Mechanism Based on Elliptic Curves," arXiv:2002.09239, 2020.

[7] D. Shumow y N. Ferguson, "Cryptanalysis of the Dual Elliptic Curve Pseudorandom Generator," Microsoft Crypto Developers Conference (talk), 2007.

[8] C. P. Mok *et al.*, "Pseudorandom vector generation using elliptic curves and bit-extraction schemes," ScienceDirect (S1071579722001381), 2023.

Cibernética inversa: detección estructural de ciberataques por ruptura de simetría ecosistémica

Carlos Mario Braga
Grp. Invest. Alarcos
Unv. de Castilla-La Mancha
UCLM, Ciudad Real
CarlosMario.Braga1@alu.uclm.es

Manuel A. Serrano
Grp. Invest. Alarcos
Unv. de Castilla-La Mancha
UCLM, Ciudad Real
Manuel.Serrano@uclm.es

Eduardo Fernández-Medina
Grp. Invest. GSyA
Unv. de Castilla-La Mancha
UCLM, Ciudad Real
Eduardo.FdezMedina@uclm.es

***Resumen*—Los ataques evasivos, especialmente aquellos de naturaleza mimética, representan un reto creciente para los sistemas de detección actuales, al replicar patrones del tráfico legítimo y eludir tanto reglas como modelos entrenados. Esta limitación evidencia la fragilidad de los enfoques tradicionales, basados en decisiones locales sobre eventos individuales.**

Como alternativa, este trabajo propone una estrategia inspirada en la cibernética clásica, que modela el entorno digital como un ecosistema dinámico en equilibrio estadístico. Dicho equilibrio se representa mediante una distribución empírica de vectores observables, que describe el comportamiento global del sistema. En este contexto, se introduce el concepto de *ruptura de simetría ecosistémica*, entendido como una desviación significativa y persistente respecto al estado de referencia legítimo, lo que refleja una alteración estructural potencialmente causada por actividad ofensiva.

Para su cuantificación se emplean la divergencia de Jensen–Shannon y la distancia de Wasserstein, capaces de anticipar desequilibrios estructurales sin depender de alertas puntuales.

La propuesta se valida sobre el conjunto CIC IDS 2017, mediante inyecciones progresivas de tráfico ofensivo evasivo. Los resultados muestran que este enfoque detecta perturbaciones sistémicas antes que técnicas convencionales como *Isolation Forest*, empleado para la detección no supervisada de anomalías, lo que respalda su utilidad como mecanismo complementario de vigilancia estructural no supervisada en ecosistemas digitales complejos.

***Index Terms*—Detección de intrusiones, ciberataques evasivos, vigilancia estructural, divergencias estadísticas, ecosistemas digitales**

I. Introducción

Los sistemas actuales de detección de intrusiones dependen en gran medida de enfoques basados en firmas o aprendizaje automático supervisado [1], incluyendo aplicaciones en dominios verticales como redes IoT médicas [2]. Si bien estos sistemas son eficaces frente a ataques conocidos y patrones aprendidos, presentan limitaciones significativas incluso ante ciberataques *evasivos*, capaces de eludir tanto sistemas basados en reglas como arquitecturas modernas no supervisadas [3]. Dentro de esta categoría, destacan los ataques *miméticos*, que replican intencionadamente características del tráfico legítimo para pasar desapercibidos incluso ante modelos de detección basados en aprendizaje automático [3].

Estos ataques no presentan anomalías individuales, pero alteran el comportamiento global incluso sin generar señales detectables en modelos centrados en la instancia [1]. Esta limitación evidencia una debilidad estructural en los sistemas de defensa, que operan sobre eventos aislados y decisiones locales, sin una visión holística del ecosistema digital ni mecanismos para detectar alteraciones emergentes en su estructura operativa [4].

Como respuesta a esta debilidad estructural, este trabajo propone un cambio de enfoque inspirado en la cibernética clásica de Norbert Wiener [5]. Bajo esta perspectiva, el entorno digital se concibe como un *ecosistema cibernético* abierto, en el que aplicaciones, mecanismos de defensa y actores ofensivos coexisten en un equilibrio dinámico, caracterizado por la continua adaptación de sus interacciones a través de flujos de información, cuya estabilidad puede verse alterada por presiones ofensivas sutiles.

Bajo esta lógica, un ciberataque puede entenderse como una *mutación ofensiva* que induce una *ruptura en la simetría estadística del ecosistema*, entendida como la pérdida del equilibrio estadístico entre los componentes del entorno digital. Dicha ruptura refleja una alteración global en la estructura de las distribuciones que describen el comportamiento legítimo del sistema, incluso antes de manifestarse como incidente detectable. Se plantea la hipótesis de que este tipo de perturbaciones estructurales pueden ser identificadas a través del análisis de diferencias estadísticas [6, 7] entre el estado actual del ecosistema P_t y una distribución de referencia P_0 que represente su comportamiento legítimo o histórico.

Formalmente, se considera que ocurre una *ruptura ecosistémica* si existe una divergencia o distancia $D \in \{JS, W1\}$ tal que:

$$D(P_t, P_0) > \tau,$$

donde τ representa un umbral empírico de significación estimado mediante remuestreo aleatorio con reemplazo (bootstrap) bajo condiciones normales. En este trabajo, JS denota la divergencia de Jensen–Shannon [6], basada en información mutua entre distribuciones, y $W1$ la distancia de Wasserstein de primer orden [7], que cuantifica el costo geométrico de transformar una distribución en otra.

El presente trabajo desarrolla una propuesta metodológica orientada a la detección estructural no supervisada de ataques evasivos. Más allá de su contribución experimental, propone una lectura epistemológica de la ciberdefensa como disciplina estructural, donde la detección de amenazas se entiende no solo como un problema de clasificación, sino como un proceso de diagnóstico sistémico de desequilibrios emergentes. La propuesta se articula en tres contribuciones principales:

- Se propone un modelo de estado ecosistémico basado en una distribución empírica multivariante en $\mathbb{R}^n$, donde cada vector representa un evento observable, capturando así la estructura colectiva del entorno digital en lugar de depender de instancias aisladas.
- Se introduce un mecanismo de diagnóstico basado en

medidas estadísticas, JS y $W1$, que permite identificar rupturas significativas en la simetría del ecosistema, incluso en ausencia de señales locales de anomalía.
- Se contrasta experimentalmente la sensibilidad temprana de esta aproximación estructural frente a técnicas no supervisadas tradicionales, como el algoritmo *Isolation Forest* [8], empleado para la detección de anomalías mediante el aislamiento iterativo de instancias, mediante inyecciones progresivas de tráfico evasivo sobre el conjunto CIC IDS 2017 [9][1].

Los resultados experimentales validan cuantitativamente la hipótesis y muestran que el análisis estructural mediante medidas estadísticas permite anticipar perturbaciones provocadas por ciberataques evasivos, incluso en fases donde técnicas tradicionales como *Isolation Forest* no generan alertas. Este comportamiento refuerza el valor del enfoque propuesto como una capa complementaria de vigilancia sistémica no supervisada, orientada a la detección temprana de desequilibrios operativos.

II. Marco teórico y conceptual

Norbert Wiener formuló la cibernética como el estudio del control y la comunicación en sistemas vivos y artificiales, poniendo el foco en el equilibrio dinámico sostenido por la retroalimentación [5]. Bajo esta perspectiva, un sistema es una entidad que interactúa con su entorno mediante flujos de información para mantener su homeostasis operativa. Esta visión puede extrapolarse a la ciberseguridad moderna, donde los entornos digitales, formados por aplicaciones, defensas, usuarios y atacantes, constituyen ecosistemas cibernéticos abiertos, sujetos a dinámicas adversas de presión, adaptación y desequilibrio [4].

Lejos de ser una analogía aislada, este enfoque ecosistémico se alinea con marcos recientes que conciben la ciberseguridad como un sistema adaptativo y resiliente, capaz de responder dinámicamente a presiones adversarias [4]. Esta perspectiva puede además interpretarse desde una dimensión sociotécnica, en la que los mecanismos de estabilidad del ecosistema digital emergen de la interacción entre factores humanos, normativos y tecnológicos. Trabajos recientes como [10, 11] refuerzan esta idea al describir la confianza y la resiliencia de los sistemas inteligentes como propiedades sistémicas que dependen de la coadaptación entre los componentes técnicos y las estructuras sociales y regulatorias que los sustentan. Esta conceptualización actualiza la cibernética wieneriana hacia una visión sociotécnica, donde los bucles de retroalimentación integran interacción humana, regulación y tecnología en un mismo equilibrio ecosistémico.

En este trabajo, se modela el estado de un ecosistema cibernético como una distribución empírica de vectores P_t, donde cada vector $x_i \in \mathbb{R}^n$ representa un evento individual, como una conexión, un acceso o una decisión automatizada, codificado mediante variables observables: duración, volumen de tráfico, marcas de control, puertos, comportamiento de red, entre otras. Esta representación multivariante permite capturar la estructura estadística del comportamiento operativo del sistema en un intervalo temporal determinado [12].

Se plantea que ciertos ataques, especialmente aquellos evasivos o miméticos, no se manifiestan como anomalías locales, sino como perturbaciones en la estructura global del ecosistema. Estas perturbaciones se interpretan como *rupturas de simetría ecosistémica*, entendidas aquí como desviaciones estadísticamente significativas respecto de una distribución de referencia P_0, que representa el estado equilibrado o históricamente legítimo del ecosistema. La calidad del análisis depende de la estabilidad de P_0, que debe construirse sin actividad ofensiva y reflejar de forma representativa el comportamiento legítimo. Aunque este trabajo parte de un baseline limpio, futuras extensiones podrían incorporar mecanismos que distingan entre cambios operativos legítimos y perturbaciones inducidas por actividad ofensiva.

Para capturar esta ruptura, se emplean medidas estadísticas que cuantifican el cambio estructural entre P_t y P_0. En particular:

- JS mide el desacuerdo promedio entre distribuciones, siendo una medida de divergencia simétrica, acotada y sensible a diferencias locales en densidad.
- $W1$ es una distancia que estima el coste geométrico de transformar una distribución en otra, capturando desplazamientos estructurales incluso si las densidades marginales son similares.

Ambas medidas permiten detectar cambios globales en el ecosistema sin requerir entrenamiento supervisado ni modelos parametrizados, lo que las hace especialmente adecuadas para vigilancia estructural no intrusiva.

Esta capacidad para anticipar desplazamientos estructurales resulta especialmente valiosa en escenarios evasivos, donde los eventos ofensivos emulan parcialmente el comportamiento legítimo del ecosistema.

Al no requerir etiquetas ni entrenamiento, este enfoque es especialmente adecuado para detectar fases tempranas de campañas ofensivas o adaptaciones adversarias, antes de que se manifiesten como anomalías locales. Su carácter estructural permite integrarlo como métrica complementaria en sistemas de detección pasiva o plataformas SIEM (Gestión de Información y Eventos de Seguridad), proporcionando alertas tempranas ante desequilibrios estadísticos durante la operación. Diversos trabajos aplican medidas estadísticas al análisis no supervisado de anomalías, ya sea desde una perspectiva teórica [6] o mediante marcos empíricos como flujos multiescala [13], típicamente en dominios como visión por computador o clasificación binaria. No obstante, estos enfoques dependen de arquitecturas entrenadas o proyecciones locales (e.g., autoencoders, PCA), y rara vez abordan rupturas estructurales en entornos distribuidos o evasivos. En cambio, nuestra propuesta opera sobre la forma global del ecosistema digital sin entrenamiento, permitiendo detectar desequilibrios sistémicos sin depender de instancias etiquetadas ni patrones individuales. Esta combinación de foco operativo y bajo requerimiento computacional la posiciona como una contribución original dentro del marco de vigilancia estructural en ciberseguridad.

III. Metodología

Para validar que los ataques evasivos rompen la simetría del ecosistema sin generar anomalías individuales, se diseñó un experimento que compara dos enfoques complementarios. Por un lado, se emplea detección puntual basada en *puntuaciones de anomalía* (*outlier scores*); por otro, un diagnóstico estructu-

[1] https://www.unb.ca/cic/datasets/ids-2017.html

Tabla I. Resumen del diseño experimental

Elemento	Descripción
Distribución de referencia P_0	Construida exclusivamente con eventos benignos. Representa el estado operativo legítimo del ecosistema.
Distribuciones modificadas P_t	Generadas inyectando progresivamente entre un 1 % y un 25 % de eventos ofensivos miméticos seleccionados de forma consecutiva.
Representación vectorial de eventos	Cada evento se codifica como un vector $x_i \in \mathbb{R}^n$, con atributos numéricos normalizados y variables categóricas codificadas mediante *one-hot encoding*.
Estrategia estructural	Cálculo de divergencias o medidas estadísticas (JS y $W1$) entre P_t y P_0, para detectar cambios en la forma global de la distribución.
Estrategia puntual	Aplicación del modelo de referencia *Isolation Forest*, entrenado únicamente con eventos benignos. Se evalúa qué porcentaje de eventos ofensivos en P_t es clasificado como anómalo.
Calibración estructural	Estimación empírica del umbral de significación mediante remuestreo aleatorio con reemplazo sobre subconjuntos benignos $P_t^{\mathrm{b}} \subset P_0$.
Objetivo comparativo	Verificar si las medidas de divergencia o distancias estructurales permiten anticipar la presencia de ataques antes de que sean detectables como anomalías individuales.

ral fundamentado en medidas estadísticas entre distribuciones empíricas.

III-A. Conjunto de datos y preprocesamiento

Para este estudio se utilizó el conjunto de datos **CIC IDS 2017** [9], una referencia ampliamente utilizada en la evaluación empírica de sistemas de detección de intrusiones. Este conjunto contiene tráfico de red etiquetado con eventos benignos y diversos tipos de ataques, entre ellos categorías especialmente evasivas, como la *infiltración* (*Infiltration*) y los *ataques por fuerza bruta sobre aplicaciones web* (*Web Attack – Brute Force*), que replican características del tráfico legítimo y suelen eludir mecanismos de detección puntual.

Cada evento fue representado como un vector $x_i \in \mathbb{R}^n$, compuesto por atributos relevantes como duración de conexión, número de paquetes, volumen de datos enviados y recibidos, tipo de protocolo, número de puerto y estadísticas derivadas del flujo. Las variables categóricas se transformaron por codificación binaria dispersa (*one-hot encoding*), y las numéricas se normalizaron mediante transformación *z-score*, tras eliminar columnas constantes o con valores no numéricos válidos.

Además, se calculó la variable *InterArrivalTime*, definida como la diferencia temporal entre eventos consecutivos ordenados por `Timestamp`, con el objetivo de capturar la densidad temporal local de los flujos, no directamente observable a partir de atributos individuales.

El preprocesamiento fue aplicado de forma conjunta sobre eventos benignos y ofensivos miméticos, garantizando consistencia estructural y condiciones realistas de escalado. Toda la preparación de datos, incluyendo la selección de variables, eliminación de columnas irrelevantes (`IP`, puertos de origen, etc.), normalización y generación de los ficheros finales, ha sido documentada y es completamente reproducible a partir del repositorio disponible en:

https://github.com/GSYAtools/DTAME

Dicho repositorio incluye todo el código fuente, los parámetros utilizados, las muestras intermedias y los resultados finales del experimento, permitiendo su validación independiente en su totalidad. Esta referencia cubre todas las fases del proceso experimental, por lo que no se repite en secciones posteriores.

III-B. Diseño experimental y calibración estructural

El experimento se estructura a partir de una distribución base P_0, construida exclusivamente con eventos benignos, que representa el estado de equilibrio operativo del ecosistema digital. A partir de P_0, se generan distribuciones modificadas P_t, inyectando progresivamente entre un 1 % y un 25 % de eventos ofensivos evasivos. Cada P_t mantiene el mismo número total de eventos que P_0, reemplazando una fracción de ellos por instancias ofensivas extraídas de forma consecutiva, con el fin de simular campañas realistas.

Para cada distribución P_t, se aplicaron dos estrategias complementarias de evaluación:

- Cálculo de la divergencia JS y la distancia $W1$ entre P_t y P_0, como medida del cambio estructural global en la forma del ecosistema.
- Evaluación de la anomalía individual mediante el modelo de referencia *Isolation Forest*, entrenado con eventos benignos, calculando el porcentaje de eventos ofensivos en P_t que superan un umbral predefinido de valor de anomalía.

El algoritmo *Isolation Forest* se emplea como modelo de referencia representativo de las técnicas clásicas de detección no supervisada, debido a su amplia adopción en la literatura y a su bajo coste computacional. Se entrena exclusivamente sobre eventos benignos y estima la rareza de nuevas instancias en función de su aislamiento dentro del espacio de características. Aunque hay métodos más complejos, este ofrece una referencia estándar y de bajo coste para evaluar detección puntual frente a ataques evasivos.

La distribución P_0 se construyó exclusivamente a partir de tráfico benigno, sin mezcla con actividad ofensiva. Su legitimidad como referencia depende de que el intervalo de observación seleccionado represente fielmente el comportamiento típico del sistema. En escenarios reales, esta condición puede verse afectada por cambios operativos legítimos en el patrón de actividad. Abordar técnicas que permitan distinguir entre estas variaciones y rupturas estructurales maliciosas constituye una línea de trabajo futura prioritaria.

Con el objetivo de fortalecer la validez estadística del diagnóstico estructural, se implementó un procedimiento de remuestreo aleatorio con reemplazo sobre P_0. Este consiste en generar múltiples subconjuntos $P_t^{\mathrm{b}} \subset P_0$, todos compuestos exclusivamente por eventos benignos seleccionados aleatoriamente, y calcular sus medidas JS y $W1$ respecto a P_0. Este análisis permite estimar la variabilidad esperada bajo condiciones normales (sin ataque), construyendo así un umbral empírico que define el límite inferior de significancia estructural. Solo las medidas observadas en P_t que superen

este umbral serán consideradas indicativas de una posible ruptura ecosistémica.

La Tabla I resume las fases clave del proceso de inyección, modelado y evaluación del experimento.

III-C. Criterio de validación

La hipótesis de ruptura de simetría ecosistémica se considera respaldada si se verifican de forma consistente las siguientes condiciones:

- La divergencia o distancia entre P_t y P_0 aumenta significativamente con el porcentaje de inyección y supera de forma sostenida el umbral empírico obtenido mediante remuestreo aleatorio con reemplazo sobre subconjuntos benignos $P_t^{\mathrm{b}} \subset P_0$.
- El porcentaje de eventos ofensivos clasificados como anómalos por el modelo de referencia *Isolation Forest* se mantiene bajo ($\leq 10\,\%$) en las primeras fases de inyección, lo que indica que estos eventos no presentan desviaciones individuales evidentes.

El cumplimiento simultáneo de estas dos condiciones permite validar que el enfoque estructural es capaz de anticipar desequilibrios provocados por ataques evasivos, incluso en ausencia de alertas locales, ofreciendo así una métrica robusta de vigilancia anticipada con control estadístico de significación.

IV. Resultados y análisis

Esta sección presenta los resultados experimentales obtenidos para validar la hipótesis de ruptura de simetría ecosistémica. Se comparan la evolución de las divergencias y distancias estadísticas y la capacidad de detección puntual del modelo de referencia, en función del porcentaje de eventos ofensivos inyectados en el ecosistema.

IV-A. Calibración de medidas en entornos benignos

Para estimar la variabilidad natural de las medidas estadísticas bajo condiciones sin ataque, se aplicó un procedimiento de remuestreo aleatorio con reemplazo sobre el conjunto P_0. Se generaron $N = 100$ subconjuntos $P_t^{\mathrm{b}} \subset P_0$, todos ellos compuestos exclusivamente por eventos benignos seleccionados aleatoriamente, y se calcularon la divergencia de JS y la distancia $W1$ entre cada subconjunto y la distribución de referencia P_0.

Este análisis permitió construir una distribución empírica de valores de JS y $W1$ en situaciones legítimas, a partir de la cual se estimaron los percentiles 95 como umbrales de significación estructural. Los resultados, resumidos en la Tabla II, muestran que los percentiles 95 se sitúan cerca de la media en ambas medidas, lo que sugiere una baja variabilidad estructural en condiciones benignas y refuerza la estabilidad del umbral estimado.

Tabla II. Resumen estadístico del procedimiento de remuestreo aleatorio con reemplazo con tráfico benigno

Métrica	Media	Percentil 95
Divergencia JS	0.076	0.096
Distancia $W1$	0.035	0.043

Las Figuras 1 y 2 muestran los histogramas obtenidos, que permiten visualizar la distribución natural de la divergencia de JS y de la distancia $W1$ respectivamente, bajo condiciones benignas. Estos resultados proporcionan un criterio empírico adicional para diferenciar perturbaciones estructurales genuinas de variaciones funcionales legítimas.

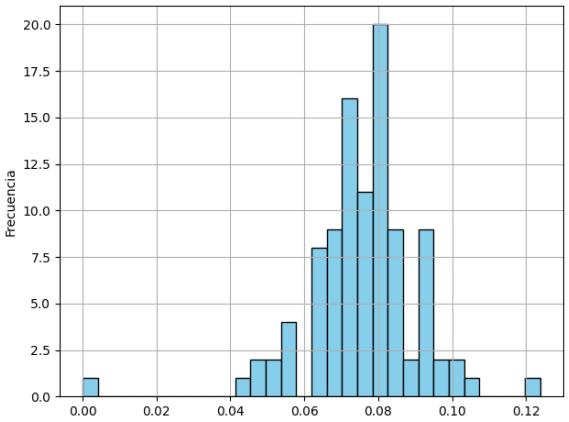


Figura 1. Distribución empírica de divergencias JS entre subconjuntos benignos (remuestreo aleatorio con reemplazo).

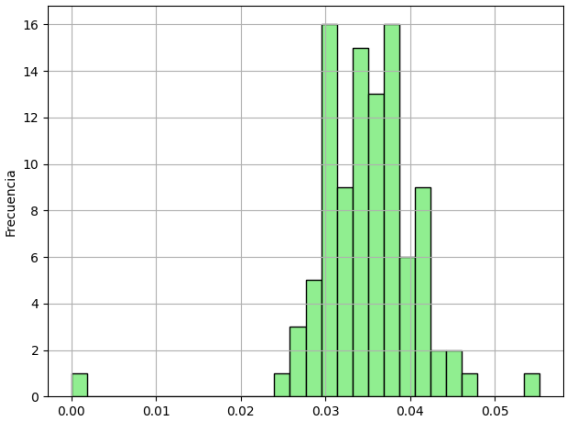


Figura 2. Distribución empírica de distancias $W1$ entre subconjuntos benignos (remuestreo aleatorio con reemplazo).

A partir de esta calibración, se analizan en la siguiente subsección las curvas de medidas obtenidas tras la inyección progresiva de eventos ofensivos, con el objetivo de determinar en qué punto superan significativamente la variabilidad estructural esperada.

IV-B. Evolución de las medidas estructurales

La Figura 3 muestra la evolución de las medidas estructurales JS y $W1$ entre la distribución de referencia P_0 y las distribuciones perturbadas P_t generadas con diferentes proporciones de eventos ofensivos.

La distancia $W1$ crece claramente y responde desde inyecciones del $1{-}5\,\%$, mostrando alta sensibilidad a desplazamientos sutiles en la estructura del ecosistema. En contraste, la divergencia de JS no muestra una evolución sistemática con el porcentaje de inyección, pero sí supera el umbral empírico desde las primeras fases, lo que la convierte en un indicador robusto de ruptura ecosistémica temprana.

Este comportamiento complementario refuerza la utilidad del enfoque propuesto: mientras $W1$ permite seguir la progresión del desequilibrio estructural, JS actúa como señal de activación inicial. Ambas medidas, desde funciones distintas, validan conjuntamente el criterio de sensibilidad estructural.

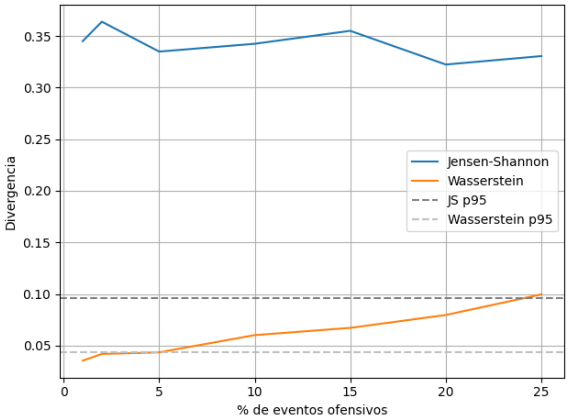


Figura 3. Evolución de las medidas estadísticas en función del porcentaje de eventos ofensivos inyectados. Las líneas punteadas indican los umbrales empíricos del percentil 95 calculados mediante remuestreo aleatorio con reemplazo sobre subconjuntos benignos.

IV-C. Limitaciones de la detección puntual

Se evaluó el comportamiento del modelo de referencia *Isolation Forest* frente a las inyecciones progresivas de tráfico ofensivo. El modelo fue entrenado sobre el conjunto completo de eventos benignos, extrayendo como umbral de decisión θ^* el valor mínimo del score observado en ese conjunto ($\theta^* = -0{,}2698$) durante la fase de entrenamiento para acotar la tasa de falsos positivos.

Con este umbral, el modelo no clasifica como anómalo ninguno de los eventos de los distintos conjuntos de datos con diferentes porcentajes de inyección, como se observa en la Figura 4). Este comportamiento valida la hipótesis de evasión local: en fases tempranas, los eventos ofensivos miméticos no presentan desviaciones suficientemente visibles para ser considerados atípicos de forma individual.

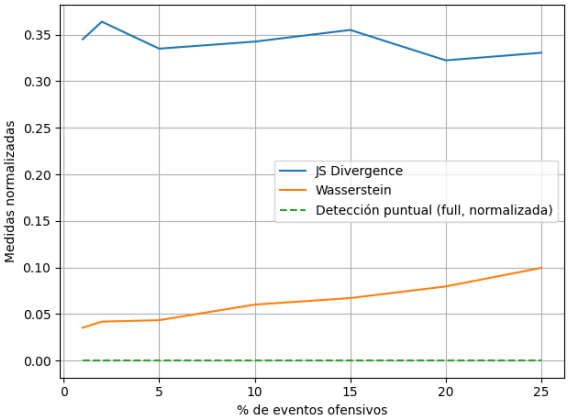


Figura 4. Comparación entre medidas estructurales y detección puntual con *Isolation Forest* calibrado con el umbral obtenido de la fase de entrenamiento.

La Figura 5 muestra un análisis complementario en el que se relaja progresivamente el umbral de decisión. Aunque este ajuste incrementa la tasa de detección de eventos ofensivos, también conduce a un aumento significativo de falsos positivos: por ejemplo, con un umbral de $-0{,}05$, más del 60 % de los eventos clasificados como anómalos pertenecen en realidad al conjunto benigno. Esta degradación en la precisión compromete la fiabilidad operativa del modelo *Isolation Forest* en entornos reales, a la vez que pone de manifiesto de nuevo que la estrategia de la búsqueda de la ruptura ecosistémica confirma la hipótesis de ruptura ecosistémica planteada inicialmente.

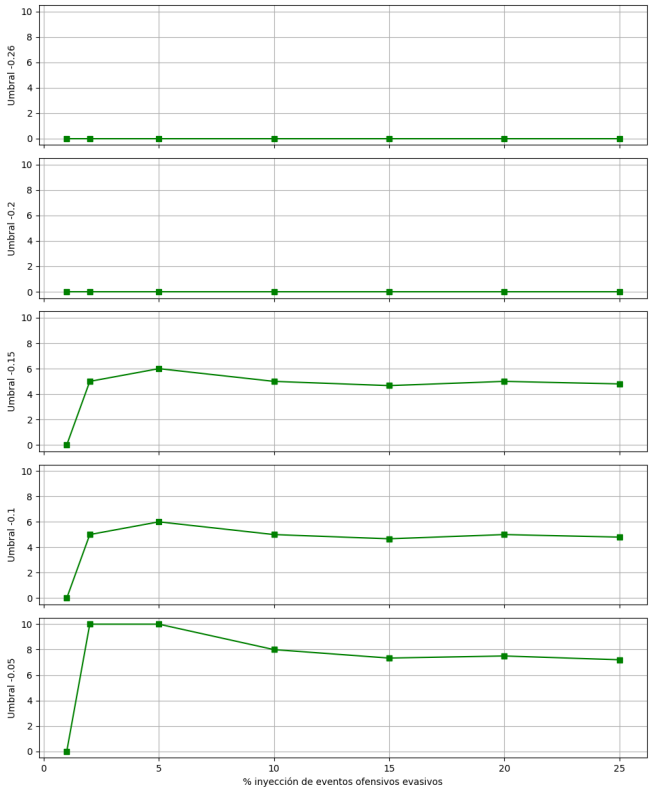


Figura 5. Porcentaje de detección individual en función del umbral de decisión utilizado.

IV-D. Síntesis comparativa

Los resultados muestran que las métricas estructurales permiten anticipar desequilibrios ecosistémicos en fases donde los métodos puntuales aún no reaccionan, validando así su utilidad como herramienta de vigilancia anticipada. En particular, la distancia $W1$ exhibe una alta sensibilidad en las primeras fases de perturbación, reaccionando de forma más pronunciada a partir del 3 % de inyección ofensiva. Esta respuesta puede explicarse por su naturaleza geométrica, que capta desplazamientos sutiles en la forma de la distribución, incluso cuando las densidades marginales se mantienen estables.

Por su parte, la divergencia de JS muestra una activación temprana sin una tendencia claramente creciente, lo que la convierte en una señal de ruptura robusta y menos sensible al ruido incremental. Esta combinación resulta especialmente prometedora, $W1$ actúa como disparador anticipado, mientras que JS aporta una confirmación persistente de que el desequilibrio estructural se mantiene.

En conjunto, ambas métricas proporcionan una señal complementaria, interpretable y no supervisada, adecuada para ser integrada como componente de vigilancia estructural en sistemas de defensa pasiva o plataformas SIEM, especialmente en escenarios donde la visibilidad local es limitada o

manipulable.

V. Discusión y Conclusiones

Los resultados del experimento validan la hipótesis formulada: la ruptura estructural del ecosistema cibernético inducida por ciberataques evasivos puede ser detectada mediante medidas estadísticas antes de que los eventos individuales sean clasificados como anómalos. Las medidas empleadas, JS y $W1$, activan señales estructurales desde fases tempranas de la inyección: JS supera el umbral empírico incluso con un 1 % de eventos ofensivos, mientras que $W1$ muestra una respuesta creciente más marcada a partir del 3–5 %. Este comportamiento confirma la validez del doble criterio definido en la metodología, basado en la sensibilidad estructural combinada con la ausencia de visibilidad local.

La propuesta introduce el concepto de vigilancia estructural del ecosistema, entendido como un mecanismo no supervisado capaz de detectar desviaciones globales respecto a un estado legítimo, sin requerir etiquetas ni conocimiento previo del ataque. Esta estrategia resulta idónea en entornos dinámicos y distribuidos con visibilidad limitada o manipulable. Las métricas estructurales no solo anticipan desequilibrios con bajo coste computacional y sin entrenamiento, sino que resultan idóneas para su integración en sistemas SIEM como señales de riesgo ecosistémico. Su planteamiento se alinea con principios de monitoreo continuo y detección estructural definidos en el *NIST Cybersecurity Framework*, así como con los requisitos de gestión del riesgo y detección temprana recogidos en el Reglamento de Inteligencia Artificial de la Unión Europea, lo que refuerza su aplicabilidad en entornos regulados.

Sin embargo, el enfoque presenta ciertas limitaciones, ya que su sensibilidad depende de la estabilidad del conjunto de referencia P_0, que puede verse alterada por variaciones funcionales legítimas no relacionadas con actividad maliciosa, lo que requiere mecanismos que discriminen entre cambios operativos y rupturas ofensivas. Además, el entorno controlado del dataset CIC IDS 2017 no refleja la complejidad temporal ni topológica de sistemas reales, lo que limita la generalización directa de los resultados.

Como líneas futuras, se plantea validar la estrategia en entornos federados y multiorigen, desarrollar mecanismos que distingan entre variaciones legítimas y ofensivas, y combinar señales estructurales con técnicas locales (p. ej., autoencoders o GNNs) para reforzar la detección híbrida. Asimismo, se explorará su aplicación en contextos de aprendizaje federado y detección distribuida, orientada a fortalecer la resiliencia colaborativa.

En conjunto, este trabajo avanza hacia una ciberdefensa capaz de anticipar desequilibrios sin depender de firmas, consolidando la estabilidad adaptativa como principio estructural.

Agradecimientos

Esta publicación no habría sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MCIN/AEI/ 10.13039/501100011033 Unión Europea (Chist-Era Program); KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MCIN/AEI/10.13039/ 501100011033/ FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la Junta de Comunidades de Castilla La Mancha, y el Fondo Europeo de Desarrollo Regional FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; y RED2024-154240-T financiado por MICIU/AEI/10.13039/501100011033.

Referencias

[1] M. M. Issa, M. Aljanabi, and H. M. Muhialdeen, "Systematic literature review on intrusion detection systems: Research trends, algorithms, methods, datasets, and limitations," *Journal of Intelligent Systems*, vol. 33, no. 1, p. 20230248, 2024.

[2] A. Alarcón, O. León, J. Pegueroles, J. Doménech *et al.*, "Técnicas de machine learning para la detección de ciberataques en redes IoT médicas," *XVIII Reunión Española sobre Criptología y Seguridad de la Información: XVIII RECSI, León 23-25 octubre 2024*, pp. 25–30, 2025.

[3] M. Macas, C. Wu, and W. Fuertes, "Adversarial examples: A survey of attacks and defenses in deep learning-enabled cybersecurity systems," *Expert Systems with Applications*, vol. 238, p. 122223, 2024.

[4] S. M. Alhidaifi, M. R. Asghar, and I. S. Ansari, "A survey on cyber resilience: Key strategies, research challenges, and future directions," *ACM computing surveys*, vol. 56, no. 8, pp. 1–48, 2024.

[5] N. Wiener, *The human use of human beings: Cybernetics and society*. Cambridge, MA: The Riverside Press, 1950.

[6] A. L. Gibbs and F. E. Su, "On choosing and bounding probability metrics," *International Statistical Review*, vol. 70, no. 3, pp. 419–435, 2002.

[7] C. Villani, *Optimal transport: Old and new*. Berlin: Springer, 2008, vol. 338.

[8] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 eighth ieee international conference on data mining*. IEEE, 2008, pp. 413–422.

[9] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani *et al.*, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." *ICISSp*, vol. 1, no. 2018, pp. 108–116, 2018.

[10] C. M. Braga, M. A. Serrano, and E. Fernández-Medina, "Towards a methodology for ethical artificial intelligence system development: A necessary trustworthiness taxonomy," *Expert Systems with Applications*, vol. 286, p. 128034, 2025.

[11] C. M. Braga, Ángel Suarez-Bárcena, M. A. Serrano, and E. Fernández-Medina, "TrustAIoT: A framework for building trustworthy AIoT platforms," *Internet of Things*, vol. 34, p. 101751, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660525002641

[12] L. Devroye, L. Györfi, and G. Lugosi, *A probabilistic theory of pattern recognition*. New York: Springer Science & Business Media, 2013, vol. 31.

[13] Y. Zhou, X. Xu, J. Song, F. Shen, and H. T. Shen, "Msflow: Multiscale flow-based framework for unsupervised anomaly detection," *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

Modeling and Training Robust Intrusion Detection Systems Using Deep Neural Networks: A Synthesis of 'Best Practices'

1st Victor Manuel González-Gorrín 2nd Josep Prieto-Blázquez

Abstract*—Training effective deep neural networks (DNNs) presents significant challenges. Without adhering to specific best practices, practitioners face issues such as convergence failure, poor generalization to new data, overfitting or underfitting, and substantial waste of computational resources. Previously, we proposed a hybrid two-stage solution machine learning (ML) detection-method approach to detect metamorphic malware. This hybrid approach provided promising experimental results, and further recommendations were made on how to enhance and explore new research avenues. The novelty arose from the combination of two distinct sets of features that enhanced the final outcome more effectively than if they were applied separately. In stage 1, sequences of opcodes are used; a discrete hidden Markov Model (dHMM) validates the input dataset; while stage 2 uses the portable executable (PE) sections from executable files as features for a random forest (RF). At the time, a question arose about the implications of replacing the RF with a DNN and how to proceed using best practices. This led to the development of this article, which emphasizes the critical need to adopt robust methodologies, such as encompassing data handling, model architecture selection, optimization strategies, and evaluation protocols that ensure the development of a reliable and efficient DNNs model for metamorphic malware detection. It needs to be stated that this contribution to the literature is a methodological synthesis of best practices for IDS design using DNNs, rather than a performance evaluation. Finally, this synthesized paper was made possible thanks to a systematic' literature review (SLR) previously conducted by the authors, covering nearly 1,000 publications between 2020 and 2025, in addition to empirical-evidence provided from an original article concerning a hybrid Machine Learning (ML) Intrusion Detection System (IDS) for Metamorphic Malware – an Adversarial Malware Attack (AMA) – submitted for publication by the authors.

***Index Terms*—Deep Learning, Neural Networks, Best Practices, Overfitting, Generalization, Optimization, Hyperparameter Tuning.**

I. Introduction

*An intrusion detection system (**IDS**) built on a multi-layer deep neural network (**DNN**) involves collecting and preprocessing data, designing a neural network (**NN**), training the model with labeled data, evaluating its performance, and deploying it for real-time detection* [1], [2]. This approach leverages the power of NNs to automatically learn complex patterns and improve detection accuracy; however, it requires large datasets and computational resources. Compared to other methods, DNNs-based IDS offers high accuracy, adaptability, scalability, a reduced need for manual feature engineering, robustness to evasion techniques, versatility in handling multiple data types, and improved detection of unknown adversarial malware attacks **(AMA)** [3]–[6], [14]. These advantages make deep learning (DL) a powerful tool in the fight against evolving and sophisticated AMA threats. However, in academia, the epistemology of AMAs in the context of an IDS has not been fully grasped. Using machine intelligence (MI), many attempts have been made to reproduce modeling techniques and methods within IDSs that can properly track, trace, and prevent AMAs from reappearing. Moreover, there seems to be a lack of comprehensive and cohesive studies of the literature '*on best practices*' regarding deep-learning-based (**DLB**) IDSs, on which existing scholars can rely to pursue future research and broaden the field of Information Security and Networks (**ISN**) research. Therefore, the primary aim of this paper is to inform an audience on how to employ best practices when building a robust DL multi-layer network, with the ultimate goal of developing a model that achieves high performance and is reliable, efficient, and capable of operating effectively in real-world, dynamic environments, despite some infrequent yet minor constraints. In other words, this contribution is a methodological synthesis of best practices for IDS design using DNNs, rather than a performance evaluation.

The paper is structured as follows. Part II illustrates the typical workflow of a robust DNN IDS system without necessarily emphasizing best practices for its design and implementation. Part III identifies the best practices recommended based on empirical evidence that new research should primarily focus on when trying to design and build a DNN IDS that is mathematically robust. Part IV highlights common mathematical pitfalls that should be avoided when building a robust DNN IDS, followed by a short conclusion and further recommendations in Part V.

II. Typical Workflow of a Robust IDS System

A robust DNN training workflow is a structured and iterative process that moves from defining the problem to continuous monitoring. It emphasizes systematic experimentation, robust data handling, and thorough evaluation to ensure reliability and efficiency. Its development will take different phases and shall require months for full implementation. More formally,

A. *Phase 1: Problem Definition and Data Acquisition*

The process begins with understanding the business problem and acquiring the necessary data.

- Define the Goal and Metrics: Clearly define what problem the model needs to solve and how success will be measured (e.g., human-interpretable metrics like accuracy, precision, and recall).
- Data Acquisition: Collect raw data from relevant sources. High-quality data are the foundation of a successful model.

B. *Phase 2: Data Preparation and Preprocessing*

Real-world data are often noisy and unorganized. This phase makes the data suitable for model training.

- Exploratory Data Analysis (**EDA**): Visualize data distributions, identify outliers, find inconsistencies (e.g., mislabeled examples), and build a "*mental model*" or "*mind map*" of the data.
- Data Cleaning and Transformation: Handle missing values, remove duplicates, and apply necessary transformations (e.g., normalization, one-hot encoding for categorical variables).
- Splitting Datasets: Divide the data into three sets. They are:
  - The training set is used to train the model.
  - The validation set is used to tune hyperparameters and check for overfitting during training.
  - The test set is always held aside until the very end to evaluate the performance of the final trained model on unseen data.
- Data Augmentation (Optional but Recommended)

C. *Phase 3: Model Engineering and Training*

This is the core of the workflow, where the model architecture is defined and optimized. The key objectives of this phase should include the following:

- Establish Baselines: Before building a complex DNN, create simple models (e.g., a simple heuristic or shallow machine learning (**ML**) model) to set a performance lower bound.
- Architecture Selection: Choose a suitable network architecture based on the problem type (e.g., convolutional neural networks (**CNNs**) for image tasks, recurrent neural networks (**RNNs**)/Transformers for sequence data).
- Initial Training and Sanity Checks: First, train a toy model or a simple version on a small batch of data. Ensure that the loss starts at an expected value (given random initialization) and that the model can overfit this small dataset. This verifies that the pipeline is working correctly.
- Overfit-Regularize Cycle:
  - Overfit: Increase model capacity (add layers/neurons) and train until it achieves a high training metric, even if the validation metric suffers.
  - Regularize: Tame the model's complexity to improve validation performance using techniques like dropout, weight decay, or early stopping (tracking the validation metric to stop training when performance degrades).
  - Hyperparameter Tuning: Systematically adjust parameters like learning rate (a crucial step), batch size, and optimizer (e.g., Adam [8], RMSProp [9]).

D. *Phase 4: Evaluation and Deployment*

Once a satisfactory model is developed, it is formally evaluated and prepared for real-world use.

- Final Evaluation: Evaluate the optimized model on the previously untouched test set using the predefined metrics to obtain an unbiased measure of its performance.
- Deployment: Integrate the model into the production environment (e.g., via APIs) so that it can make predictions on new real-world data.

E. *Phase 5: Monitoring and Maintenance*

The workflow is continuous, as models degrade over time due to changing data patterns (concept/data drift).

- Continuous Monitoring: Track key metrics in real-time in the production environment to detect performance degradation or anomalies - *e.g.; avoid data drift and concept drift.*
- Feedback Loops: Collect user feedback and log live data to update the training dataset - e.g.; set up automated alerting pipelines and a plan for periodic retraining with fresh data.
- Retraining and Improvement: Periodically retrain the model with fresh data to ensure that it remains accurate and relevant over time.

III. Best Practices for Modeling and Training Deep Neural Networks

Best practices for modeling and training DNNs should focus on mitigating common issues such as overfitting, slow convergence, and poor generalization, some of which shall be clearly detailed next. Adherence to these practices ensures that the models are robust, efficient to train, and perform reliably in production environments.

A. *Data Preparation and Handling*

High-quality data are the foundation for effective deep learning. More specifically,

- Normalize Input Data: Scale numerical input features to have a zero mean and unit variance (e.g., using z-score normalization). This ensures that all features contribute equally to the loss calculation and help gradient-based optimization algorithms to converge faster.
- Handle Data Imbalance: If one class heavily outweighs the others, the model may become biased. Use techniques like oversampling the minority class, undersampling the majority class, or using weighted loss functions.
- Data Augmentation: Systematically create new training examples by applying relevant transformations to existing data (e.g., rotation, scaling, cropping images; pitch shifting audio). This increases the size of the dataset and improves the robustness and generalization of the model.
- Robust Train/Validation/Test Split: Maintain a strict, non-overlapping separation between training, validation, and

test datasets. The test set must remain untouched until the final evaluation to prevent "data leakage" and ensure an unbiased measure of performance.

Things to avoid:

- preprocessing before splitting the data.
- using expected data instead of historical data.
- using duplicated data.
- removing missing values from the dataset.

B. *Model Architecture and Initialization*

Choosing the right structure and starting point is crucial for stable training. That is,

- Choose an Appropriate Architecture: Select an architecture that is suitable for the data type.
- Use Heuristics for Weight Initialization: Avoid all zeros or random initialization with simple distributions.
- Favor ReLU Activations.
- Use Batch Normalization: Insert batch normalization layers between linear/convolutional layers and activation functions.

Things to avoid:

- complexity but rather starting with a baseline model.
- resampling the same test set causing overfitting instead of adding a validation set first.

C. *Optimization and Training Dynamics*

Effective training requires careful selection and tuning of the optimization process. That is,

- Employ Adaptive Optimizers: Optimizers like Adam [8] or RMSProp [9] are generally recommended over basic stochastic gradient descent (SGD) [10] because they adapt the learning rate for each parameter, leading to faster convergence and often better results.
- Tune the Learning Rate (Crucial): The learning rate is perhaps the most important hyperparameter.
  - Too high: Training diverges or oscillates wildly.
  - Too low: Training is slow and may get stuck in a poor local minimum.
- Use techniques like learning rate schedules (decay over time) or learning rate finders (from libraries like Keras Tuner or PyTorch Lightning) to find an optimal rate.
- Implement Early Stopping: Monitor the model's performance in the validation set during training.

Things to avoid:

- narrowed optimizations instead of searching for the best configuration.

D. *Regularization and Preventing Overfitting*

A robust model generalizes well to new data. These techniques enforce simplicity and prevent memorization of the training set. That is,

- Dropout: Randomly drop a percentage of neurons during training iterations. This forces the network to be less reliant on any single path and improves robustness.
- L1/L2 Regularization (Weight Decay): Add a penalty term to the loss function based on the magnitude of the model weights, which discourages overly large weights.
- Monitor Both Training and Validation Loss: The gap between the training loss and the validation loss is your primary indicator of overfitting. A large gap indicates that the model is overfitting and requires more regularization.

E. *Experiment Tracking and Reproducibility*

Maintain a systematic approach to research and development. That is,

- Log Everything: Use experiment tracking tools (e.g. Weights & Biases, MLflow, TensorBoard) to log hyperparameter choices, data versions, and metrics (loss, accuracy, F1-score) for every run.
- Establish a Clear Baseline: Before deploying a complex DNN, compare its performance against simpler, traditional ML models or even simple heuristics to ensure that the deep learning approach offers tangible value.
- Code Version Control: Use Git to manage all code changes, ensuring that every experiment can be reproduced exactly.

Using a mind map (or conceptual overview), "Fig. 1" synthesizes best practices for training DNNs and includes recommendations commonly reported in the literature, followed by a breakdown of best practices provided in the following key stages.

IV. Avoiding Common Mathematical Pitfalls

In no particular order, they are:

- **Non-saturating activation functions**: Unlike tanh and sigmoid, ReLU and its variants do not saturate for positive values, which prevents gradients from vanishing during back-propagation. A summary of best practices is illustrated in Table I.

TABLE I
Non-saturating activation (**NSA**) functions

NSA Function	Objective
ReLU	Which has a constant derivative of 1 for positive inputs, preventing the gradient from shrinking in that range. It is simple and computationally efficient. Its main drawback is the "dying ReLU"[1] problem, where neurons can become permanently inactive for negative inputs.
Leaky ReLU	Addresses the dying ReLU problem by introducing a small, non-zero gradient for negative inputs.
Exponential Linear Unit (**ELU**)	Like Leaky ReLU, ELU also helps with the dying ReLU problem and often leads to faster learning.

[1]If a neuron's weights are adjusted in such a way that its input is always negative, that neuron will forever output 0, its derivative will always be 0, and it will stop learning permanently.



Fig. 1. **Mind map showing recommended best practices for Modeling and Training Robust IDSs Using DNNs**. Adherence to these practices ensures that the models are robust, efficient to train, and perform reliably in production environments.

"Fig. 2", "Fig. 3" and "Fig. 4" show the behavior of the ReLU, Leaky ReLU, and the ELU NSA functions, respectively. However, note that the derivative of the ReLU activation function is undefined in $x = 0$. Although the function is composed of two linear pieces (a horizontal line and a line with a slope of 1), the "kink" at $x = 0$ makes the overall function non-linear. This non-linearity is essential for a neural network to learn complex patterns and approximate non-linear functions of the input data, which linear functions alone cannot do. The main mathematical drawback of ReLU is the "*dying ReLU*" problem. If a neuron's weights are adjusted in such a way that its input is always negative, that neuron will forever output 0, its derivative will always be 0, and it will stop learning permanently. Variants like Leaky ReLU or ELU address this by providing a small, non-zero slope for negative inputs.

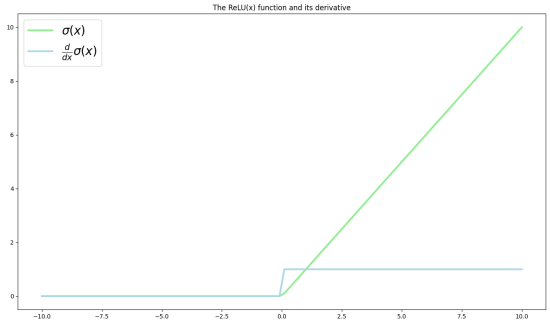


Fig. 2. The ReLU activation function generated in Python. X axis shows the input value to the neuron. Y axis shows the output value of the neuron.

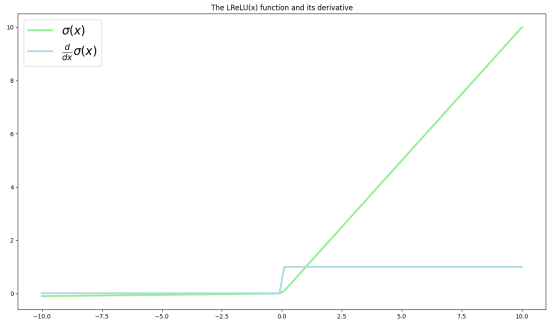


Fig. 3. The Leaky ReLU activation function generated in Python. X axis shows the input value to the neuron. Y axis shows the output value of the neuron.

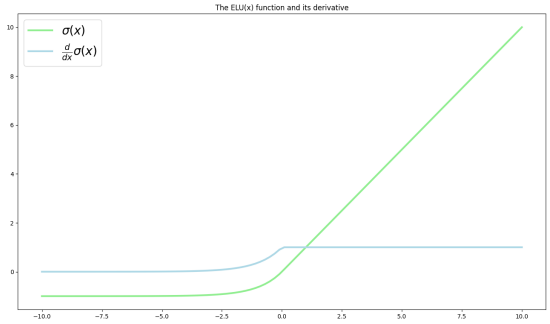


Fig. 4. The ELU activation function using Python. X axis shows the input value to the neuron. Y axis shows the output value of the neuron.

- **Choose a proper weight initialization scheme**: Appropriate weight initialization helps stabilize gradient flow at the start of training. Some useful techniques are the following.
  - He [11] initialization: Best for networks with ReLU activation functions.
  - Glorot (Xavier) [12] initialization: Suitable for networks with sigmoid or tanh activation functions.
- Stabilize the network using batch normalization: Batch Normalization standardizes the inputs to each layer for every mini-batch. This practice offers several benefits.
  - Reduces internal covariate shift: Stabilizes the input distribution to each layer, allowing for higher learning rates and faster convergence.
  - Regularization effect: Acts as a form of regularization, reducing the need for dropout in some cases.
  - Mitigates gradient problems: Helps control the magnitude of gradients, making the network more robust against vanishing and exploding issues.
- **Adopting a modern architecture**: Certain architectures are specifically designed to address the challenges of deep networks.
  - Residual Networks (**ResNets**): Use "*skip connections*" to add the output of a previous layer to a later one. This creates a "*highway*" for gradients to flow directly through the network, which effectively combats degradation and vanishing gradient problems.
  - Long Short-Term Memory (**LSTM**) Networks: Specifically designed to handle vanishing gradients in recurrent networks by using gated mechanisms to regulate the flow of information.
- **Control and optimize the training process by utilizing advanced optimizers**: Move beyond basic Stochastic Gradient Descent (**SGD**) to more sophisticated algorithms that adapt the learning rate during training. For example:
  - Adam [8]: A combination of Momentum and RMSProp, Adam is often the default choice for its efficiency and good performance on a variety of tasks.
  - RMSProp [9]: Adapts the learning rate based on a moving average of squared gradients, helping to accelerate training.
- **Regularize to avoid overfitting**: Deep networks have a large capacity to memorize training data, so it is essential to use regularization to encourage generalization.
  - L1 and L2 regularization [13]: Add a penalty to the loss function based on the magnitude of the weights of the network, which discourages high weight values.
  - Dropout: Randomly "*drops*" neurons and their connections during training, which forces the network to learn more robust features and prevents co-adaptation among neurons.

  - Data Augmentation: Artificially increases the size and diversity of the training dataset by applying transformations (e.g., rotation, scaling, flipping) to existing data.
- **Use transfer learning**: If a similar problem has already been solved by a large, pre-trained network, use that as a starting point. By fine-tuning the model's top layers on your specific data, you can achieve excellent performance with much less training time and data.
- **Monitor training and use early stopping**
  - Monitor performance: Keep track of training and validation metrics (such as loss and accuracy) to diagnose issues such as underfitting or overfitting.
  - Early stopping: Stop training when performance on a validation set begins to degrade. This prevents the model from overfitting and often results in the model with the best generalization performance.

V. Conclusion

This conference paper highlights the need to employ best practices when building an IDS based on DNNs and how to avoid common mathematical pitfalls in the algorithms for training and testing these systems. Initially, the paper highlights the workflow to be followed, emphasizing systematic experimentation, robust data handling, and thorough evaluation that ensures reliability and efficiency. Then, the paper dives into the specifics of best practices that one should follow during the modeling and training of the IDS based on DNNs. The concepts and techniques introduced are high level at best, but they are part of further developments and research papers to follow in the future from the authors to complement this brief introduction. It needs to be stated that this contribution to the literature is a methodological synthesis of best practices for IDS design using DNNs, rather than a performance evaluation. Finally, this synthesized paper was made possible thanks to a systematic' literature review (SLR) previously conducted by the authors, covering nearly 1,000 publications between 2020 and 2025, in addition to empirical-evidence provided from an original article concerning a hybrid Machine Learning (**ML**) Intrusion Detection System (**IDS**) for Metamorphic Malware – an Adversarial Malware Attack (**AMA**) – submitted for publication by the authors.

Acknowledgment

This work was supported by the Spanish Ministry of Science and Innovation through the project PID2021-125962OB-C31 "SECURING/CYBER" and PID2024-156914OB-C41 "SAFE/CYBER". Additional funding was provided by the ARTEMISA International Chair in Cybersecurity (C057/23) and the DANGER Strategic Project of Cybersecurity (C062/23), both funded by the Spanish National Institute of Cybersecurity through the European Union NextGenerationEU and the Recovery, Transformation and Resilience Plan.

References

[1] Victor Manuel González-Gorrín and J. Prieto-Blázquez, "Machine Intelligence Approaches for Preventing Adversarial Malware Attacks in Intrusion Detection Systems: A Systematic Review," unpublished.

[2] Victor Manuel González-Gorrín, J. Prieto-Blázquez and J. Arnedo-Moreno, "A Hybrid Machine Learning Method for Detecting Metamorphic Malware," in 39th International Conference on Advanced Information Networking and Applications (AINA 2025), pp. 297–307, Springer.

[3] D. Vasan, M. Alazab, S. Wassan, B. Safaei and Q. Zheng, "Image-Based Malware Classification Using Ensemble of CNN Architectures (MCEC)," in Computers and Security 92, 101748, 2020.

[4] J. Hemalatha, S.A. Roseline, S. Geetha, S. Kadry and R. Damaševičius, "An Efficient Densenet-Based Deep Learning Model for Malware Detection," in Entropy 23(3), 2021.

[5] C. Li, Q. Lv, N. Li, Y. Wang, D. Sun, D. and Y. Qiao, "A Novel Deep Framework for Dynamic Malware Detection Based on API Sequence Intrinsic Features," in Computers and Security 116, 2022.

[6] A.M. Ali, F. A. Ghaleb and F. Saeed, "AEDGAN: A Semi-Supervised Deep Learning Model for Zero-Day Malware Detection," in International Journal of Advanced Computer Science and Applications 16(3), 2025, pp.759—769.

[7] Victor Manuel González-Gorrín and J. Prieto-Blázquez, "A Hybrid Machine Learning Intrusion Detection Method for Metamorphic Malware," unpublished.

[8] P. Diederik, B. Kingma and B. Jimmy, "Adam: A Method for Stochastic Optimization," in International Conference on Learning Representations, 2015.

[9] T. Tieleman, and G. Hinton, "Lecture 6.5-RMSProp: Divide the Gradient by a Running Average of its Recent Magnitude," in COURSERA: Neural Networks for Machine Learning, 4(2), 2012, pp.26–31.

[10] H. Robbins and S. Monro, "A stochastic approximation method," in Annals of Mathematical Statistics 22(3), 1951, pp. 400–407.

[11] K. He, X. Zhang, S. Ren and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1026–1034.

[12] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feed-forward neural networks," in Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS), 2010.

[13] R. Tibshirani, "Regression shrinkage and selection via the lasso," in Journal of the Royal Statistical Society: Series B (Methodological), 58(1), pp. 267–288, 1996.

[14] J. Zhang and X. Wang and C. Li and Q. Zhang and G. Yang and X. Li and F. Cui and R. Gu and P. Qi and S. Liu, "CNNRes-DIndRNN: A New Method for Detecting TLS-Encrypted Malicious Traffic," in Future INternet, 18(1):8, 2026.

Maritime cybersecurity: protecting digital seas

Ferney MARTÍNEZ OSSA
Grupo de Seguridad y Auditoría GSyA - PRODIN
Universidad de Castilla-La Mancha - COTECMAR
Ferney.Martinez@alu.uclm.es

Luis Enrique SÁNCHEZ
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
Luise.Sanchez@uclm.es

Antonio SANTOS-OLMO
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
Antonio.Santosolmo@uclm.es

David G. ROSADO
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
David.GRosado@uclm.es

Eduardo FERNÁNDEZ-MEDINA
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
Eduardo.Fdezmedina@uclm.es

***Resumen*—Este es un resumen extendido del artículo publicado en [1]. El ámbito marítimo es un ecosistema ciberfísico interconectado —buques, puertos, redes informáticas, comunicaciones— con una superficie de ataque que crece más rápido que los procesos de adopción de contramedidas sectoriales. El trabajo sintetiza un estudio publicado que revisa sistemáticamente el estado del arte y, a partir de las brechas identificadas, plantea como solución un marco integral de gestión de ciberseguridad marítima. La revisión cuantifica tendencias (metodologías, marcos, modelos, técnicas y procesos), perfilando las debilidades persistentes (gobernanza, cultura, taxonomía e idiosincrasia marítima) y consolida lineamientos normativos de diferentes estándares, normas y guías para el sector. La propuesta se articula en cuatro bloques (Etapas, Soporte Técnico, Ciclo de Gestión y Soporte Documental), con subcomponentes diferenciales (p. ej., Prevención por misión, Cultura de seguridad marítima, Gestión de escenarios, Redes dedicadas). El marco mejora la trazabilidad entre activos–amenazas–controles y direcciona la priorización de diferentes acciones.**

***Palabras clave*—Ciberseguridad marítima, gestión de riesgos, Marco de Ciberseguridad**

I. Introducción

El dominio marítimo constituye una red crítica de buques destinados a operaciones militares, transporte de carga y pasajeros. La digitalización acelerada ha transformado al buque en un centro de datos móvil que integra sistemas de Internet de las Cosas (IoT) y controla equipos remotos. Si bien esta evolución eleva la eficiencia, también amplía la superficie de ataque en sistemas de Tecnología de Operación (OT) y Tecnologías de la Información (IT), exponiendo procesos críticos a errores y ciberamenazas.

La creciente sofisticación de los ataques y actividades delictivas, como el ransomware, ha obligado a priorizar la seguridad en subsistemas críticos como el Sistema de Identificación Automática (AIS) y el Sistema de Información y Visualización de Cartas Electrónicas (ECDIS). En este escenario, el factor humano y la integridad de las redes emergen como elementos transversales de vulnerabilidad.

En respuesta, organismos internacionales han emitido marcos de referencia esenciales: la Organización Marítima Internacional (IMO), con sus directrices de gestión de riesgos vigentes desde 2021; el Instituto Nacional de Estándares y Tecnología (NIST), con perfiles específicos para buques; y la Agencia de la Unión Europea para la Ciberseguridad (ENISA), con guías para puertos.

El objetivo principal de este artículo es presentar el marco POSEIDON, una solución integral diseñada para operacionalizar la gestión de ciberseguridad marítima. El trabajo sintetiza una revisión sistemática del estado del arte para identificar brechas de gobernanza y, basándose en ellas, propone una arquitectura que facilita la trazabilidad entre activos, amenazas y controles, asegurando la alineación con la normativa internacional vigente.

II. Metodología de revisión sistemática

El protocolo de revisión sistemática fue diseñado para asegurar rigor, trazabilidad y reproducibilidad de la síntesis. Inició con las preguntas de investigación centradas en identificar los marcos, métodos y evidencias empíricas relevantes para la ciberseguridad marítima; se definió un plan previo de búsqueda, criterios de inclusión/exclusión y un proceso de identificación aplicado en bases de alto prestigio académico (ACM, IEEE Xplore, ScienceDirect, Springer, Scopus) y fuentes institucionales.

La extracción de datos se realizó con una plantilla estandarizada y la clasificación temática se apoyó en categorías predefinidas. La calidad de los estudios y el riesgo de sesgo se evaluaron con un esquema Evaluación de Calidad/Riesgo de Sesgo (QA/ROB) y doble revisión, cuyos umbrales y procedimientos se detallan en el capítulo 3 de [1]. Finalmente, se efectuó un análisis de sensibilidad y una agregación ponderada por calidad para robustecer las conclusiones. Los aspectos a destacar:

- Búsqueda en ACM, IEEE Xplore, ScienceDirect, Springer, Scopus y fuentes institucionales; cadena focalizada in maritime cybersecurity / framework / risk management.
- Identificados: 157 trabajos; selección final: 42 (inclusión: foco marítimo, revisión por pares u organismo oficial; exclusión: duplicidad, desalineación, antigüedad pre-2016).
- Clasificación: Metodologías (11), Marcos (7), Modelos (7), Técnicas (6), Procesos (5) y Otros (6).
- Bibliometría (VOSviewer): dominios predominantes (cybersecurity, ships, risk assessment, network security) y concentración geográfica en EE. UU., Reino Unido, Noruega, Alemania, Francia, España, Bélgica, Países Bajos, Corea del Sur, China, Grecia y Finlandia.

Luego del proceso riguroso de la revisión sistemática, se establecieron los hallazgos de la revisión en dos planos complementarios: (i) las tendencias observadas en la literatura reciente y vacíos persistentes que condicionan la práctica, y (ii) un conjunto de características deseables para orientar el diseño y la evaluación de propuestas en el dominio marítimo.

II-A. Tendencias y brechas

La evidencia recopilada muestra un patrón transversal: la adaptación parcial de marcos transversales al entorno marítimo, la terminológia y la focalización en casos específicos. Este contexto limita la madurez de las soluciones y explica la disparidad entre capacidades propuestas y necesidades operacionales a bordo y en tierra. Las principales tendencias y brechas identificadas del dominio marítimo son:

- Adaptación de marcos transversales (NIST, ISO 27000, IEC 63154, MITRE ATT&CK, entre otros) al dominio marítimo.
- Brechas persistentes: Se concentran en la falta de especificidad y operativización de los marcos existentes. Estas deficiencias clave incluyen una gobernanza que a menudo es genérica y se enfoca en controles técnicos. De igual forma, la cultura de la ciberseguridad marítima y la idiosincrasia marítima están escasamente operativizadas, ya que los marcos carecen de una metodología clara para fomentar y evaluar el factor humano.
- Equipos y redes críticas: La creciente digitalización ha expuesto una serie de equipos y redes críticas que son esenciales para la seguridad y operación de la unidad marítima. La principal debilidad en estos sistemas de a bordo, es la falta de mecanismos completos de protección que trasciendan los controles perimetrales básicos.
- Evidencia empírica: Si bien existen reportes anecdóticos de incidentes de ciberseguridad en el sector marítimo, la evidencia empírica sigue siendo incompleta. Una limitación crítica es la ausencia de repositorios formalizados de lecciones aprendidas que permitan una trazabilidad técnica-operativa clara. Esta falta de datos estructurados dificulta la definición y uso de métricas comparables.

Los Rasgos Consolidados identificadas en la literatura y en la práctica del sector marítimo que se derivan de este análisis son: SRA (Sectoral Risk Analysis), MRM (Maritime Risk Management), MSG (Maritime Security Governance), MSC (Maritime Safety Culture), IRP (Incident Response Plan), CS (Cyber Security), AMS (Aimed at Maritime Sector), MT (Mission Tailored), MI (Mission Integration), DAV (Directed to Autonomous Vessels), y DIS (Directed to Industrial Sector). Sin embargo, la evidencia analizada muestra que las soluciones actuales presentan coberturas parciales en aspectos clave como la Gobernanza Marítima de Seguridad (MSG), la Cultura de Seguridad Marítima (MSC), el Ajuste a la Misión (MT) y la Integración de la Misión (MI).

III. Propuesta: el marco POSEIDON

A partir de la síntesis establecida en la revisión sistemática (tendencias y brechas) y del análisis de calidad aplicado a los 42 estudios incluidos, sumado a los requisitos regulatorios (IMO) y los marcos de referencia (ISO, ENISA, NIST, IEC) y las restricciones operativas de entornos de Tecnologías de la Información y Tecnologías Operativas (IT/OT) a bordo y en tierra, se propone POSEIDON (Procedures, Operations and Standards for the detection of integral Events and the Development of naval operating capacity): un marco dirigido al sector marítimo que busca operacionalizar esos hallazgos y cerrar carencias en gobernanza, cultura, taxonomía e idiosincrasia marítimas.

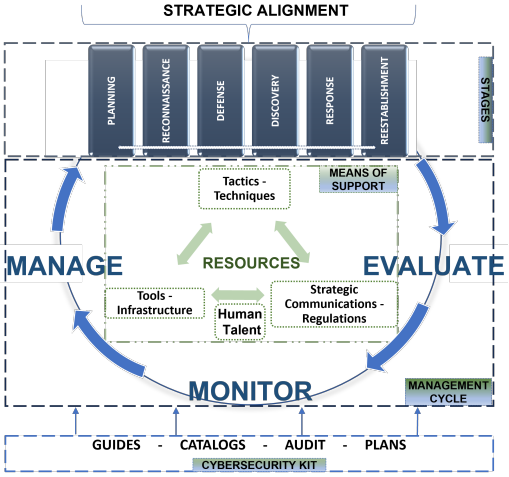


Figura 1. Propuesta de primer nivel de POSEIDON (de [1])

El marco se plantea con tres objetivos: (1) proporcionar un glosario especializado y un catálogo común para modelar activos, amenazas y controles; (2) Brindar una guía a las decisiones de riesgo, inversión y trazabilidad a IMO/ISO/ENISA/NIST/IEC; y (3) habilitar gestión continua mediante indicadores y procesos de verificación. El marco se organiza en cuatro bloques complementarios que abarcan desde la planificación por misión y la defensa en profundidad hasta la monitorización, la respuesta y la continuidad. Cada bloque define principios, artefactos y resultados esperados.

La *Figura 1* presenta la vista general de POSEIDON y su composición en cuatro componentes —Etapas, Soporte Técnico, Ciclo de Gestión y Soporte Documental—. Además, la figura muestra la manera en que se proyectan sus relaciones clave, conectando cada bloque con las brechas identificadas en la revisión sistemática. El detalle de cada bloque se presenta a continuación.

III-A. Bloque 1 — Etapas

Este bloque define el ciclo operativo del marco propuesto —Planificar, Reconocer, Defender, Descubrir, Responder, Restablecer— y establece para cada etapa las entradas/salidas, los responsables, hitos y criterios de paso. En esta instancia se incorporan aspectos propios del ámbito marítimo como la planificación por misión y la defensa en profundidad, integrando estrategias metodológicas operacionales para ajustar continuamente el plan o perfilarlo de mejor manera acorde a la misión operacional de la unidad marítima. Entre los resultados esperados se tienen: matriz de misión, escenarios priorizados, mapa de activos/amenazas y la identificación de controles y experimentos, definición de perfiles del talento humano. La Tabla I detalla los elementos diferenciales de este bloque.

Tabla I
ELEMENTOS DIFERENCIALES DEL BLOQUE 1 DE POSEIDON

Elemento	Descripción
Prevención por misión	Adopta las posibles estrategias, entrenamientos y cooperaciones externas condicionadas al objetivo operativo (comercial, humanitario, vigilancia, control de área, militar).
Perfil del Personal	Se identifican y establecen los roles/competencias para operar sistemas críticos acorde a las misiones operacionales; se establecen los criterios de idoneidad.
Lecciones aprendidas	Integración del registro y análisis del ante/post de incidentes, conectividad portuaria y requisitos de evidencia.
Gobernanza de seguridad	Se prioriza en el marco propuesto aspectos estratégicos como las políticas, responsabilidades, auditoría y presupuesto que guían adquisiciones y defensas.
Medidas con terceros	Procesos y actividades que permiten el intercambio de información, realizar una evaluación conjunta de eventos, verificación periódica de la cadena de suministro y una respuesta coordinada.
Clasificación de datos y misiones	Se realizan mapeo de flujos y clasificación de la criticidad por misión para establecer una prioridad de controles y continuidad operativa.
Inteligencia de incidentes	Realimenta identificación de activos críticos, amenazas y vulnerabilidades.
Cultura de seguridad marítima	Se plantean procesos adicionales que van más allá de la formación, con ciclos de retroalimentación a bordo y en tierra.
Monitorización de seguridad marítima y protección de medios digitales	Se proponen mecanismos para la vigilancia de las conexiones satelitales, enlaces de datos tácticos y radios digitales.
Gestión de escenarios	A partir de la información operativa, el marco de trabajo inicia los procesos para la parametrización de escenarios de defensa/ataque para contar con una respuesta graduada con evidencia para procesos posteriores.
Capas de defensa en profundidad	Se identifican las actividades que permiten establecer las barreras redundantes y escalonadas, para contar con degradación tolerable y contención por dominios (redes IT/OT, sistemas de puente, carga, propulsión).
Redes dedicadas	Clasificación/segmentación (datos, controles, Comando y Control, servicios portuarios), aislamiento y recuperación segura.

III-B. Bloque 2 — Soporte Técnico

Es el eje funcional del marco POSEIDON. Al interior del marco, este bloque concentra las capacidades y medios que habilitan las etapas (tácticas, técnicas, herramientas, infraestructura, comunicaciones estratégicas, regulación y talento humano). Este componente asegura que todas las fases del Ciclo Operativo (Bloque 1) cuenten con los recursos necesarios para la prevención, detección y respuesta, proveyendo la línea de base técnica y la arquitectura de referencia necesaria. Además, garantiza la alineación con estándares o metodologías esenciales como MITRE ATT&CK, NIST CSF, ISO 27000, IEC 62443 e IMO, estableciendo perfiles de capacidad y playbooks. Los elementos clave de este soporte técnico se detallan a continuación en la tabla II.

III-C. Bloque 3 — Ciclo de Gestión

Es el componente que garantiza la sostenibilidad y la mejora continua del marco POSEIDON, cerrando el bucle de gobernanza mediante la secuencia Dirigir–Monitorear–Evaluar. Este módulo establece los procesos formales para la definición de objetivos de seguridad, la gestión de riesgos y la monitorización continua a través de indicadores clave. Su función esencial es asegurar la rendición de cuentas, establecer las cadencias operacionales y proveer revisiones ejecutivas periódicas, logrando la alineación estratégica con estándares internacionales críticos. Además, este bloque incluye una extensión novedosa de estos marcos regulatorios hacia los sistemas críticos de navegación del buque, asegurando una evaluación continua de los activos marítimos. El detalle de los elementos, procesos y resultados de este bloque se presentan en la Tabla III.

Tabla II
COMPONENTES DEL BLOQUE 2 – SOPORTE TÉCNICO

Elemento	**Descripción**
Elementos de Soporte	Define perfiles de capacidad con niveles de madurez, requisitos de interoperabilidad, catálogos de controles y playbooks.
Alineación con Estándares	Mapea con MITRE ATT&CK, NIST CSF, ISO 27000, IEC 62443 e IMO.
Alcance Operacional	Provee medios tecnológicos, normativos y humanos para prevención, detección, respuesta y continuidad (alineado con MITRE ATT&CK, IMO y ENISA).
Resultados Esperados	Desarrollo de la línea de base técnica, la arquitectura de referencia para redes IT/OT, los kits de pruebas necesarios y guías de hardening.

Tabla III
COMPONENTES DEL BLOQUE 3 – CICLO DE GESTIÓN (DIRIGIR–MONITOREAR–EVALUAR)

Elemento	**Descripción**
Procesos y Cadencias	Se establecen las cadencias (semanal/mensual/trimestral), los mecanismos de escalado, y los criterios de aceptación de riesgo.
Alineación Estratégica	Se realiza la alineación con ISO 27001/IEC 62443 y la gestión de continuidad. En este nivel se espera: la consolidación del cuadro de mando, la generación de informes de riesgo, implementar los planes de mejora y desarrollar las evidencias de cumplimiento.
Extensión de Estándares	Se realiza una extensión de estandares estableciendo objetivos sectoriales, planes de respuesta, una monitorización de sistemas críticos del buque (AIS, cartas electrónicas, tráfico y comunicaciones) con una evaluación continua con ejercicios e indicadores.
Resultados Esperados	Consolidación del cuadro de mando, la generación de informes de riesgo, implementar los planes de mejora y desarrollar las evidencias de cumplimiento.

III-D. Bloque 4 — Soporte Documental

Es fundamental para la trazabilidad, reproducibilidad y transferencia del conocimiento dentro de POSEIDON. Este bloque establece los artefactos formales (guías, catálogos, y planes) que documentan cada procedimiento y decisión ejecutada a lo largo de todo el marco de trabajo. Su propósito es asegurar la integridad del sistema, proveer la evidencia necesaria para futuras auditorías y, en última instancia, construir una base de conocimiento reutilizable que protege los activos y reduce el riesgo operativo. En la tabla IV se presentan los elementos, artefactos y resultados de este bloque.

Tabla IV
COMPONENTES DEL BLOQUE 4 – SOPORTE DOCUMENTAL

Elemento	**Descripción**
Artefactos Incluidos	Incluye el seguimiento actualizado de taxonomías marítimas, un diccionario operativo, registro de los procedimientos operativos adecuados y decisiones ejecutadas, matrices RACI, protocolos de prueba y diferentes planes necesarios (respuesta, continuidad, crisis).
Objetivo de Seguridad	Todo con un objetivo claro de proteger los activos y reducir el riesgo.
Resultados Esperados	Contar con un repositorio versionado y la evidencia necesaria para desarrollar una auditoría y una base de conocimiento reutilizable.

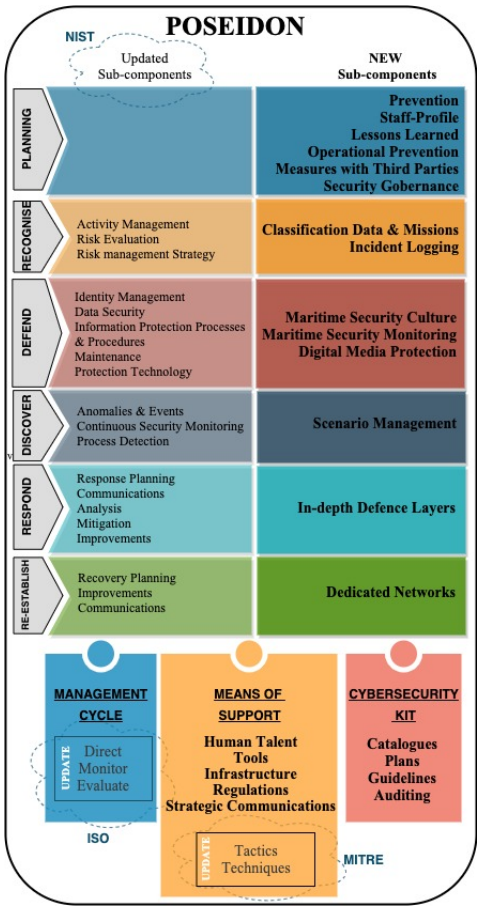


Figura 2. Componentes y adaptaciones del marco POSEIDON (de [1])

IV. DISCUSIÓN Y CONTRIBUCIONES

La matriz de cobertura demuestra contribuciones en Gestión de Riesgos Marítimos (MRM), en la alineación por misión, controles y continuidad. También se observan refuerzos en Gobernanza de Seguridad Marítima (MSG) y Cultura de Seguridad Marítima (MSC), que abordan políticas, responsabilidades y cultura organizacional.

Asimismo, el Transporte Marítimo (MT) y la Infraestructura Marítima (MI) se operacionalizan mediante catálogos y prácticas propias. Finalmente, la Detección de Amenazas y Vulnerabilidades (DAV) y la Gestión de Incidentes de Seguridad (DIS) se integran como ejes transversales para la autonomía, la automatización y la protección de la cadena industrial.

Para los armadores, astilleros y Autoridades Competentes, el marco POSEIDON proporciona beneficios operativos clave, tales como: (i) Trazabilidad integral entre activos, amenazas, controles y decisiones. (ii) Priorización accionable mediante segmentación IT/OT. (iii) Cumplimiento eficiente con normativas internacionales (IMO, NIST, IEC 63154, ISO/IEC 27001/62443). (iv) Desarrollo de la cultura de la ciberseguridad marítima y el talento humano por medio de métricas de riesgo clave (KRI) comparables.

Trabajo futuro: Se propone generar procesos de automatización de KRI; integración de datos operacionales en tiempo real mediante canales seguros; validaciones en escenarios multi-buque/multi-escenario (civil–militar).

V. CONCLUSIONES

A partir de los resultados de la revisión sistemática con la identificación de tendencias y brechas en el ámbito de la ciberseguridad marítima, se propone POSEIDON. Este es un marco dirigido específicamente al sector marítimo que busca operacionalizar los hallazgos cerrando carencias persistentes en la gobernanza, cultura, taxonomía e idiosincrasia marítimas.

Como respuesta directa, el trabajo proporciona un mapa estructurado del estado del arte, se identifican debilidades persistentes —notablemente en gobernanza, cultura de la ciberseguridad marítima, taxonomía e idiosincrasia marítimas— y consolida los requisitos regulatorios y de buenas prácticas clave (IMO, ENISA, ISO, NIST, IEC). Sobre esta base, El marco de ciberseguridad marítimo POSEIDON se presenta como un modelo integral que operacionaliza dichas necesidades mediante cuatro bloques fundamentales (Etapas, Soporte Técnico, Ciclo de Gestión y Soporte Documental), facilitando la trazabilidad entre activos, amenazas y controles.

En términos prácticos, POSEIDON orienta las decisiones críticas de riesgo, inversión y cumplimiento en entornos de sistemas IT/OT, promueve la estrategia de defensa en profundidad y la gestión continua a través de Indicadores Clave de Riesgo e Indicadores Clave de Desempeño (KRI/KPI).

AGRADECIMIENTOS

Esta publicación no habría sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MICIU/AEI/10.13039/501100011033 y la Unión Europea (Chist-Era Program); KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MICIU/AEI/10.13039/501100011033 /FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la JCCM, y FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; y Grant RED2024-154240-T financiado por MICIU/AEI/10.13039/501100011033. También agradecemos el apoyo de la Corporación Científica y Tecnológica para el Desarrollo de la Industria Naval, Marítima y Fluvial, COTECMAR, por su participación, disposición y compromiso en este trabajo.

REFERENCIAS

[1] F. Martínez, L. E. Sánchez, A. Santos-Olmo, D. G. Rosado, y E. Fernández-Medina, “Maritime cybersecurity: protecting digital seas”, en *International Journal of Information Security*, vol. 23, n. 2, pp. 1429–1457, 2024.

When Do Membership Inference Attacks Succeed? An Empirical Study of Overfitting in Fine-Tuned Large Language Models (LLMs)

Jean Dié, Najeeb Jebreel, David Sánchez, and Josep Domingo-Ferrer
Dept. of Computer Engineering and Mathematics and CYBERCAT, Universitat Rovira i Virgili, Tarragona
jean.die@fundacio.urv.cat, {najeeb.jebreel, david.sanchez, josep.domingo}@urv.cat

***Abstract*—Membership inference attacks (MIAs) barely succeed on pre-trained language models but may become effective after fine-tuning. We investigate under what conditions these attacks succeed. We evaluate five fine-tuning methods (full fine-tuning plus four parameter-efficient approaches) against five attack types across training epochs, using Pythia models (1B-12B parameters) on ArXiv abstracts with validated clean data splits. We find strong positive correlations between attack effectiveness and the training-validation loss gap across all experimental conditions, with reference-based attacks showing amplified sensitivity to overfitting compared to reference-free methods. These findings suggest that MIA effectiveness may be associated with overfitting rather than fundamental architectural vulnerabilities, indicating that loss gap monitoring could provide a practical privacy risk indicator during fine-tuning.**

***Index Terms*—membership inference attacks, fine-tuning, large language models, privacy-preserving machine learning, differential privacy**

I. Introduction

Organizations increasingly fine-tune large language models (LLMs) on proprietary datasets for domain-specific applications. This raises privacy concerns when training data contain sensitive information or copyrighted material.

Membership inference attacks (MIAs) aim to determine whether a specific example was used in model training [1]. This has implications for privacy auditing, regulatory compliance, and investigation of unauthorized data use, in addition to their potential privacy threat. Although recent work shows that MIAs barely outperform random guessing on pre-trained LLMs [2], [3], they may become more effective against fine-tuned LLMs [4], [5].

This stark difference in attack effectiveness raises a fundamental question: *under what conditions do MIAs succeed?* If MIAs succeed even against well-regularized models with good generalization, this would indicate an inherent privacy problem requiring novel defenses. Conversely, if they only succeed when models are overfitted –that is, when they fit training data better than validation data– then privacy-preserving fine-tuning could reduce to standard machine learning best practices, such as early stopping, regularization, and generalization monitoring.

To answer this question, we examine the relationship between the training-validation loss gap –a standard metric that quantifies overfitting– and the effectiveness of MIAs. The loss gap can be monitored during fine-tuning to prevent overfitting. If MIA success consistently correlates with the loss gap across various experimental conditions, this would provide evidence that attacks primarily detect overfitting rather than fundamental privacy flaws.

We present exploratory results examining this hypothesis through artifact-controlled evaluation on ArXiv abstracts using Pythia models ranging from 1B to 12B parameters. We evaluate five fine-tuning methods (full fine-tuning plus four parameter-efficient approaches: LoRA, BitFit, Adapter, Prefix) against five state-of-the-art MIAs across training epochs.

This work makes two contributions:

- It provides empirical evidence that MIA effectiveness strongly correlates with the training-validation loss gap across diverse experimental conditions (five fine-tuning methods, five attack types, three model scales), suggesting that attacks primarily detect overfitting rather than fundamental privacy vulnerabilities.
- It reports a training dynamics analysis across epochs showing that this correlation emerges consistently despite methodological confounding. This provides evidence that standard generalization practices may offer privacy benefits alongside model quality improvements.

II. Related Work

To contextualize our investigation, we review prior work on pre-trained models, fine-tuning-specific attacks, and parameter-efficient methods whose privacy properties remain unexplored.

A. Membership Inference on Pre-trained Language Models

Membership inference attacks [1], which aim to determine whether specific data was used in training, have been extensively studied on pre-trained models. Multiple studies have found that MIAs barely outperform random guessing for such models. Duan et al. [2] evaluated five attack methods on Pythia models trained on The Pile dataset, achieving an AUC metric below 0.6 in most domains (close to the 0.5 level that signals random guessing). Hayes et al. [3] found non-monotonic vulnerability across scales using the Likelihood Ratio Attack with 128 reference models, with models of size 85M being the most vulnerable (AUC 0.699, far from random guessing) and models of size 489M being the most resistant (AUC 0.547, close to random guessing).

Critically, Meeus et al. [6] demonstrated that bag-of-words (BoW) classifiers achieved great effectiveness (AUC > 0.98) on benchmarks by exploiting temporal markers and topic distributions rather than detecting true memorization. They introduced BoW as a diagnostic tool to detect distribution

shifts and emphasized that members and non-members must differ only in training inclusion for valid evaluation.

B. Membership Inference on Fine-tuned Models

Fine-tuning creates conditions distinct from pre-training through substantially smaller datasets and repeated multi-epoch exposure. Membership inference attacks can be categorized as reference-free (using only the output of target models) or reference-based (calibrating scores by comparing against a reference model, such as the pre-trained base model) [4].

Mireshghallah et al. [4] compared full, head-only, and adapter fine-tuning for GPT-2, finding head fine-tuning most vulnerable despite yielding the fewest modified parameters. They identified a "memorization only" training phase where the attack recall increased without corresponding training loss changes.

Reference-based attacks require a reference dataset with similar distribution to the training data to calibrate attack scores and reduce false positives from inherently over-represented texts. However, obtaining such datasets is often impractical. Fu et al. [5] introduced SPV-MIA to address this limitation by self-prompting: the target LLM generates its own reference dataset by completing short prompt texts. This shows that reference-based calibration can be achieved without requiring access to data similar to those used for training.

C. Parameter-Efficient Fine-tuning

Given the evidence that fine-tuning increases MIA vulnerability, understanding whether different fine-tuning strategies exhibit distinct vulnerability patterns becomes important.

Parameter-efficient fine-tuning (PEFT) methods modify small parameter subsets: LoRA [7] adds low-rank matrices to frozen weights; BitFit [8] updates only biases; adapters [9] insert bottleneck layers; prefix tuning [10] prepends learned prompts. Although widely adopted for computational efficiency, whether different parameter modification strategies create different MIA vulnerability patterns remains an open question.

III. Methodology

A. Threat Model

We evaluate membership inference under the standard threat model from MIA literature where the adversary seeks to determine whether specific examples were in the training dataset.

Following standard threat models in membership inference literature [1], [11], the adversary is assumed to possess:

- Candidate examples to test for membership
- Knowledge of the training data domain (e.g., ability to sample from similar data distributions)
- Query access to model outputs (logits and losses)
- Access to the publicly available pre-trained base model

This represents a conservative assessment that provides an upper bound on privacy risk.

For membership verification, the adversary collects non-member examples from the same domain to serve as calibration data. Attack scores computed on candidate examples are compared against the distribution of scores on non-members to determine membership likelihood.

B. Evaluation Dataset Construction

Following recommendations for randomized MIA evaluation on finetuned models [6], we construct evaluation datasets from the ArXiv dataset [12] using abstracts. ArXiv abstracts enable temporal control through publication dates, providing cleaner member/non-member splits that avoid distribution artifacts. Papers are randomly partitioned into disjoint member and non-member sets using an interleaved assignment strategy. Following Meeus et al. [6], we validate the splits through a bag-of-words (BoW) evaluation, achieving AUC 0.507 (near-random). Each split contains 250 member examples used for fine-tuning and 250 non-member examples (500 total) reserved for evaluation, providing balanced classes for ROC analysis.

C. Fine-tuning Methods

We evaluate five fine-tuning approaches:

- **Full Fine-tuning**: Updates all parameters (baseline)
- **LoRA** [7]: Adds low-rank matrices ($r = 4$, $\alpha = 16$)
- **BitFit** [8]: Updates only bias terms (<0.1% parameters)
- **Adapter** [9]: Inserts bottleneck layers ($d_{\text{bottleneck}}$=64)
- **Prefix** [10]: Prepends learned prompts (10 tokens/layer)

We use Pythia models [13] due to their open-source availability, well-documented training data (The Pile), publicly available base models enabling reference attacks, and manageable computational requirements for controlled experiments. All methods train for 5 epochs on 250 member ArXiv abstracts at three scales (1B, 6.9B, 12B parameters). We use method-specific learning rates empirically tuned to enable effective learning for each method (Full FT 10^{-5}, LoRA 5×10^{-4}, BitFit 2×10^{-3}, Adapter 5×10^{-5}, Prefix 5×10^{-3}). This introduces a confounding between fine-tuning method and learning rate, as using a uniform learning rate would prevent some methods from learning effectively. Our goal is to compare MIA effectiveness across methods achieving comparable overfitting levels, rather than isolating architectural effects.

D. Membership Inference Attacks

We evaluate five MIA methods, categorized by whether they require access to the pre-trained base model.

Reference-Free Attacks. These attacks operate solely on the fine-tuned model.

Loss-based attack [14] computes the average log-probability, with higher scores indicating membership. For a sequence of tokens $x = (x_1, \ldots, x_n)$:

$$\text{Score}_{\text{loss}}(x) = \frac{1}{n} \sum_{i=1}^{n} \log P_\theta(x_i \mid x_{<i}) \tag{1}$$

Min-K% [15] averages the $k = 20\%$ lowest token log-probabilities to detect outliers:

$$\text{Score}_{\text{Min-K\%}}(x) = \frac{1}{|S_k|} \sum_{i \in S_k} \log P_\theta(x_i \mid x_{<i}) \tag{2}$$

where S_k contains the 20% lowest-probability tokens.

Min-K%++ [16] enhances Min-K% through z-normalization against corpus-level token frequencies, reducing false positives from inherently rare tokens. That is,

the log-probabilities within the summation of Expression (2) must be replaced by

$$\frac{\log P(x_t \mid x_{<t}) - \mu_{\cdot|x_{<t}}}{\sigma_{\cdot|x_{<t}}},$$

where μ and σ are mean and standard deviation of log-probabilities over the vocabulary, averaging the lowest 20% of normalized values.

ZLIB [17] uses compression ratio as a memorization proxy.

Reference-Based Attack. This attack compares fine-tuned model behavior against the base model.

Reference attack [4] measures the behavioral change from fine-tuning by comparing the loss in the fine-tuned model versus the base model.

$$\text{Score}_{\text{ref}}(x) = \mathcal{L}_{\text{base}}(x) - \mathcal{L}_{\text{fine-tuned}}(x) \tag{3}$$

Training examples show larger loss reductions than non-members. This approach requires access to the pre-trained base model (before fine-tuning), which is publicly available for Pythia models but avoids the computational cost of training multiple reference models on auxiliary datasets.

For all attacks, we compute scores on both member and non-member sets, using ROC curves and AUC to measure discriminative power.

E. Evaluation Protocol

Following standard practice in membership inference evaluation [1], [11], we evaluate the effectiveness of attacks at each epoch checkpoint using ROC analysis, with AUC as the primary metric. AUC measures the ability to distinguish members from non-members across all classification thresholds, providing a threshold-independent measure of attack effectiveness.

To examine whether standard overfitting metrics correlate with membership inference vulnerability, we compute the training-validation loss gap at each epoch checkpoint. Specifically, we evaluate average token-level negative log-likelihood on the member set (250 training examples) and non-member set (250 held-out validation examples):

$$\text{gap} = \mathcal{L}_{\text{nonmember}} - \mathcal{L}_{\text{member}} \tag{4}$$

where $\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N} \log P(x_i)$ is the average negative log-likelihood per token. Positive gap values indicate overfitting, as the model assigns a lower loss to training data than to validation data.

This metric can be monitored during standard fine-tuning, making it a practical privacy indicator that does not require additional computation.

IV. Experimental Results

A. Loss Gap Tracks Attack Effectiveness

Figure 1 reports results for each combination of model scale, fine-tuning method, and epoch, spanning models with 1B to 12B parameters, five fine-tuning approaches, and progressive training stages.

The results show a *very strong positive correlation* between the training-validation loss gap and the effectiveness of attacks for all experimental configurations, as reported in Table I.

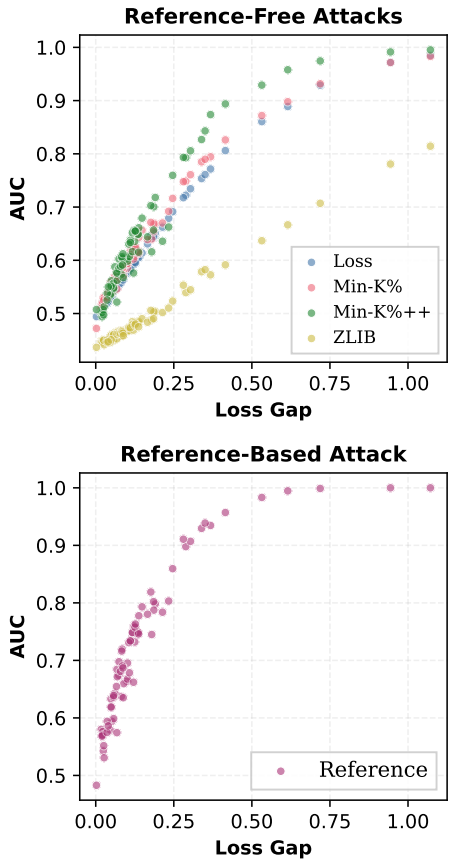


Figure 1. MIA effectiveness versus training-validation loss gap. **Top**: Reference-free attacks (Loss, Min-K%, Min-K%++, ZLIB). **Bottom**: Reference-based attack. Each point represents a combination of method, scale, and epoch. Min-K%++ shows the strongest performance among reference-free methods.

Reference-free attacks (top diagram of Figure 1) exhibit a striking pattern: probability-based methods (Loss, Min-K%, Min-K%++) demonstrate highly similar trajectories. All three attacks follow nearly parallel curves, scaling from near-random performance (AUC $\approx$ 0.5) at minimal loss gap to substantially elevated effectiveness (AUC $\approx$ 0.8-1.0) as overfitting increases. This similarity in scaling behavior suggests that these attacks exploit comparable signals from the model's probability distribution. Within this consistent pattern, Min-K%++ achieves marginally superior performance, followed closely by Loss and Min-K%. This relative ordering mirrors the effectiveness hierarchy observed on pre-trained models [15], [16], indicating that these probability-based attacks maintain their comparative advantages across both pre-training and fine-tuning contexts.

In contrast, ZLIB compression-based scoring substantially underperforms, remaining close to the random baseline (AUC $\approx$ 0.45-0.50) across all loss gap values. Although ZLIB has shown effectiveness for detecting pre-training data in previous work [17], it proves to be markedly less effective than probability-based methods in our fine-tuning evaluation setting. This suggests that the compression ratio captures weaker membership signals than model probability distributions.

Reference-based attacks (bottom diagram of Figure 1) show amplified sensitivity to overfitting. The Reference attack

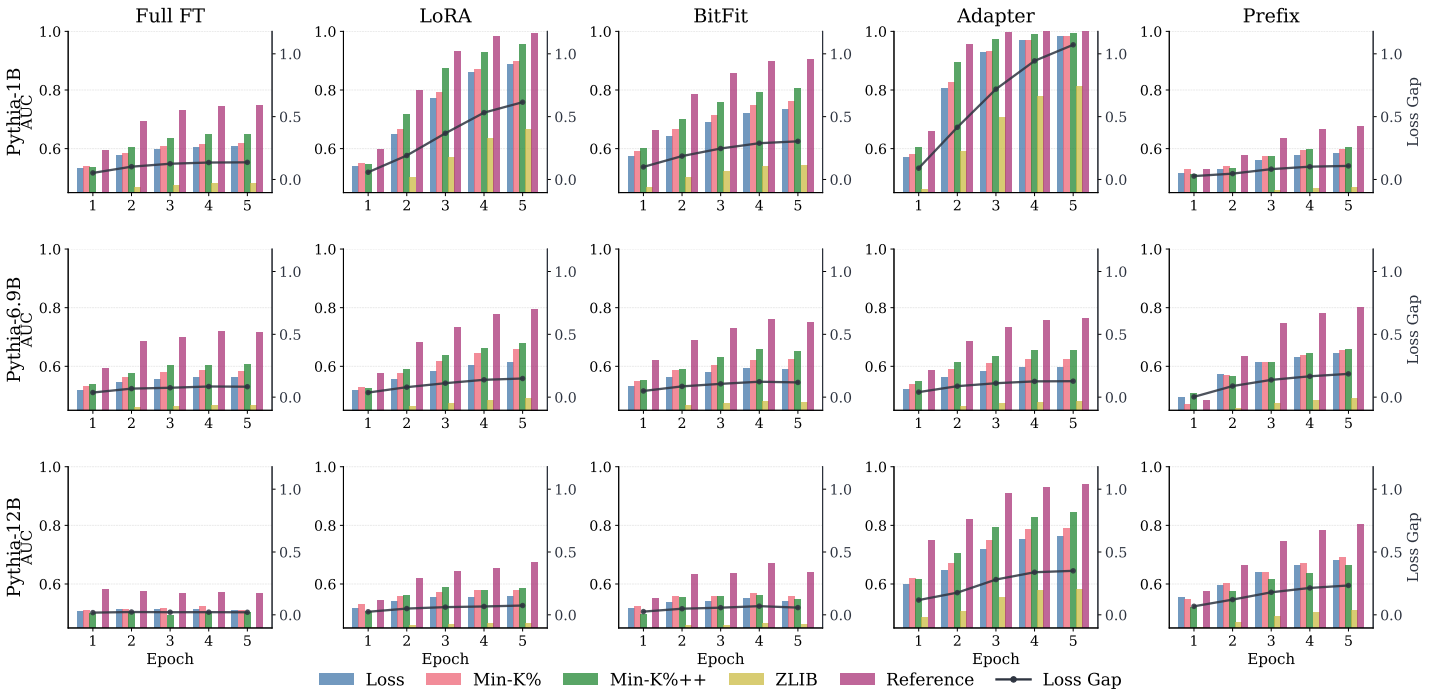


Figure 2. Loss gap-MIA effectiveness relationship across diverse experimental conditions. Each diagram shows one of 15 configurations (3 model scales × 5 fine-tuning methods) with temporal evolution across 5 epochs. Colored bars represent attack AUC; black line shows loss gap (right y-axis).

climbs from AUC ≈ 0.55-0.60 at minimal loss gap (≈ 0.02) to substantially higher values as overfitting increases. This increased effectiveness compared to reference-free methods suggests that explicitly measuring the behavioral change between base and fine-tuned models provides stronger membership signals than analyzing fine-tuned model outputs alone. The Reference attack's amplified response to loss gap indicates that it directly exploits the differential behavior created by overfitting: training examples receive disproportionately lower loss on the fine-tuned model compared to the base model when generalization degrades.

Quantitative correlation. These visual patterns are confirmed by *very strong* quantitative correlations between loss gap and attack AUC across 75 fine-tuning configurations (5 methods × 3 scales × 5 epochs). All correlations are statistically significant ($p < 0.001$). Table I presents correlation statistics for each attack method. Probability-based attacks exhibit the strongest linear correlations (Pearson r=0.919-0.970), while the Reference attack shows somewhat lower Pearson correlation (r=0.838) but high rank correlation (Spearman ρ=0.967), suggesting nonlinear scaling. This may arise because reference attacks measure relative behavioral change (Δloss), which compounds with overfitting severity. These correlation values confirm that the loss gap serves as a reliable quantitative predictor of MIA vulnerability across diverse experimental conditions.

B. Generalization Across Experimental Conditions

Figure 2 demonstrates that the loss gap-MIA effectiveness relationship observed in Figure 1 is maintained under various experimental conditions. Figure 2 presents 15 configurations that span three model scales (1B, 6.9B, 12B parameters) and five fine-tuning methods, with each subplot showing the temporal evolution over 5 epochs.

Table I
CORRELATION BETWEEN LOSS GAP AND ATTACK AUC.

Attack Method	Pearson r	Spearman ρ
Loss	0.970	0.998
Min-K%	0.954	0.990
Min-K%++	0.919	0.969
ZLIB	0.996	0.977
Reference	0.838	0.967

Consistent pattern across conditions. The loss gap-MIA effectiveness relationship emerges across all 15 configurations. As training progresses and models overfit (increasing loss gap), membership inference vulnerability grows correspondingly. The Reference attack consistently achieves the highest effectiveness when overfitting occurs, followed by Min-K%++, Loss, and Min-K%. ZLIB remains near-random across all conditions.

The consistency of this pattern across model scales (1B-12B), diverse parameter modification strategies (full fine-tuning, LoRA, BitFit, adapters, prefix tuning), and training epochs suggests the loss gap-MIA relationship is likely to be a general phenomenon in fine-tuned language models.

V. DISCUSSION

Our results raise the question: do these attacks detect fundamental privacy vulnerabilities or simply overfitting?

If MIAs succeeded even on well-regularized models with a minimal training-validation gap, this would indicate a fundamental privacy problem requiring novel defenses beyond standard machine learning practices. However, our results

show that the effectiveness of MIAs seems to be clearly connected with model overfitting and not so much with the nature of language models *per se*. If this interpretation is confirmed through further studies, it would suggest that standard regularization practices promoting model generalization (*i.e.*, early stopping, weight decay, dropout) may help reduce MIA vulnerability alongside their benefits for model quality.

Reference attacks show detectable effectiveness even at minimal loss gaps (AUC $\approx$ 0.55-0.60 at gap $\approx$ 0.02), indicating sensitivity to behavioral changes from fine-tuning under regularized conditions. This observation aligns with recent work on self-calibrated reference attacks [5], which demonstrate that reference-based calibration can detect membership using self-prompt methods that avoid requiring auxiliary datasets.

Context-dependent privacy impact. Although membership detection for the public abstracts we employ in our evaluation poses minimal harm, the same effectiveness for medical records or personal communications would constitute a privacy breach. Our results characterize the capability for technical detection, not domain-specific privacy risks.

However, our results do not definitively establish whether MIAs *exclusively* detect overfitting. Yeom et al. [14] established theoretical connections between overfitting and MIA vulnerability in classification settings, which our results extend to generative language models. Yet, overfitting (generalization failure) differs conceptually from memorization (encoding specific examples in parameters).

Previous work on pre-trained models provides suggestive evidence for the overfitting hypothesis: Duan et al. [2] found that MIAs barely outperform random guessing (AUC $<$ 0.6) on Pythia models trained on The Pile, where models see massive diverse data with minimal per-example overfitting. Our finding that MIAs become effective when fine-tuning introduces overfitting aligns with this pattern. However, pre-trained models still exhibit some vulnerability, and it remains unclear whether this residual effectiveness stems from subtle overfitting on repeated sequences in large corpora or from memorization phenomena orthogonal to overfitting metrics. Answering this would clarify to what extent MIAs are effective in practice on well-engineered models.

A. Reference-Based Attacks Amplify Weak Signals

Our results reveal a critical distinction: while probability-based methods (Loss, Min-K%, Min-K%++) require moderate overfitting (loss gaps $>$ 0.3-0.4) for effective detection (AUC $>$ 0.8), reference attacks achieve this same effectiveness level (AUC $>$ 0.8) at substantially lower loss gaps ($>$ 0.18). This amplified sensitivity stems from directly measuring behavioral change by comparing fine-tuned against base model predictions. Even well-regularized models maintaining small loss gaps remain vulnerable to reference-based membership inference, complicating the use of loss gap as a privacy indicator, and motivating formal defenses like differentially private-stochastic gradient descent (DP-SGD).

B. Differential Privacy as a Formal Defense

Our finding that MIA effectiveness correlates with the training-validation loss gap raises a question about the relationship between standard regularization and formal privacy defenses for language models. Whereas techniques such as weight decay and dropout are insufficient to prevent memorization during fine-tuning [4], ϵ-differential privacy with DP-SGD provides formal worst-case guarantees independent of attack strategy [18]. However, overfitting is sufficient but not necessary for privacy leakage [14] —well-generalized models can leak information about outliers. The critical unresolved question concerns whether DP-SGD's privacy benefits arise from explicit noise addition or from implicit overfitting reduction. Our results demonstrating a strong correlation between the loss gap and the effectiveness of MIAs suggest that mechanisms that reduce overfitting may provide informal privacy benefits [19], although this correlation cannot replace formal privacy accounting through ϵ bounds.

C. Architectural Considerations

Our experiments reveal patterns of vulnerability across fine-tuning methods, but an important confounding factor must be acknowledged: learning rates differ across methods following established practices from the literature. This confounding limits our ability to attribute the observed vulnerability differences solely to architectural properties.

Nevertheless, the architectural distinctions between PEFT methods suggest a fruitful direction for future controlled experiments. Different methods modify fundamentally different parameter types: LoRA introduces low-rank updates to weight matrices, BitFit modifies only bias terms, Adapter tuning adds bottleneck layers, and Prefix tuning learns continuous prompt representations. Previous work provides preliminary evidence that modification location may influence privacy properties. Mireshghallah et al. [4] found unexpected vulnerability patterns where head-only fine-tuning was most vulnerable despite modifying the fewest parameters, suggesting that parameter quantity does not directly predict privacy risk.

Further experiments with controlled learning rates would enable investigation of whether these architectural differences manifest distinct privacy characteristics. For example, LoRA's restriction to low-rank subspaces may constrain memorization differently than BitFit's sparse bias modifications. Understanding these relationships would require systematic ablation studies that vary the learning rate, parameter budget, and architectural choice independently. Such controlled investigations could inform privacy-preserving fine-tuning recommendations based on architectural properties rather than method-specific empirical findings.

D. Limitations and Future Work

Experimental scope. We evaluated models with 1B-12B parameters, while modern deployments use 70B+ models. Although we used a single split of ArXiv abstracts, evaluating across multiple independent splits would strengthen generalizability. It would also be interesting to isolate the parameter modification strategy from the optimization dynamics. Future work should systematically vary learning rates within methods, validate at industry scales (70B+ parameters) with contemporary architectures (Llama, Mistral), and test across diverse domains.

Understanding the dynamics of the loss gap. Our focus on the loss gap as a monitoring signal must account for

findings by Mireshghallah et al. [4], who identified a "memorization only" training phase where attack recall increased without corresponding training loss changes. This suggests that the loss gap may not capture all memorization dynamics during fine-tuning.

Memorization-focused analysis. While our study focuses on overfitting (generalization failure), memorization can occur even in models that generalize well [20]. The distinction between overfitting and memorization phases observed by Mireshghallah et al. [4] suggests that future work should complement our overfitting-based analysis with direct memorization metrics to investigate whether MIA effectiveness correlates more strongly with one or the other.

Differential privacy validation. Future work should validate our overfitting hypothesis through controlled DP-SGD experiments. Recent analysis [19] suggests that standard anti-overfitting techniques achieve comparable privacy protection to DP-SGD at commonly used privacy budgets (ϵ=2-8). Experiments with strong privacy ($\epsilon \leq 1$) would clarify whether loss gap monitoring can serve as a practical privacy indicator or whether formal DP guarantees remain necessary.

VI. Conclusion

We investigated under what conditions membership inference attacks succeed on fine-tuned language models. Across five fine-tuning methods, five attack types, and three model scales (1B-12B parameters), we found that attack effectiveness consistently correlates with the training-validation loss gap. This pattern emerges for 75 experimental conditions (15 configurations × 5 epochs).

These findings suggest that MIAs primarily succeed when models are overfitted, rather than exploiting fundamental architectural privacy vulnerabilities. This interpretation aligns with prior work showing that MIAs barely succeed on pre-trained models (where overfitting is minimal) but become effective during fine-tuning (where overfitting emerges). If confirmed through systematic validation, this would suggest that standard generalization practices may reduce MIA vulnerability alongside improving model quality.

Acknowledgments

Partial support to this work has been received from the Government of Catalonia (ICREA Acadèmia Prizes to J. Domingo-Ferrer and to D. Sánchez, and grant 2021SGR-00115), MCIN/AEI under grant PID2024-157271NB-I00 "CLEARING-IT", and the EU's NextGenerationEU/PRTR via INCIBE (project "HERMES" and INCIBE-URV cybersecurity chair).

References

[1] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 3–18, arXiv:1610.05820.

[2] J. Duan, F. Kong, S. Wang, X. Shi, and K. Xu, "Do membership inference attacks work on large language models?" in *Conference on Language Modeling (COLM)*, 2024, arXiv:2402.07841.

[3] J. Hayes, M. Meeus, I. Jiang, J. Duan, and A. Weller, "Strong membership inference attacks on massive datasets and llms," in *arXiv preprint*, 2024, arXiv:2501.01869.

[4] F. Mireshghallah, A. Uniyal, T. Wang, D. Evans, and T. Berg-Kirkpatrick, "An empirical analysis of memorization in fine-tuned autoregressive language models," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, December 2022, pp. 1816–1826. [Online]. Available: https://aclanthology.org/2022.emnlp-main.119/

[5] W. Fu, H. Gao, G. Jia, Y. Lou, T. Zhang, Y. Li, D. Jin, and Y. Li, "Practical membership inference attacks against fine-tuned large language models via self-prompt calibration," in *AAAI Conference on Artificial Intelligence*, 2024, arXiv:2311.06062.

[6] M. Meeus, I. Shilov, S. Jain, M. Faysse, M. Rei, and Y.-A. de Montjoye, "Sok: Membership inference attacks on llms are rushing nowhere (and how to fix it)," in *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 2025, arXiv:2406.17975.

[7] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *International Conference on Learning Representations (ICLR)*, 2022, arXiv:2106.09685.

[8] E. B. Zaken, S. Ravfogel, and Y. Goldberg, "Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022, arXiv:2106.10199.

[9] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for nlp," in *International Conference on Machine Learning (ICML)*, 2019, arXiv:1902.00751.

[10] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," in *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021, arXiv:2101.00190.

[11] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, "Membership inference attacks from first principles," in *2022 IEEE Symposium on Security and Privacy (SP)*, 2022, pp. 864–881.

[12] arXiv.org submitters, "arxiv dataset," https://www.kaggle.com/dsv/7548853, 2024.

[13] S. Biderman, H. Schoelkopf, Q. Anthony, H. Bradley, K. O'Brien, E. Hallahan, M. A. Khan, S. Purohit, U. S. Prashanth, E. Raff, A. Skowron, L. Sutawika, and O. van der Wal, "Pythia: A suite for analyzing large language models across training and scaling," in *International Conference on Machine Learning (ICML)*, 2023, arXiv:2304.01373.

[14] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *31st IEEE Computer Security Foundations Symposium (CSF)*, 2018, pp. 268–282.

[15] W. Shi, A. Ajith, M. Xia, Y. Huang, D. Liu, T. Blevins, D. Chen, and L. Zettlemoyer, "Detecting pretraining data from large language models," in *International Conference on Learning Representations (ICLR)*, 2024, arXiv:2310.16789.

[16] Z. Zhang, J. Zhou, J. Wang, J. Lai, Z. Liu, Y. Zhang, E. Zhang, Q. Huang, N. Feng, T. Liu, D. Ong, J. Ott, and G. Dyer, "Min-k%++: Improved baseline for detecting pre-training and detecting pre-training and copyright data in large language models," in *arXiv preprint*, 2025, arXiv:2404.02936.

[17] N. Carlini, F. Tramèr, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, Ú. Erlingsson, A. Oprea, and C. Raffel, "Extracting training data from large language models," in *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, Aug. 2021, pp. 2633–2650, arXiv:2012.07805. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting

[18] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2016, pp. 308–318.

[19] A. Blanco-Justicia, D. Sanchez, J. Domingo-Ferrer, and K. Muralidhar, "A critical review on the use (and misuse) of differential privacy in machine learning," *ACM Computing Surveys*, vol. 55, no. 8, pp. 1–26, 2023, arXiv:2206.04621.

[20] G. J. J. van den Burg and C. K. I. Williams, "On memorization in probabilistic deep generative models," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, 2021, pp. 27 916–27 928, arXiv:2106.03216.

Mejora de la anonimización de texto mediante explicabilidad orientada al riesgo de re-identificación

Benet Manzanares-Salor
Universitat Rovira i Virgili, CYBERCAT
Tarragona, España
benet.manzanares@urv.cat

David Sánchez
Universitat Rovira i Virgili, CYBERCAT
Tarragona, España
david.sanchez@urv.cat

***Resumen*—Este es un resumen extendido del artículo publicado en [1]. La anonimización de texto es una tarea compleja realizada habitualmente por expertos humanos de forma manual, cosa que conlleva costes significativos. Los mecanismos automáticos propuestos en la literatura se basan principalmente en *reconocimiento de entidades nombradas* (NER), una técnica reconocida como insuficiente en términos de protección de la privacidad. Para abordar esta limitación se propone PETRE, un método diseñado para mejorar la protección de la privacidad que proporciona garantías empíricas contra la re-identificación. Este emplea técnicas de explicabilidad para detectar y enmascarar iterativamente los términos que causan el mayor riesgo de re-identificación hasta que se alcanza el nivel de k-anonimato deseado. Los experimentos demuestran que PETRE es capaz de mejorar de manera significativa y consistente la protección de la privacidad de las anonimizaciones basadas en NER, a la vez que muestran las limitaciones de la anonimización manual.**

***Index Terms*—Protección de la privacidad, Modelos de lenguaje, Explicabilidad, Anonimización de texto, Re-identificación**

I. Introducción

La sociedad digital actual requiere grandes volúmenes de información, siendo el texto una de las fuentes más comunes y ricas en información. No obstante, los datos textuales a menudo contienen información sensible sobre individuos (como historiales médicos o casos judiciales), creando un conflicto directo con regulaciones de privacidad como el *reglamento general de protección de datos (GDPR)* [2] en Europa. Para mitigar estos riesgos, los gestores de datos recurren a la anonimización, consistente en el enmascaramiento irreversible de los datos para que no puedan atribuirse a los sujetos a los que se refieren.

Mientras que la anonimización de datos estructurados (tabulares) es un campo maduro con modelos formales como k-anonimato [3] y la privacidad diferencial [4], la anonimización de datos no estructurados como el texto es una tarea mucho más compleja. La ambigüedad inherente y la dependencia del contexto del lenguaje exigen un amplio conocimiento del dominio para identificar la información que facilita la re-identificación. Debido a estas dificultades, la anonimización de texto se realiza a la práctica principalmente de forma manual [5], [6].

Para automatizar o asistir este proceso, se han propuesto mecanismos basados en el *reconocimiento de entidades nombradas (NER)* [7], [8], [9]. Estos métodos utilizan modelos de aprendizaje automático para detectar tipos predefinidos de entidades (por ejemplo, nombres o lugares), que se asumen como facilitadores de la re-identificación. Sin embargo, este enfoque presenta serias deficiencias, ya que proporciona una protección insuficiente, carece de una garantía de privacidad y no incorpora un mecanismo para configurar el grado de protección deseado. Por estas razones, los métodos basados en NER pueden servir como un punto de partida para la anonimización manual, no son una solución completa.

I-A. Contribuciones

Considerando las limitaciones del NER, se propone *PETRE (privacy enhancement for text via risk-oriented explainability)*, un método fundamentado en la privacidad que busca mejorar las anonimizaciones basadas en NER imponiendo una reducción del riesgo de re-identificación definida por el usuario. Sus principales contribuciones incluyen:

- **Generalidad:** El método es aplicable a cualquier método de anonimización de partida, independientemente de su principio de funcionamiento.
- **Privacidad configurable:** Permite configurar de forma intuitiva el nivel de privacidad deseado, mediante la especificación de los sujetos a proteger y del riesgo de re-identificación máximo permitido, expresado en términos de *k-anonimato probabilístico*.
- **Personalización por sujeto:** A diferencia del k-anonimato estándar, el conocimiento de fondo para evaluar los riesgos puede ser específico para cada sujeto, reduciendo el enmascaramiento innecesario.
- **Anonimización iterativa:** El riesgo de re-identificación se revalúa después de cada enmascaramiento hasta alcanzar el umbral de privacidad, minimizando el sobre-enmascaramiento.
- **Garantía empírica:** Al ser guiado por el riesgo observado, el resultado anonimizado presenta un riesgo empírico de re-identificación $\leq 1/k$, eliminando la necesidad de evaluaciones manuales *a posteriori*.
- **Automatización:** El flujo de trabajo es totalmente automático, sin requerir intervención humana.

Los experimentos demuestran que PETRE reduce el riesgo de re-identificación de cualquier anonimización de partida y compite con métodos de vanguardia sin sus costes asociados.

II. Fundamentos

II-A. Anonimización

La anonimización consiste en transformar los datos de forma irreversible para impedir la identificación de los sujetos. Una vez adecuadamente anonimizados, los datos dejan de ser personales y quedan fuera del alcance de marcos legales como la GDPR [2].

Este proceso introduce distorsión y reduce la utilidad de los datos, implicando un balanceo entre privacidad y usabilidad [10], [9].

En este contexto, los atributos de los datos se clasifican en: *identificadores* (p. ej., nombre o DNI), que permiten la identificación directa; *cuasi-identificadores* (p. ej., fecha + lugar de nacimiento + empleo), que posibilitan la re-identificación cuando se combinan con conocimiento de fondo, es decir, información disponible públicamente sobre los individuos; *atributos confidenciales* (p. ej., diagnósticos o salarios), que contienen información sensible; y *atributos no confidenciales*, que no implican riesgo ni revelan datos sensibles.

La anonimización elimina los identificadores y transforma solo los cuasi-identificadores necesarios para garantizar el nivel de privacidad requerido, a la vez que preserva la máxima utilidad que la protección permita.

II-B. k-Anonimato

El k-anonimato [3] es un modelo de privacidad ampliamente utilizado en datos estructurados, que exige que cada individuo sea indistinguible de al menos otros $k-1$ respecto a sus cuasi-identificadores. Esto impone un límite superior en la probabilidad de re-identificación:

$$\text{Probabilidad de re-identificación} \leq \frac{1}{k} \tag{1}$$

El parámetro k permite ajustar el equilibrio entre la protección de la privacidad y la utilidad de los datos. El k-anonimato normalmente se implementa mediante supresión y generalización de atributos hasta formar registros idénticos en conjuntos de tamaño mínimo k.

El k-anonimato probabilístico [11] relaja este requisito manteniendo el mismo límite de riesgo, pero sin necesidad de generar registros indistinguibles. Esta propiedad lo hace más adecuado para texto, donde imponer el requisito indistinguibilidad puede mermar significativamente la utilidad.

II-C. Anonimización de texto basada en NER

Los enfoques más comunes de anonimización automática se basan en NER, y asumen que los (cuasi-)identificadores corresponden a tipos específicos de entidades nombradas (personas, lugares, etc.). Estos métodos detectan un conjunto predefinido de NE y las eliminan o substituyen por su tipo (PERSONA, LUGAR).

Sin embargo, presentan limitaciones importantes:

- **Premisa cuestionable, protección insuficiente:** Limitar los (cuasi-)identificadores a entidades nombradas no ofrece una protección robusta, ya que cualquier término relacionado con un individuo y disponible públicamente puede permitir re-identificación. Como consecuencia, muchos términos quedan sin enmascarar y persiste un riesgo elevado.

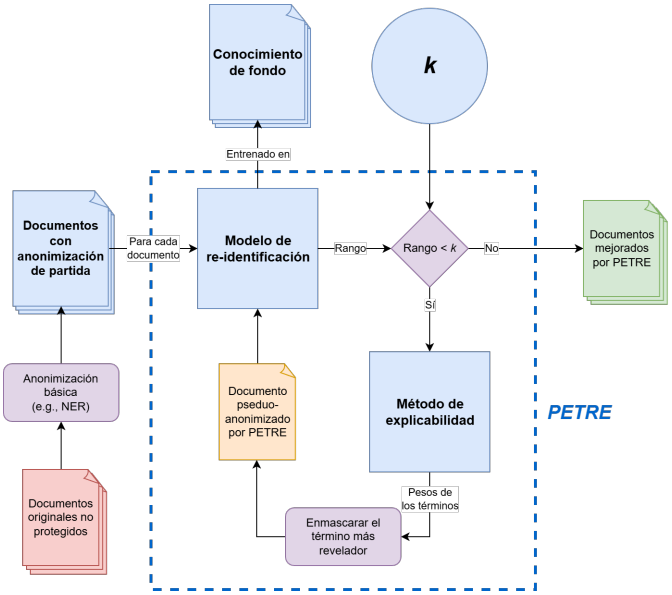


Figura 1. Arquitectura de PETRE

- **Falta de garantía:** No proporcionan una garantía de privacidad, por lo que el nivel de protección solo puede evaluarse *ex post*.
- **Incapacidad de configuración:** No incluyen parámetros que permitan controlar el nivel de privacidad y, por tanto, ajustar la relación entre protección y utilidad deseada.

III. Metodología

PETRE parte de texto anonimizado deficientemente (a partir de NER, por ejemplo) y lo protege de forma automática e iterativa hasta alcanzar un nivel de privacidad predefinido por el usuario en términos de k-anonimato probabilístico.

El usuario debe especificar los documentos a procesar, el valor deseado de k y el conocimiento de fondo. PETRE evalúa iterativamente el riesgo de re-identificación, y utiliza explicabilidad para detectar y enmascarar progresivamente los términos más reveladores hasta que la probabilidad de re-identificación sea $\leq 1/k$. Este diseño iterativo, detallado en la Figura 1, garantiza que solo se enmascaren los términos necesarios, evitando el sobre-enmascaramiento.

Los componentes centrales de PETRE son:

- **Documentos anonimizados de partida:** Conjunto de documentos de entrada, resultado de una anonimización básica (e.g., NER), etiquetados con la identidad del individuo a proteger. Alternativamente, la etiqueta puede referirse a un grupo de individuos u organización que solo existe en dicho documento.
- **Conocimiento de fondo:** Colección de documentos públicos identificados utilizados para entrenar el modelo de re-identificación. El solapamiento entre este conocimiento y los documentos de partida determina el riesgo; un conocimiento de fondo más escaso reduce el enmascaramiento necesario para un determinado k.
- **Modelo de re-identificación:** Clasificador entrenado con el conocimiento de fondo que predice la identidad. Evalúa el nivel de privacidad de un documento generando una distribución de probabilidad sobre las identidades

posibles. El riesgo se evalúa a partir del rango (posición) de la identidad correcta en esa distribución.

- k **(umbral de privacidad):** Un entero mayor que uno que define el nivel de riesgo máximo permitido ($1/k$) y establece la condición de parada del proceso iterativo. El proceso se detiene cuando el rango predicho es $Rango \geq k$. Un valor de $k \geq 2$ evita la re-identificación inequívoca, aunque se esperan valores superiores que impliquen menor riesgo (por ejemplo, $k > 10$).
- **Método de explicabilidad:** Módulo que estima la influencia (pesos) de los términos del documento en la probabilidad predicha para la identidad correcta. Pesos positivos indican que un término contribuye a la re-identificación, y serán éstos los que se prioricen para enmascaramiento en cada iteración.

III-A. Formalización

El conjunto de documentos de partida se define como $A = \{(D_1, I_1), \ldots, (D_n, I_n)\}$, donde cada documento D_i está asociado a una identidad I_i. El modelo de re-identificación $\Gamma(d)$ devuelve una distribución de probabilidad $P = (p_1, \ldots, p_n)$, y la re-identificación se mide mediante la posición de p_i en dicha distribución, denotada como $Rango(d, i, \Gamma)$. El límite del riesgo se deriva de la siguiente manera:

$$Rango(d, i, \Gamma) = k \rightarrow \text{Riesgo de re-identificación}(d) \leq 1/k \quad (2)$$

Esto corresponde al k-anonimato probabilístico, acotando el riesgo a $1/k$ sin requerir indistinguibilidad entre documentos.

Si el $Rango$ está por debajo de la k fijada, se requiere enmascaramiento. La función de explicabilidad $\xi(d, i, \Gamma)$ genera los pesos $(w_1, \ldots, w_m)$ asociados a los términos $(t_1, \ldots, t_m)$ de d, donde los pesos positivos señalan influencia directa a la re-identificación. Entonces, el término obteniendo el mayor peso se enmascara.

El documento resultante se revalúa, repitiendo iterativamente el proceso hasta alcanzar un $Rango \geq k$.

IV. Experimentos

A continuación se detallan los experimentos realizados para la validación y análisis de PETRE como método de mejora de anonimización de texto. El código y datos se encuentran disponibles en el siguiente repositorio de GitHub: https://github.com/BenetManzanaresSalor/PETRE.

IV-A. Conjunto de datos

Los experimentos se realizaron sobre un conjunto de 553 resúmenes de biografías de la Wikipedia seleccionadas aleatoriamente [12]. Estos resúmenes se anonimizaron manualmente, cosa que utilizaremos como referencia. Hasta dónde sabemos, este es el mayor corpus de propósito general anonimizado por humanos disponible públicamente.

IV-B. Implementación

El conocimiento de fondo y el valor de k de PETRE se ajustaron para igualar el nivel de protección de la anonimización humana. Los propios resúmenes de cada artículo actúan como conocimiento de fondo, replicando la información disponible para los expertos humanos y representando un *peor escenario* habitual en la evaluación de la privacidad. Para determinar k, se utilizó el modelo de re-identificación propuesto en [13] con dicho conocimiento de fondo. El 81 % de los documentos anonimizados manualmente obtuvieron un $rango = 1$, lo que sugiere que los expertos buscaban un nivel de anonimización no mayor a $k = 2$ (el mínimo que evita la re-identificación). Por ello, se adopta $k = 2$ en los experimentos posteriores. El artículo original incluye análisis adicionales sobre cómo varían los resultados con distintos valores de conocimiento de fondo y k. Finalmente, como método de explicabilidad se utilizó SHAP [14].

IV-C. Anonimización de partida

Evaluamos el efecto de PETRE sobre un conjunto de cuatro métodos basados en NER reportados en la literatura [15], [16], [13], [17], [18]. Tres de ellos se basan en el paquete de anotación Standford NER [19], diferenciándose en los tipos de entidades detectadas: `St.NER3`, `St.NER4` y `St.NER7`. El cuarto corresponde al proyecto de código abierto de Microsoft denominado `Presidio`[1].

Adicionalmente, evaluamos dos estrategias orientadas a la *publicación de datos con preservación de privacidad (PPDP)*: `Word2Vec` [15] y k`-anonymity greedy` [12]. Dichos enfoques no dependen de conjuntos fijos de entidades predefinidas, y representan el estado del arte en cuanto a protección de texto [10], [13], [17], [18]. No obstante, presentan limitaciones prácticas, como por ejemplo, depender de recursos externos complejos de obtener, como grafos de conocimiento.

IV-D. Resultados

Para evaluar el alineamiento entre las anonimizaciones automáticas y manuales, empleamos las métricas *recall* y *precisión*, habituales en la literatura [16], [15], [10].

Recall representa el porcentaje de términos identificadores marcados por los expertos humanos que también fueron detectados por el método automático, lo cual refleja el nivel de protección alcanzado. Como se aprecia en la figura 2, PETRE mejora el *recall* de forma significativa y consistente en todas las anonimizaciones de partida. Esto es particularmente evidente en aquellas basadas en NER, que parten de un valor más bajo. La mejora permite situarlas en niveles comparables a los métodos orientados a PPDP, siendo PETRE más sencillo de implementar gracias a su menor complejidad y la ausencia de dependencias externas. Asimismo, PETRE logra aumentar (aunque más levemente) el *recall* de estos últimos métodos.

Precisión corresponde al porcentaje de términos detectados automáticamente que coinciden con los identificadores establecidos por los expertos. La figura 3 muestra que PETRE afecta negativamente este valor, reduciéndolo aproximadamente al 50 %. Esto puede justificarse considerando que el 81 % de las anonimizaciones manuales obtuvieron un $rango = 1$, lo que evidencia una protección insuficiente. En consecuencia, PETRE se ha visto obligado a enmascarar numerosos (cuasi-)identificadores no contemplados en la anonimización manual para cumplir con el requisito $k = 2$.

V. Conclusiones y trabajo futuro

PETRE representa una solución práctica y configurable para mejorar la anonimización de texto, especialmente para los métodos basados en NER, que son los más utilizados en la

[1] https://github.com/microsoft/presidio/releases/tag/2.2.354

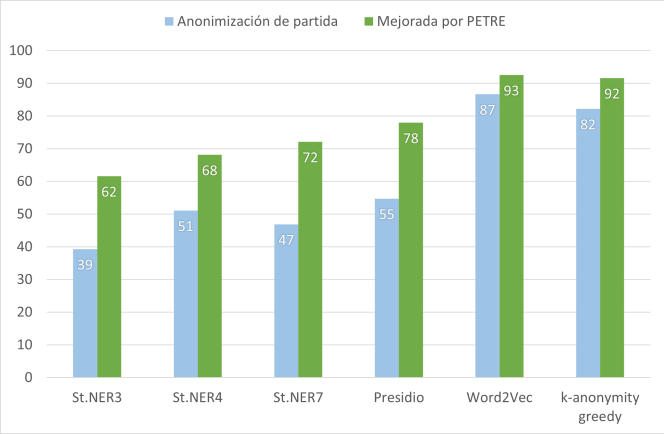


Figura 2. Recall antes y después de PETRE

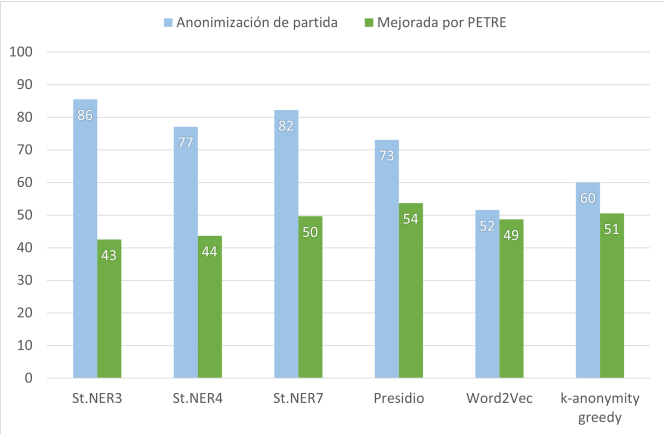


Figura 3. Precisión antes y después de PETRE

práctica pero que carecen de garantías. Los resultados experimentales confirman que PETRE mejora significativamente y consistentemente el nivel de privacidad de las anonimizaciones de partida.

La evaluación del riesgo empírico de PETRE también ha puesto de manifiesto deficiencias en la protección lograda por la anonimización manual.

El principal inconveniente de la propuesta es el tiempo de ejecución causado por la naturaleza iterativa del proceso y el elevado coste computacional del método de explicabilidad (SHAP[14]), siendo esta la operación más costosa. Como trabajo futuro, se pretende investigar métodos de explicabilidad alternativos con un menor coste computacional.

Agradecimientos

Este trabajo ha sido financiado por la Generalitat de Catalunya (Premio ICREA Acadèmia a D. Sánchez, y ayuda 2021SGR-00115), el MCIN/AEI bajo la ayuda PID2024-157271NB-I00 "CLEARING-IT", y los fondos NextGenerationEU/PRTR de la Unión Europea a través del INCIBE (proyecto "HERMES" y Cátedra de Ciberseguridad INCIBE-URV). B. Manzanares-Salor cuenta con el apoyo del Gobierno de España bajo la ayuda FPU23/01785.

Referencias

[1] B. Manzanares-Salor and D. Sánchez, "Enhancing text anonymization via re-identification risk-based explainability," *Knowledge-Based Systems*, vol. 310, p. 112945, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095070512401579X

[2] European Commission, "Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation)," 2016. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj

[3] P. Samarati, "Protecting respondents identities in microdata release," *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010–1027, 2001.

[4] C. Dwork, "Differential privacy," in *International Colloquium on Automata, Languages and Programming*, vol. 4052. Springer, 2006, Conference Proceedings, pp. 1–12.

[5] E. Bier, R. Chow, P. Golle, T. H. King, and J. Staddon, "The rules of redaction: Identify, protect, review (and repeat)," *IEEE Security & Privacy*, vol. 7, no. 6, pp. 46–53, 2009.

[6] S. Lima-López, E. Farré-Maduell, L. Gasco Sánchez, A. López Rueda, S. Frid, A. Conesa, X. Pastor, and M. Krallinger, "Carmen-i: Anonymization protocol for clinical reports in english," Zenodo, Tech. Rep., 2023.

[7] Y. Huang, Z. Song, D. Chen, K. Li, and S. Arora, "Texthide: Tackling data privacy in language understanding tasks," in *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, 2020, pp. 1368–1382.

[8] F. Hassan, J. Domingo-Ferrer, and J. Soria-Comas, "Anonymization of unstructured data via named-entity recognition," in *International Conference on Modeling Decisions for Artificial Intelligence*. Mallorca, Spain: Springer, 2018, Conference Proceedings, pp. 296–305.

[9] I. Pilán, B. Manzanares-Salor, D. Sánchez, and P. Lison, "Truthful text sanitization guided by inference attacks," *Applied Soft Computing*, vol. 185, p. 114013, 2025.

[10] P. Lison, I. Pilán, D. Sánchez, M. Batet, and L. Øvrelid, "Anonymisation models for text data: State of the art, challenges and future directions," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021, pp. 4188–4203.

[11] J. Soria-Comas and J. Domingo-Ferrer, "Probabilistic k-anonymity through microaggregation and data swapping," in *2012 IEEE International Conference on Fuzzy Systems*, 2012, pp. 1–8.

[12] A. Papadopoulou, P. Lison, L. Øvrelid, and I. Pilán, "Bootstrapping text anonymization models with distant supervision," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, Jun. 2022, pp. 4477–4487.

[13] B. Manzanares-Salor, D. Sánchez, and P. Lison, "Evaluating the disclosure risk of anonymized documents via a machine learning-based re-identification attack," *Data Min. Knowl. Discov.*, vol. 38, no. 6, pp. 4040–4075, 2024. [Online]. Available: https://doi.org/10.1007/s10618-024-01066-3

[14] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30. Association for Computing Machinery, 2017, Conference Proceedings, pp. 4768–4777.

[15] F. Hassan, D. Sánchez, and J. Domingo-Ferrer, "Utility-preserving privacy protection of textual documents via word embeddings," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, pp. 1058–1071, 2023.

[16] I. Pilán, P. Lison, L. Ørelid, A. Papadopoulou, D. Sánchez, and M. Batet, "The text anonymization benchmark (tab): A dedicated corpus and evaluation framework for text anonymization," *Computational Linguistics*, vol. 48, no. 8, pp. 1053–1101, 2022.

[17] B. Manzanares-Salor, D. Sánchez, and P. Lison, "Automatic evaluation of disclosure risks of text anonymization methods," in *Privacy in Statistical Databases - International Conference, PSD 2022, Paris, France, September 21-23, 2022, Proceedings*, ser. Lecture Notes in Computer Science, J. Domingo-Ferrer and M. Laurent, Eds., vol. 13463. Springer, 2022, pp. 157–171.

[18] B. Manzanares-Salor and D. Sánchez, "A comparative analysis, enhancement and evaluation of text anonymization with pre-trained large language models," *Expert Systems with Applications*, vol. 297, p. 129474, 2026. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417425030908

[19] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Baltimore, Maryland, USA, 2014, pp. 55–60.

Predicción de Carga Segura y Privada en Smart Grids mediante Incentivos Basados en Blockchain

Joan Ferré-Queralt *, Jordi Castellà-Roca*, Alexandre Viejo*

* Universitat Rovira i Virgili
Dept. d'Enginyeria Informàtica i Matemàtiques, CYBERCAT,
Av. Països Catalans 26, E-43007 Tarragona, Spain
{joan.ferreq, alexandre.viejo, jordi.castella}@urv.cat

***Resumen*—Éste es un resumen extendido de la publicación [1]. Las redes eléctricas inteligentes requieren predicciones de carga precisas, y aun utilizando *federated learning*, éste sigue presentando riesgos de privacidad y ataques de participantes maliciosos. Este trabajo presenta un marco seguro para *federated learning* en *smart grids*, basado en envíos anónimos, *zero-knowledge proofs* y firmas ciegas para proteger la identidad y los datos del usuario. Además, se introduce un mecanismo de reputación en *blockchain* que incentiva el comportamiento honesto y un *clustering* distribuido para manejar la heterogeneidad de los datos. La evaluación muestra que el sistema preserva la privacidad, mantiene la integridad del modelo y funciona eficientemente en dispositivos IoT.**

***Index Terms*—smart grid, privacidad, seguridad, load forecasting, federated learning, blockchain**

I. Introducción

La predicción precisa de la demanda eléctrica constituye un elemento clave en las *smart grids* modernas, especialmente ante el incremento global del consumo [2]. La gestión eficiente de los recursos energéticos se complica aún más con la creciente adopción de recursos energéticos distribuidos (DERs). Aunque estos recursos contribuyen a la reducción de las emisiones y de la dependencia de combustibles fósiles, su variabilidad inherente aumenta la complejidad del equilibrio entre generación y demanda, reforzando la necesidad de técnicas avanzadas de *load forecasting* [3].

La *Advanced Metering Infrastructure* (AMI) facilita este proceso al recopilar datos detallados y en tiempo real del consumo energético de los usuarios. Diversos enfoques han demostrado mejoras significativas en la precisión de las predicciones [4], por ejemplo modelos de *deep learning* como las redes LSTM [3]. Sin embargo, el acceso a datos tan detallados plantea serios riesgos de privacidad: los patrones de consumo pueden revelar información sensible sobre los hábitos, presencia en el hogar y actividades diarias de los usuarios [5].

Para ello, resulta fundamental diseñar mecanismos que garanticen la confidencialidad de los datos y fomenten la participación segura de los usuarios. En este contexto, la tecnología *blockchain*, gracias a su naturaleza descentralizada y resistente a la manipulación, ofrece capacidades prometedoras para construir entornos de intercambio de datos confiables [6].

I-A. Trabajos relacionados

Los primeros trabajos en privacidad para *load forecasting*, como Soykan et al. [7], aplicaron *differential privacy* a modelos LSTM, pero dependían de un entrenamiento centralizado que introduce riesgos al concentrar los datos en un único agregador. Como alternativa, el *federated learning* evita compartir datos brutos y permite entrenar modelos de forma distribuida. Taik et al. [8] demostraron su viabilidad en smart grids, y posteriores estudios compararon variantes como FedSGD y FedAVG [9]. Aún así, la privacidad puede verse comprometida al inferirse información sensible a partir de las actualizaciones del modelo [10].

Diversas propuestas han tratado de reforzar la privacidad: Zhao et al. [11] incorporan *differential privacy* en las actualizaciones, aunque con pérdida de precisión; He et al. [12] aplican *k-means* para abordar la heterogeneidad, pero requieren compartir datos finos de consumo con el agregador; Savi et al. [13] y Tun et al. [14] intentan mitigar esta exposición compartiendo solo promedios o propiedades del hogar, aunque aún exponen información y requieren confiar en el agregador.

Para eliminar la dependencia del agregador durante el *clustering*, se han explorado métodos distribuidos. He et al. [15] presentan un enfoque jerárquico que mejora este aspecto. Sin embargo, no solucionan el problema de que se pueda inferir información a partir de las actualizaciones del modelo de los usuarios.

Otros trabajos emplean cifrado homomórfico y MPC para ocultar completamente las actualizaciones [16]. Aunque evita filtraciones, introduce un problema crítico: las actualizaciones cifradas no pueden atribuirse a usuarios concretos, impidiendo mecanismos de seguridad esenciales como Krum [17], cuya aplicación requiere inspeccionar los gradientes para detectar ataques de *model poisoning*.

Finalmente, Jebreel et al. [18] proponen un esquema basado en fragmentación de parámetros y reputación que protege parcialmente la privacidad. No obstante, asume la homogeneidad de datos y su sistema reputacional puede ser manipulado: un atacante puede enviar fragmentos maliciosos a usuarios honestos, provocando que los marque erróneamente como adversarios. Además, la gestión centralizada del registro de reputación concentra la confianza en el agregador, permitiendo comportamientos injustos como bloquear recompensas legíti-

mas.

I-B. Contribución

La principal contribución de este trabajo es la incorporación de un mecanismo de reputación dentro del federated learning, adaptado específicamente a entornos de smart grid. La propuesta aborda retos clave del *load forecasting*, como la heterogeneidad de datos, los riesgos de privacidad y la necesidad de colaboración confiable.

II. Visión general del sistema

Esta sección presenta los actores que intervienen, los requisitos principales y un resumen del funcionamiento del protocolo.

II-A. Actores y componentes

Los *prosumidores* (U_i) son los usuarios finales que poseen datos privados de consumo X_i, formados por distintos patrones eléctricos. Cada patrón diario E_d^i contiene los valores de carga del día d:

$$E_d^i = \begin{bmatrix} E_{d,1}^i & E_{d,2}^i & \dots & E_{d,T}^i \end{bmatrix}.$$

Los prosumidores entrenan localmente un modelo utilizando estos datos, evitando su revelación a terceros y preservando la desvinculación entre su identidad y las actualizaciones enviadas.

El objetivo del sistema es entrenar un modelo general de *load forecasting*. El modelo $\mathcal{M}(E_{d-1}^i, \theta)$ predice la demanda del día siguiente $Y_{i,d}$ a partir del perfil del día anterior:

$$Y_{i,d} = \mathcal{M}(E_{d-1}^i, \theta) = \begin{bmatrix} y_1 & y_2 & \dots & y_T \end{bmatrix}.$$

Los parámetros θ se ajustan minimizando el error cuadrático medio $MSE(\theta)$. En federated learning, cada usuario mantiene un modelo local $\mathcal{M}^i$ con parámetros θ^i, actualizados mediante descenso de gradiente $\theta^i \leftarrow \theta^i - \alpha \nabla MSE(\theta^i)$.

Tras las actualizaciones locales, se realiza la agregación global mediante FedAVG [19]:

$$\theta = \frac{1}{N} \sum_{i=1}^{N} \theta^i.$$

Para mejorar la precisión y tratar la heterogeneidad de datos, un *General Aggregator* (A_G) coordina un proceso distribuido de *clustering*, generando grupos de usuarios con características similares. Cada grupo es gestionado por un *Cluster Aggregator* (A_C), encargado de su entrenamiento federado.

El sistema integra además un mecanismo de reputación implementado mediante los siguientes *smart contracts*:

- *Reputation Smart Contract* ($\mathcal{RSC}$): Gestiona la puntuación de reputación de cada usuario.
- *Membership Smart Contract* ($\mathcal{MSC}_l$): Define los usuarios participantes en la ronda l, gestiona recompensas y actualiza reputaciones.
- *Bulletin Board Smart Contract* ($\mathcal{BBSC}$): Publica de forma transparente las recompensas emitidas por el proveedor del servicio.

II-B. Requisitos del sistema

El marco de federated learning propuesto para *load forecasting* debe cumplir los siguientes requisitos:

1. *Reputación dinámica*: Debe incorporarse un sistema de reputación que evalúe continuamente la fiabilidad de los participantes y oriente su selección.
2. *Canal anónimo*: Las comunicaciones no deben permitir vincular identidades con actualizaciones.
3. *Autenticación privada*: Los participantes deben autenticarse sin revelar su identidad real.
4. *Detección de actualizaciones maliciosas*: El *Cluster Aggregator* debe identificar intentos de *poisoning* sin que terceros accedan a los parámetros.
5. *Incentivos*: El sistema debe recompensar comportamientos honestos y penalizar contribuciones maliciosas.
6. *No falsificación de recompensas*: Las recompensas deben ser válidas únicamente si derivan de participación legítima.
7. *Prevención de doble gasto*: Cada recompensa debe canjearse una sola vez por ronda.
8. *Eficiencia en IoT*: El diseño debe ser viable en dispositivos con recursos limitados, como los smart meters.

II-C. Protocolo de Federated learning

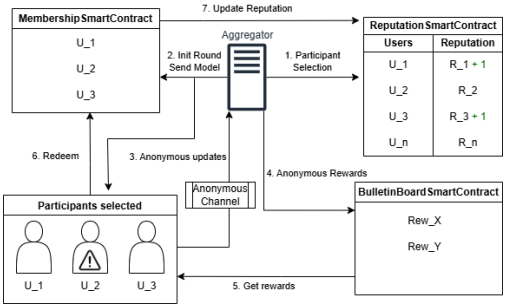


Figura 1. Visión general del protocolo propuesto.

Tras la fase de *clustering* distribuido coordinada por el agregador general (A_G), el protocolo continúa según los pasos mostrados en la Figura 1, involucrando a los *cluster aggregators* (A_C) y a los usuarios (U_i):

1. En cada ronda, A_C selecciona un subconjunto de usuarios basándose en sus puntuaciones de reputación almacenadas en $\mathcal{RSC}$.
2. A_C despliega un smart contract $\mathcal{MSC}$ para registrar a los usuarios seleccionados y envía el modelo inicial.
3. Cada U_i realiza el entrenamiento local y envía sus actualizaciones mediante un canal anónimo, junto con un token anónimo para reclamar la recompensa y una prueba de participación.
4. A_C verifica las actualizaciones y las pruebas recibidas. Si estas son válidas, firma el token correspondiente y lo publica en la $\mathcal{BBSC}$.
5. Cada U_i verificado como honesto puede recuperar su token firmado desde la $\mathcal{BBSC}$.
6. El usuario canjea el token en la $\mathcal{MSC}_l$, siguiendo un segundo protocolo de verificación para recibir la recompensa.
7. Finalmente, la reputación de cada U_i recompensado se actualiza en la $\mathcal{RSC}$.

III. Nuestra Propuesta

Esta sección describe de forma resumida el funcionamiento del sistema propuesto, estructurado en tres fases principales: clustering distribuido, inicialización del sistema y federated learning basado en reputación.

III-A. Paso-1: Clustering distribuido

El sistema emplea el esquema de clustering distribuido de [15] para agrupar a los usuarios sin compartir datos brutos. El agregador general (A_G) publica los centroides iniciales y, en cada iteración:

1. Cada usuario U_i asigna sus patrones diarios E_d^i al centroide más cercano.
2. Cada U_i calcula sus medias locales y envía las actualizaciones $\Delta_k^{i,l}$ y $n_k^{i,l}$ a A_G.
3. A_G actualiza los centroides.

El proceso finaliza por convergencia, y cada usuario se asigna al centroide más cercano según su patrón medio $\bar{E}^i$.

III-B. Paso-2: Inicialización del sistema

Tras el *clustering*, se inicializan los componentes necesarios:

1. Cada A_C despliega los smart contracts $\mathcal{RSC}$ y $\mathcal{BBSC}$.
2. Cada usuario genera un par de claves $pk_i = \mathcal{H}(sk_i)$ y se identifica mediante credenciales verificables.
3. A_C registra los pk_i en $\mathcal{RSC}$ e inicializa reputaciones.

III-C. Paso-3: Federated learning con reputación

El entrenamiento se desarrolla en rondas. Cada ronda l consta de cuatro subetapas.

III-C1. Paso-3.1: Inicialización de la ronda:

1. A_C genera un par de llaves EdDSA $(sk_{A_C}^l, PK_{A_C}^l)$.
2. Selecciona participantes según su reputación.
3. Genera un smart contract $\mathcal{MSC}_l$.
4. Registra los pk_i seleccionados en $\mathcal{MSC}_l$ mediante un Merkle tree.
5. Publica puntos aleatorios de la curva P_i' y envía el modelo global θ a cada U_i.

III-C2. Paso-3.2: Actualizaciones anónimas: Cada usuario entrena localmente su modelo y envía sus actualizaciones de forma anónima:

1. U_i construye un hashclaim $HC_i = \mathcal{H}(sk, P_i')$ y una prueba ZK π_i^P demostrando pertenencia al Merkle tree y conocimiento de sk_i sin revelar pk_i ni sk_i.
2. Reclama un punto P_i' en $\mathcal{MSC}_l$ desde una cuenta anónima.
3. Ciega el punto y genera un token t y su versión cegada t' utilizando el punto.
4. Construye otra ZK-proof π_i^T demostrando validez del token y posesión de sk_i.
5. Encripta la actualización θ^i con ECIES y envía (c_i, t', HC_i, π_i^T) a A_C por un canal anónimo.

III-C3. Paso-3.3: Actualización del modelo global y distribución de recompensas:

1. A_C desencripta c_i para obtener θ^i.
2. Verifica π_i^T y descarta aportaciones inválidas.
3. Detecta comportamientos maliciosos usando, por ejemplo, Krum [17].
4. Firma cada token cegado aceptado σ_i' y publica (t_i', σ_i') en $\mathcal{BBSC}$.

III-C4. Paso-3.4: Descegado del token y canjeado:

1. U_i recupera σ_i' y lo descega: σ.
2. Construye una ZK-proof π_i^R demostrando la validez de la firma sin revelar información sensible.
3. Envía $(P_i, pk_i, \sigma, \pi_i^R)$ a $\mathcal{MSC}_l$.
4. Si todo es válido, $\mathcal{MSC}_l$ marca el token como canjeado y actualiza la reputación del usuario en $\mathcal{RSC}$.

IV. Evaluación

Esta sección evalúa los requisitos de la Sección II-B.

Claim 1. *El sistema incorpora un mecanismo robusto de reputación almacenado de forma inmutable y resistente a manipulaciones. (R1)*

Proof: Las reputaciones se almacenan en $\mathcal{RSC}$, protegido por primitivas de blockchain: hashes resistentes a colisiones y firmas no falsificables. Solo se actualizan tras una redención válida en $\mathcal{MSC}_l$, lo que impide aumentos fraudulentos. ■

Claim 2. *El sistema es compatible con cualquier implementación de canal anónimo que asegure desvinculación entre identidad y mensaje. (R2)*

Proof: El protocolo requiere únicamente anonimato y unlinkability; no depende de propiedades específicas del canal. Por tanto, puede emplear cualquier tecnología de canal anónimo estado del arte que satisfaga estos requisitos. ■

Claim 3. *Cada participación en una ronda se verifica como proveniente de un usuario preseleccionado. (R3)*

Proof: Como se describe en el protocolo, cada usuario envía un token cegado y una prueba ZK que demuestra pertenencia al Merkle tree de $\mathcal{MSC}_l$ sin revelar su identidad. Solo los usuarios seleccionados pueden generar tokens válidos. ■

Claim 4. *El agregador no aprende ninguna información identificativa del participante durante la verificación. (R3)*

Proof: El uso de blinding y pruebas ZK asegura que el agregador verifica validez y pertenencia sin aprender pk_i, sk_i ni el vínculo entre token y usuario. ■

Claim 5. *El agregador puede detectar y filtrar actualizaciones maliciosas mediante análisis estadístico. (R4)*

Proof: Una vez descifradas, las actualizaciones permiten emplear técnicas robustas como Krum [17] para identificar gradientes anómalos y excluirlos del modelo. ■

Claim 6. *El sistema incentiva la participación honesta y penaliza el comportamiento malicioso. (R5)*

Proof: Las simulaciones (Figura 2) muestran que las reputaciones de usuarios honestos crecen y se estabilizan, mientras que los usuarios maliciosos acumulan menor reputación; además, su presencia en rondas sucesivas disminuye. ■

Claim 7. *Los tokens son infalsificables; solo un participante legítimo puede producir y canjear un token válido. (R6)*

Proof: Los tokens cegados t' se firman con EdDSA una llave privada. Sin conocerla, generar una firma válida es computacionalmente inviable. Hashes y firmas garantizan unicidad y autenticidad. ■

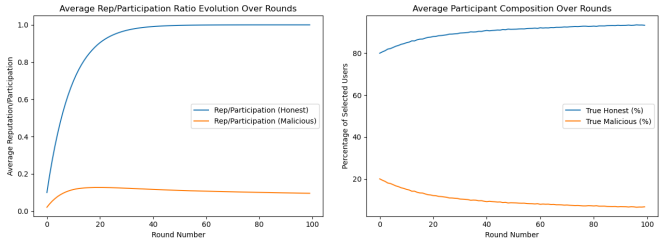


Figura 2. El primer gráfico muestra la evolución de la reputación, el segundo, la composición de los usuarios en cada ronda.

Claim 8. *Un usuario no puede gastar dos veces un token para recibir más de una recompensa por ronda. (R7)*

Proof: Cada $\mathcal{MSC}_l$ marca cada pk_i como "redeemed" tras el primer canje. Además, cada ronda emplea un $PK^l_{A_C}$ distinto, por lo que tokens entre rondas no son reutilizables. ■

Claim 9. *Las operaciones criptográficas son eficientes en dispositivos IoT. (R10)*

Proof: Las pruebas realizadas en una Raspberry Pi 3B (Tabla I) muestran que las operaciones principales (blind signatures, ZK proofs y verificación) finalizan en ∼23 s aun con 32 usuarios, con un crecimiento logarítmico, lo que confirma su viabilidad práctica. ■

# Users	Total
8 Users	22.39
16 Users	22.59
32 Users	22.99

Tabla I
TIEMPO DE EJECUCIÓN

V. CONCLUSIONES Y TRABAJO FUTURO

El trabajo propone un marco de federated learning seguro y preservador de la privacidad para *smart grids*, combinando anonimato, zero-knowledge proofs, blind signatures y un sistema de reputación basado en blockchain. El enfoque garantiza anonimato, integridad del modelo, detección de comportamientos maliciosos y rendimiento adecuado en dispositivos IoT. Los resultados muestran que el sistema gestiona eficazmente la heterogeneidad, incentiva la honestidad y reduce el impacto de participantes adversarios.

Como trabajo futuro, se planea implementar un entorno realista de federated learning para evaluar ataques avanzados y extender el marco a escenarios como *dynamic pricing*.

AGRADECIMIENTOS

Esta investigación está financiada por el Proyecto HERMES, financiado por la Unión Europea NextGenerationEU/PRTR a través de INCIBE; por los Proyectos PID2021-125962OB-C32 ≪SECURING/DATA≫ y PID2024-156914OB-C42 ≪SAFE/DATA≫, financiados por MCIN/AEI/10.13039/501100011033; y por la Ayuda SGR2021-00115, financiada por AGAUR, Generalitat de Catalunya. Joan Ferré-Queralt cuenta con el apoyo del Gobierno de España mediante una beca FPU (ref. FPU23/01275). Alexandre Viejo es Serra-Húnter Fellow.

REFERENCIAS

[1] J. Ferré-Queralt, J. Castellà-Roca, and A. Viejo, "Secure and privacy-preserving load forecasting in smart grids through blockchain-based incentives," in *International Conference on Network and System Security*. Springer, 2025, p. TOAPPEAR.

[2] I. E. Agency, "Global electricity demand rising in 2021," *IEA*, 2021. [Online]. Available: https://www.iea.org/reports/global-energy-review-2021

[3] N. Ahmad, Y. Ghadi, M. Adnan, and M. Ali, "Load forecasting techniques for power system: Research challenges and survey," *IEEE Access*, vol. 10, pp. 71 054–71 090, 2022.

[4] N. Amral, C. Ozveren, and D. King, "Short term load forecasting using multiple linear regression," in *2007 42nd International universities power engineering conference*. IEEE, 2007, pp. 1192–1198.

[5] Z. Bilgin, E. Tomur, M. A. Ersoy, and E. U. Soykan, "Statistical appliance inference in the smart grid by machine learning," in *2019 IEEE 30th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC Workshops)*. IEEE, 2019, pp. 1–7.

[6] M. B. Mollah, J. Zhao, D. Niyato, K.-Y. Lam, X. Zhang, A. M. Ghias, L. H. Koh, and L. Yang, "Blockchain for future smart grid: A comprehensive survey," *IEEE Internet of Things journal*, vol. 8, no. 1, pp. 18–43, 2020.

[7] E. U. Soykan, Z. Bilgin, M. A. Ersoy, and E. Tomur, "Differentially private deep learning for load forecasting on smart grid," in *2019 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2019, pp. 1–6.

[8] A. Taïk and S. Cherkaoui, "Electrical load forecasting using edge computing and federated learning," in *ICC 2020-2020 IEEE international conference on communications (ICC)*. IEEE, 2020, pp. 1–6.

[9] M. N. Fekri, K. Grolinger, and S. Mir, "Distributed load forecasting using smart meter data: Federated learning with recurrent neural networks," *International Journal of Electrical Power & Energy Systems*, vol. 137, p. 107669, 2022.

[10] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE symposium on security and privacy (SP)*. IEEE, 2019, pp. 691–706.

[11] Y. Zhao, W. Xiao, L. Shuai, J. Luo, S. Yao, and M. Zhang, "A differential privacy-enhanced federated learning method for short-term household load forecasting in smart grid," in *2021 7th International Conference on Computer and Communications (ICCC)*. IEEE, 2021, pp. 1399–1404.

[12] Y. He, F. Luo, G. Ranzi, and W. Kong, "Short-term residential load forecasting based on federated learning and load clustering," in *2021 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*. IEEE, 2021, pp. 77–82.

[13] M. Savi and F. Olivadese, "Short-term energy consumption forecasting at the edge: A federated learning approach," *IEEE Access*, vol. 9, pp. 95 949–95 969, 2021.

[14] Y. L. Tun, K. Thar, C. M. Thwal, and C. S. Hong, "Federated learning based energy demand prediction with clustered aggregation," in *2021 IEEE International Conference on Big Data and Smart Computing (BigComp)*. IEEE, 2021, pp. 164–167.

[15] Y. He, F. Luo, M. Sun, and G. Ranzi, "Privacy-preserving and hierarchically federated framework for short-term residential load forecasting," *IEEE Transactions on Smart Grid*, vol. 14, no. 6, pp. 4409–4423, 2023.

[16] J. Lei, L. Wang, Q. Pei, W. Sun, X. Lin, and X. Liu, "Privgrid: Privacy-preserving individual load forecasting service for smart grid," *IEEE Transactions on Information Forensics and Security*, 2024.

[17] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Advances in neural information processing systems*, vol. 30, 2017.

[18] N. M. Jebreel, J. Domingo-Ferrer, A. Blanco-Justicia, and D. Sánchez, "Enhanced security and privacy via fragmented federated learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[19] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *20th International Conference on Artificial In telligence and Statistics (AISTATS)*, 2017.

Resumen extendido de "Calyptography: Secure Secret Storage Inspired by Cryptography and Steganography"

Daniel Lerch-Hostalot*, Jordi Puiggalí†, David Megías*
*Centro de Investigación UOC-TECH, Estudios de Informática, Multimedia y Telecomunicación,
Universitat Oberta de Catalunya (UOC)
CYBERCAT - Center for Cybersecurity Research of Catalonia
E-mail: {dlerch, dmegias}@uoc.edu
†SecretsVault.xyz - Limitless Technologies and Applications, S.L.
E-mail: jordi.puiggali@secretsvault.xyz

***Resumen*—Este *resumen extendido* compila y sintetiza los resultados publicados en [1] sobre *caliptografía*, un nuevo enfoque para el almacenamiento seguro de secretos inspirado en la criptografía y la esteganografía, pero sin modificar el medio portador. La idea central consiste en vincular un secreto a una imagen de referencia mediante un *hash perceptual robusto*, combinando sus características visuales con una *frase secreta* del usuario para derivar las claves de cifrado y descifrado. Se presenta una arquitectura basada en DCT por bloques, muestreo aleatorio de parches y XOR bit a bit con codificación BCH a nivel de mensaje, junto con una evaluación experimental de la tasa de error de bit (BER) frente a compresión JPEG, filtros fotográficos y redimensionamiento. Además, se aportan nuevos resultados obtenidos al testear la metodología con bases de datos generadas sintéticamente mediante Stable Diffusion, confirmando la robustez y generalización del enfoque frente a variaciones de contenido y estilo visual. Los experimentos muestran alta robustez (por ejemplo, BER inferior al 0,20 % con JPEG de calidad 50 en imágenes naturales e inferior al 0,32 % en imágenes generadas) en comparación con métodos tradicionales de *hashing* perceptual, consolidando a la caliptografía como una alternativa práctica y usable para la gestión de secretos, preservando la integridad del medio.**

***Index Terms*—Almacenamiento seguro, *Hash* perceptual, *Hash* robusto, Procesamiento de imágenes**

I. Introducción

El almacenamiento seguro de información sensible, o secretos, es uno de los problemas fundamentales de la seguridad de la información. Tradicionalmente, esta necesidad se ha abordado mediante dos grandes enfoques: la criptografía y la esteganografía. La criptografía transforma el mensaje original en un texto cifrado ilegible mediante una clave secreta, garantizando la confidencialidad del contenido siempre que la clave se mantenga protegida [2]. La esteganografía, por su parte, busca ocultar la propia existencia del mensaje, incrustándolo en un medio portador aparentemente inocuo, como una imagen, un audio o un texto, sin que se perciban modificaciones perceptibles ni estadísticas significativas [3].

Pese a sus logros, ambos enfoques presentan limitaciones en escenarios de almacenamiento personal o vinculación de secretos a objetos digitales. En la criptografía convencional, la clave depende únicamente de la memoria o gestión del usuario, lo que dificulta su recuperación y propicia vulnerabilidades por mala gestión. En la esteganografía clásica, el mensaje se oculta alterando el medio portador, lo que la hace susceptible al análisis forense o estegoanálisis. En respuesta, se han propuesto enfoques alternativos como la esteganografía *coverless* [4], que evita modificar la imagen y asocia el mensaje con propiedades visuales preexistentes (brillo, textura, contornos o descriptores semánticos), o la esteganografía por selección de portador [5], que escoge una imagen de entre muchas posibles para representar el mensaje. Sin embargo, estos métodos se enfrentan a limitaciones de capacidad, complejidad y escalabilidad.

Paralelamente, los métodos de *hash* perceptual y/o robusto se han consolidado como herramientas eficaces para identificar imágenes de forma tolerante a transformaciones comunes, como compresión, filtrado o redimensionamiento. Frente a los *hashes* criptográficos tradicionales, extremadamente sensibles a pequeñas variaciones, los perceptuales producen descriptores similares para imágenes visualmente parecidas. Entre las técnicas más relevantes se encuentran Average Hash (A-Hash), Difference Hash (D-Hash), Perceptual Hash (P-Hash) y Wavelet Hash (W-Hash) [6], [7], [8]. También se han propuesto métodos más avanzados basados en transformadas robustas [9], [10], [11], en combinaciones híbridas como DCT-SVD o DWT-SVD [12], o en modelos de aprendizaje profundo [13], [14]. Estas técnicas son capaces de extraer características estables ante modificaciones benignas, lo que las hace especialmente atractivas para usos en autenticación, análisis forenses o autenticación multimedia [15].

El artículo original introduce la caliptografía [1], un nuevo paradigma de almacenamiento seguro de secretos que combina ideas de la criptografía y la esteganografía, pero sin modificar el medio portador. En este enfoque, el secreto se asocia a una imagen de referencia mediante un *hash* perceptual robusto y una contraseña del usuario, de modo que su recuperación depende tanto de algo que se sabe como de algo que se posee. Este enfoque supone una mejora práctica respecto al multifactor clásico, al permitir que el factor de posesión se almacene y distribuya de forma natural en plataformas públicas como redes sociales, sin necesidad de infraestructura adicional. En el siguiente apartado de este resumen extendido se presentan el concepto y los fundamentos técnicos de dicha propuesta.

Las contribuciones de este resumen extendido son: (i) sintetizar el concepto de caliptografía introducido en [1], (ii)

describir el algoritmo de generación de *hash* dependiente de la imagen y *passphrase*, (iii) resumir los resultados de robustez sobre ALASKA [16], y (iv) añadir una evaluación preliminar sobre imágenes generadas con Stable Diffusion [17].

Por razones de espacio, algunos detalles y resultados descritos en [1] no se incluyen aquí. El lector interesado puede consultar dicho trabajo para una descripción completa.

II. Caliptografía: concepto y fundamentos

La *caliptografía* se presenta como un nuevo paradigma para el almacenamiento y la protección de información sensible, inspirado en los principios de la criptografía y la esteganografía, pero con un mecanismo operativo distinto. Su propósito fundamental es permitir que un secreto se asocie a un medio de referencia —por ejemplo, una imagen— sin que dicho medio sufra modificaciones y sin que se revele la existencia de la información protegida.

El término proviene de las palabras griegas καλύπτω (cubrir, ocultar) y γράφειν (escribir o representar), reflejando la idea de "escritura oculta" mediante la asociación entre un secreto y un contenido perceptual. En este contexto, la caliptografía actúa como un puente conceptual entre la criptografía, centrada en la transformación matemática de los datos, y la esteganografía, orientada a su disimulación dentro de un medio. Sin embargo, se diferencia de ambas al no depender exclusivamente de claves arbitrarias ni de la modificación del portador, sino del aprovechamiento de las propiedades perceptuales estables de una imagen como parte del proceso criptográfico.

En esencia, el sistema caliptográfico (ilustrado en la Fig. 1) utiliza el *hash perceptual robusto* de una imagen como componente clave para derivar las claves de cifrado y descifrado de un secreto. Esto permite que la protección dependa tanto de un conocimiento (por ejemplo, una contraseña) como de la posesión o disponibilidad de una imagen concreta o visualmente equivalente. De esta forma, el medio visual se convierte en un factor adicional de autenticación, mejorando la seguridad y reduciendo la exposición del secreto ante accesos no autorizados.

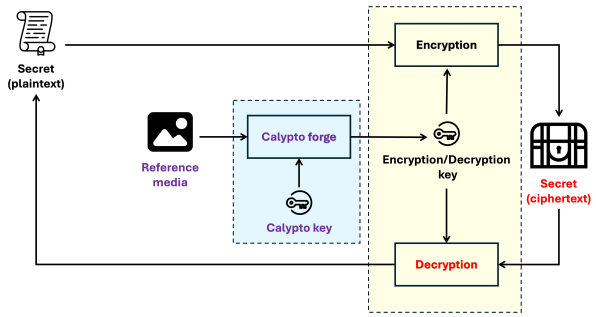


Fig. 1: Diagrama de los procesos caliptográficos. La imagen de referencia y la clave caliptográfica se combinan para obtener la clave de cifrado o descifrado del secreto.

Aunque la caliptografía comparte fundamentos conceptuales con la criptografía y la esteganografía, su filosofía y objetivos prácticos difieren en varios aspectos clave.

En la criptografía, la confidencialidad de la información se garantiza mediante operaciones matemáticas que transforman el mensaje en un texto cifrado, dependiente de una clave secreta. La seguridad se apoya en la dificultad computacional de invertir este proceso sin conocer la clave. Sin embargo, el sistema no oculta la existencia del mensaje cifrado, ya que el texto cifrado es fácilmente identificable como tal.

Por el contrario, la esteganografía persigue disimular la existencia del mensaje, ocultándolo dentro de un medio aparentemente inocuo. A pesar de su capacidad para evadir la detección visual, los métodos esteganográficos tradicionales presentan vulnerabilidades frente a técnicas de estegoanálisis, que pueden identificar patrones estadísticos o alteraciones mínimas introducidas en el portador. Además, la necesidad de modificar el medio puede afectar su integridad o calidad.

La caliptografía, en cambio, combina las ventajas de ambos enfoques al emplear el medio visual como una clave perceptual, sin modificar su estructura. En lugar de incrustar el mensaje dentro de la imagen, utiliza sus características perceptuales para derivar o reconstruir la clave de acceso al secreto cifrado. Así, la imagen actúa como un *token* o "segundo factor" de autenticación, cuya pérdida o alteración perceptual significativa podría invalidar el acceso al secreto, pero sin comprometer la confidencialidad de este.

III. Método propuesto

El artículo original describe con detalle el procedimiento técnico que permite vincular el secreto con una imagen mediante la generación de un *hash* perceptual robusto, combinando las características visuales de la imagen con la *passphrase* del usuario. El proceso completo se resume en los Algoritmos 1 y 2, que también se incluyen en este resumen extendido.

El primer componente, representado en el Algoritmo 1, se encarga de generar el *hash* de la imagen. Para ello, la imagen se convierte primero a escala de grises y se redimensiona a un tamaño mínimo común que garantice consistencia entre diferentes resoluciones. A partir de esta imagen se seleccionan aleatoriamente varios parches de gran tamaño, de forma determinista mediante un generador pseudoaleatorio inicializado con una semilla derivada de la *passphrase*. Cada parche se redimensiona a una resolución menor y se divide en bloques de 8×8 píxeles, sobre los que se aplica la Transformada Discreta del Coseno (DCT). De cada bloque se extrae el coeficiente DC, que contiene la información de baja frecuencia más estable frente a transformaciones comunes como la compresión JPEG.

A continuación, la lista de coeficientes obtenida se reordena aleatoriamente de nuevo empleando la misma semilla, y de la comparación de valores consecutivos se generan los bits del hash final. Si el valor de un bloque es mayor que el siguiente, se asigna un 1; en caso contrario, un 0, Este procedimiento, basado en comparaciones relativas y no en umbrales absolutos, aporta una mayor robustez frente a variaciones ligeras de brillo, contraste o compresión. Además, la combinación de parches seleccionados aleatoriamente y el barajado dependiente de la contraseña introduce entropía adicional y hace que el *hash* resultante dependa simultáneamente de la imagen y de la *passphrase*.

El Algoritmo 2 describe el segundo componente del método, responsable del cifrado y descifrado del secreto. A partir del *hash* generado, se deriva una clave binaria que se utiliza para proteger el mensaje mediante una operación bit a bit (por ejemplo, XOR). Antes del cifrado, el mensaje se codifica

con un esquema de corrección de errores, como BCH o Reed–Solomon, de modo que el sistema pueda recuperar el secreto incluso si el *hash* extraído presenta pequeñas diferencias respecto al original debido a transformaciones de la imagen. Durante el descifrado, la misma imagen y contraseña permiten regenerar el *hash,* reconstruir la clave y, tras aplicar la decodificación del código corrector, obtener de nuevo el mensaje original.

El artículo original demuestra que este esquema basado en DCT por bloques y selección aleatoria de parches mantiene un bajo ratio (o tasa) de error de bits (*Bit Error Rate,* BER) frente a compresión, filtrado y redimensionamiento, superando en robustez a métodos perceptuales clásicos como A-Hash, D-Hash, P-Hash o W-Hash. Esta estabilidad del *hash* es esencial para garantizar que la recuperación del secreto sea fiable, aun cuando la imagen de referencia haya sido transformada o procesada levemente.

Algorithm 1: Image-based Hash Generation

Input: RGB image I, passphrase P
Output: Binary hash string H
// Preprocessing
$I \leftarrow \texttt{grayscale}(I)$
$I \leftarrow \texttt{resize_to_min}(I, \texttt{MIN_IMAGE_SIZE})$
$seed \leftarrow \texttt{sha256}(P)$
$\texttt{init_prng}(\text{seed})$
// Patch extraction and transformation
$patches \leftarrow \texttt{random_patches}(I, \texttt{LargePatchSize}, N)$
$V \leftarrow$ empty list
foreach $patch \in patches$ **do**
  $patch \leftarrow \texttt{resize}(patch, \texttt{SmallPatchSize}, \texttt{SmallPatchSize})$
  $blocks \leftarrow \texttt{divide}(patch, 8 \times 8)$
  foreach $block \in blocks$ **do**
    $D \leftarrow \texttt{dct2}(block)$
    $V \leftarrow V \cup \{D_{0,0}\}$
  end
end
// Hash generation
$V \leftarrow \texttt{shuffle}(V, seed)$
$H \leftarrow \texttt{compare_pairs}(V)$
return H

Algorithm 2: Encrypt/Decrypt using Image-Based Hash and ECC

Input: Hash H, passphrase P, message M, mode $\in \{\text{encrypt}, \text{decrypt}\}$
Output: Encrypted or decrypted message
// Key derivation
$seed \leftarrow \texttt{sha256}(P)$
$\texttt{init_prng}(\text{seed})$
$key \leftarrow \texttt{hash_to_key}(H, seed)$
if $mode = encrypt$ **then**
  // Apply error correction encoding
  $M_{ecc} \leftarrow \texttt{encode_ecc}(M)$
  // Encrypt the message with the derived key
  $C \leftarrow \texttt{bitwise_XOR}(M_{ecc}, key)$
  return C
else
  // Decrypt the ciphertext
  $M_{ecc} \leftarrow \texttt{bitwise_XOR}(M, key)$
  // Attempt to correct errors
  $M' \leftarrow \texttt{decode_ecc}(M_{ecc})$
  return M'
end

IV. Evaluación experimental

Con el objetivo de analizar la robustez del *hash* perceptual utilizado en el método descrito, el artículo original llevó a cabo una serie de experimentos comparando diferentes estrategias de *hashing* frente a transformaciones comunes en imágenes digitales. El propósito fue evaluar la robustez ante operaciones como compresión JPEG, filtrado y redimensionamiento, identificando la configuración más resistente.

Las pruebas se centraron exclusivamente en la generación del *hash* (Algoritmo 1), sin incluir las etapas de cifrado ni de corrección de errores. La robustez se midió mediante el BER, calculado entre el *hash* original y el obtenido tras aplicar una transformación a la imagen. Esta métrica, expresada en porcentaje, permite comparar de forma uniforme la sensibilidad de cada método. En todos los casos, los *hashes* se truncaron a 128 o 64 bits para mantener la comparabilidad con los métodos clásicos A-Hash, D-Hash, P-Hash y W-Hash. Nótese que el truncado tiene como fin la comparabilidad experimental y no afecta a la seguridad del método; en implementaciones reales se usa un *hash* de 386 bits con códigos BCH (127, 92,5) y (63,38,5) para claves de 256 bits (https://secretsvault.xyz/).

El conjunto experimental se basó en 5.000 imágenes en formato RAW del ALASKA *Steganography Detection Challenge* [16], recortadas y redimensionadas a 1080×1080 píxeles para garantizar condiciones homogéneas. Este corpus es ampliamente utilizado en la evaluación de esteganografía y hashing perceptual por su diversidad y ausencia de artefactos de compresión.

Además, se generó un segundo conjunto de 1.000 imágenes sintéticas mediante el modelo *Stable Diffusion* [17], a partir de descripciones textuales aleatorias, con el fin de incorporar variabilidad visual y evaluar el comportamiento del método frente a contenido generado por inteligencia artificial.

La combinación de ambos conjuntos permitió comprobar la consistencia del enfoque propuesto en escenarios realistas y heterogéneos, confirmando que el *hash* basado en DCT y selección aleatoria de parches mantiene una baja tasa de error frente a compresión, filtrado y redimensionamiento, y supera en estabilidad a los métodos perceptuales tradicionales.

La Tabla I resume los resultados obtenidos en las pruebas comparativas entre el método propuesto y los algoritmos de *hashing* perceptual más utilizados: A-Hash, D-Hash, P-Hash y W-Hash. En todos los casos se midió el BER ante tres tipos de transformaciones: compresión JPEG, aplicación de filtros fotográficos y redimensionamiento. Se presentan los resultados para dos conjuntos de datos ALASKA, compuesto por imágenes reales en formato RAW, y *Stable Diffusion* (SD), formado por imágenes generadas sintéticamente.

En el conjunto ALASKA, el método propuesto muestra la menor tasa de error en todos los escenarios, especialmente bajo compresión JPEG severa (factores de calidad 70 y 50), donde mantiene un BER por debajo del 0,20 %, muy inferior al de los métodos tradicionales. Frente a la aplicación de filtros de imagen, aunque todos los métodos muestran una ligera degradación, el propuesto conserva una estabilidad notable, manteniendo un BER próximo al 0,8 % incluso con transformaciones de alto impacto visual. De forma similar, ante redimensionamientos (factores de 0,5 a 1,5), la variación del hash es mínima, con valores inferiores al 0,1 %.

En el conjunto de imágenes sintéticas generadas por *Stable Diffusion*, el método conserva también un comportamiento robusto, aunque con una ligera pérdida de estabilidad debido a las variaciones de textura y contraste características del contenido generado artificialmente. En este caso, los valores de BER se mantienen por debajo del 0,33 % bajo compresión, del 2,08 % con filtros fotográficos y del 0,18 % con redimensionamiento, lo que confirma la capacidad del algoritmo para generalizar más allá de imágenes naturales.

En conjunto, los resultados demuestran que el enfoque basado en DCT y selección aleatoria de parches propuesto en el artículo original supera claramente a los métodos per-

Tabla I: BER [%] de distintos métodos de hashing frente a compresión JPEG, filtrado y redimensionamiento

Método	Factor de Calidad				Filtro					Factor Redim.			
	100	**80**	**70**	**50**	**Grayscale**	**Sepia**	**Clarendon**	**Lark**	**Juno**	**1,5**	**0,9**	**0,7**	**0,5**
A-Hash	0,16	0,21	0,25	0,33	0,78	1,16	1,28	0,95	1,21	0,14	0,18	0,19	0,22
D-Hash	0,47	0,65	0,77	1,04	1,94	1,84	2.30	1,65	1,87	0,41	0,51	0,55	0,62
P-Hash	0,34	0,46	0,56	0,62	0,48	2.50	3.07	1,80	2.61	0,30	0,36	0,40	0,42
W-Hash	0,11	0,20	0,22	0,30	0,03	0,62	0,72	0,61	0,56	0,08	0,11	0,15	0,17
Propuesto (ALASKA)	**0,05**	**0,11**	**0,15**	**0,20**	**0,01**	**0,60**	**0,80**	**0,56**	**0,62**	**0,04**	**0,06**	**0,07**	**0,09**
Propuesto (SD)	**0,05**	**0,16**	**0,21**	**0,32**	**0,04**	**1,13**	**2.07**	**1,16**	**1,34**	**0,08**	**0,11**	**0,13**	**0,17**

ceptuales clásicos tanto en imágenes reales como sintéticas. Su capacidad para mantener un *hash* estable ante transformaciones comunes confirma su idoneidad como componente central de la caliptografía, garantizando la recuperación fiable de secretos incluso cuando la imagen de referencia ha sido modificada o procesada.

V. Conclusiones

Este resumen ha sintetizado la propuesta de caliptografía como enfoque intermedio entre criptografía y esteganografía, orientado al almacenamiento seguro de secretos sin modificar el medio portador. La técnica emplea un *hash* perceptual robusto, generado mediante DCT por bloques y muestreo aleatorio dependiente de la contraseña, que actúa como material de clave y exige tanto la imagen como la *passphrase* para la recuperación del secreto.

Los algoritmos presentados permiten una implementación modular: el *hash* perceptual se usa para derivar la clave junto con un código de corrección de errores, de modo que pequeñas alteraciones por compresión, filtrado o redimensionamiento no impiden descifrar el mensaje.

Los experimentos confirman la superioridad del método frente a A-Hash, D-Hash, P-Hash y W-Hash, tanto en el conjunto ALASKA como en imágenes sintéticas, donde la BER sigue siendo corregible.

Como trabajo futuro se propone mejorar la robustez ante transformaciones geométricas y explorar variantes de *hashing* perceptual basadas en aprendizaje profundo. En conjunto, la caliptografía se presenta como una alternativa práctica para la gestión segura de secretos usando medios visuales.

Finalmente, cabe destacar que la caliptografía ha sido validada en un entorno real mediante su adopción por la empresa *Secrets Vault*, que emplea este enfoque para la protección y recuperación de secretos digitales a partir de imágenes como factor de posesión, demostrando su viabilidad práctica más allá del ámbito académico. Más información en https://secretsvault.xyz/.

Agradecimientos

Los autores agradecen la financiación obtenida a través de los proyectos “SAFE” (PID2024-156914OB-C41) y “SECURING” (PID2021-125962OB-C31) financiados por el Ministerio de Ciencia, Innovación y Universidades mediante la Agencia Estatal de Investigación (AEI) y el Fondo Europeo de Desarrollo Regional (FEDER), así como la Cátedra Internacional de Ciberseguridad ARTEMISA (C057/23) y el Proyecto Estratégico de Ciberseguridad DANGER (C062/23), ambos financiados por el Instituto Nacional de Ciberseguridad a través de la Unión Europea – NextGenerationEU y el Plan de Recuperación, Transformación y Resiliencia.

Referencias

[1] D. Lerch-Hostalot, J. Puiggalí, and D. Megías, “Calyptography: Secure secret storage inspired by cryptography and steganography,” in *Availability, Reliability and Security: ARES 2025 International Workshops, Ghent, Belgium, August 11–14, 2025, Proceedings, Part III*. Berlin, Heidelberg: Springer-Verlag, 2025, p. 290–308.

[2] J. Katz and Y. Lindell, *Introduction to modern cryptography: principles and protocols*. Chapman and hall/CRC, 2007.

[3] F. Petitcolas, R. Anderson, and M. Kuhn, “Information hiding-a survey,” *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, 1999.

[4] J. Qin, Y. Luo, X. Xiang, Y. Tan, and H. Huang, “Coverless image steganography: A survey,” *IEEE Access*, vol. 7, pp. 171 372–171 394, 2019.

[5] Z. Zhou, H. Sun, R. Harit, X. Chen, and X. Sun, “Coverless image steganography without embedding,” in *Cloud Computing and Security: First International Conference, ICCCS 2015, Nanjing, China, August 13-15, 2015. Revised Selected Papers 1*. Springer, 2015, pp. 123–132.

[6] A. Drmic, M. Silic, G. Delac, K. Vladimir, and A. S. Kurdija, “Evaluating robustness of perceptual image hashing algorithms,” in *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2017, pp. 995–1000.

[7] M. Hamadouche, K. Zebbiche, M. Guerroumi, H. Tebbi, and Y. Zafoune, “A comparative study of perceptual hashing algorithms: Application on fingerprint images,” 2021.

[8] C. Zauner, “Implementation and benchmarking of perceptual image hash functions,” Master’s thesis, Upper Austria University of Applied Sciences, Hagenberg Campus, 2010.

[9] R. Venkatesan, S.-M. Koon, M. Jakubowski, and P. Moulin, “Robust image hashing,” in *Proceedings 2000 International Conference on Image Processing (Cat. No.00CH37101)*, vol. 3, 2000, pp. 664–666 vol.3.

[10] Y. N. Li, P. Wang, and Y. T. Su, “Robust image hashing based on selective quaternion invariance,” *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2396–2400, 2015.

[11] C.-P. Yan, C.-M. Pun, and X.-C. Yuan, “Quaternion-based image hashing for adaptive tampering localization,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 12, pp. 2664–2677, 2016.

[12] S. Kozat, R. Venkatesan, and M. Mihcak, “Robust perceptual image hashing via matrix invariants,” in *2004 International Conference on Image Processing, 2004. ICIP ’04.*, vol. 5, 2004, pp. 3443–3446 Vol. 5.

[13] L. Wu, Y. Wang, Z. Ge, Q. Hu, and X. Li, “Structured deep hashing with convolutional neural networks for fast person re-identification,” *Computer Vision and Image Understanding*, vol. 167, pp. 63–73, 2018.

[14] Y. Peng, J. Zhang, and Z. Ye, “Deep reinforcement learning for image hashing,” *IEEE Transactions on Multimedia*, vol. 22, no. 8, pp. 2061–2073, 2020.

[15] L. Du, A. T. Ho, and R. Cong, “Perceptual hashing for image authentication: A survey,” *Signal Processing: Image Communication*, vol. 81, p. 115713, 2020.

[16] R. Cogranne, Q. Giboulot, and P. Bas, “The ALASKA Steganalysis Challenge: A First Step Towards Steganalysis,” in *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, ser. IH&MMSec’19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 125–137.

[17] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “ High-Resolution Image Synthesis with Latent Diffusion Models ,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2022, pp. 10 674–10 685.

Detección de Contenido Sexual en Dispositivos Móviles para la Preservación de la Privacidad

Sandra Pérez Arteaga, Ana Lucila Sandoval Orozco, Luis Javier García Villalba*

Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {sandrp08, asandov, javierv}@ucm.es

***Resumen*—La detección de contenido sexual en imágenes en dispositivos móviles es una herramienta clave para la protección de los usuarios, especialmente en la prevención del intercambio de material sensible o privado. Este análisis plantea desafíos significativos vinculados a la seguridad y privacidad de los datos implicados. Este trabajo realizara un análisis del modelo de Machine Learning adecuado para ser implementado en un dispositivo móvil y realizar un análisis de imágenes de contenido sexual dentro del dispositivo.**

El análisis de las imágenes se realizará en un procesamiento local en el dispositivo, implementando el modelo y la arquitectura Training and testing on the devices, esto reducirá la necesidad de transferir datos a servidores externos. Estas técnicas permiten reducir significativamente los riesgos de exposición de datos personales asegurando que los datos procesados no puedan ser utilizados para identificar a los usuarios o revelar información privada.

***Index Terms*—Arquitectura, Dispositivos Móviles, Imágenes, Machine Learning, MobileNet, Privacidad.**

Tipo de contribución: *Investigación original.*

I. Introducción

La conexión constante a Internet ha transformado los dispositivos móviles en herramientas esenciales. Si bien facilitan la comunicación, el trabajo y las tareas escolares, esta hiperconectividad ha convertido a las redes sociales en plataformas centrales para compartir aspectos íntimos y sensibles de la vida diaria, desde la ubicación hasta actividades cotidianas. Esta práctica, aunque habitual, conlleva un riesgo inherente para la integridad de la información y la seguridad del usuario, exponiéndolos a una variedad de riesgos y delitos digitales.

Un aspecto crítico y creciente en este entorno es la interacción en estas plataformas. Al estar conectados a múltiples redes sociales, los usuarios están potencialmente expuestos a individuos con intenciones maliciosas. Un fenómeno particularmente alarmante es la solicitud o, más concretamente, la recepción de imágenes de contenido sexual no solicitado a través de las diferentes redes sociales directamente en los dispositivos móviles.

Este hecho no solo vulnera la privacidad y el consentimiento del usuario, sino que también puede ser el inicio de formas de control, extorsión o la posterior difusión no consentida de dichas imágenes. Es crucial analizar cómo la facilidad de intercambio de archivos en las plataformas sociales y el acceso constante a estas a través de smartphones amplifican el riesgo de recibir contenido sexual explícito por parte de terceros, poniendo en serio peligro la estabilidad emocional y digital de las víctimas.

El uso de la Inteligencia Artificial (IA) en dispositivos móviles es fundamental, ya que su integración mejora significativamente las funcionalidades, haciendo que estos sean más inteligentes, eficientes y rápidos, lo que optimiza la experiencia del usuario y los convierte en dispositivos electrónicos verdaderamente inteligentes. La IA avanzada permite a los móviles reconocer el comportamiento de los usuarios y comprender sus necesidades [1].

En este contexto de hiperconectividad y riesgos digitales previamente descritos, el presente trabajo se enfoca en proteger la información que los usuarios almacenan en sus dispositivos móviles. Para ello, se propone el desarrollo de un análisis de imágenes, tanto recibidas como almacenadas en el móvil, utilizando algoritmos de Machine Learning. El objetivo es verificar de manera automática la presencia de contenido sexual en dichas imágenes. Este análisis busca no solo preservar la seguridad y privacidad de los datos, sino también verificar la viabilidad y el rendimiento del uso de algoritmos de inteligencia artificial directamente en los dispositivos móviles para este fin. Con este propósito, se han realizado pruebas utilizando clasificadores de imágenes que permiten medir el rendimiento de los algoritmos bajo diferentes supuestos y escenarios de uso del dispositivo.

Todo el procesamiento de datos se realiza directamente en el dispositivo basados en la arquitectura de Machine Learning on Mobile Devices específicamente en la arquitectura de entrenamiento y testeo dentro del dispositivo, ayudando a evitar los costes y los requisitos de ancho de banda asociados al uso de servicios en la nube y no requerir conexión a servidores externos, lo que garantiza un alto nivel de seguridad y confidencialidad de la información, eliminando el riesgo de exposición de la información sensible a analizar. Además, el procesamiento puede permitir a los usuarios identificar y recibir alertas sobre contenido potencialmente sensible o de carácter sexual antes de su envío o recepción de la información, brindando una barrera preventiva eficaz contra el cibercrimen.

El resto del trabajo se organiza como sigue: en la Sección II describe trabajos relacionados con la detección de imágenes sexuales, así como trabajos que implementan los modelos en dispositivos de bajos recursos, como los móviles, y su rendimiento. En la Sección III escribe los modelos utilizados

para la clasificación de imágenes sexuales. En la Sección IV se muestran los resultados del rendimiento de los modelos implementados en el dispositivo móvil y, finalmente, en la Sección V se incluyen las conclusiones del trabajo.

II. Trabajos relacionados

Se desarrolló en 1996 una herramienta [2] que era capaz de detectar cuerpos desnudos mediante el análisis del color y la textura de la piel. Para determinar si los grupos de píxeles correspondían al cuerpo humano, se implementó un modelo adecuado y efectivo que lograba identificar formas humanas. El sistema obtuvo un 52 % de recuperación y un 60 % de precisión, resultados aceptables para la época; sin embargo, el tiempo requerido para detectar contenido explícito en una imagen superaba los seis minutos, lo cual no cumplía con las expectativas de eficiencia temporal. Evaluar si una imagen era sexual o no únicamente en función de la cantidad de piel visible resultaba impreciso, lo que llevó a explorar alternativas más avanzadas. Posteriormente, en 2008, [3] propuso un modelo basado en la técnica de bag-of-visual-words (BOVW), previamente utilizada en [4] para clasificar y categorizar objetos en imágenes, obteniendo resultados prometedores. Aunque los resultados fueron satisfactorios, se habría logrado una mayor precisión combinando las palabras visuales con características relacionadas con el color de la piel.

Con el avance de la tecnología, se propusieron soluciones basadas en aprendizaje profundo. Los autores en [5] propone una solución para la detección de pornografía infantil en imágenes. El enfoque se centra en el uso de redes neuronales convolucionales (CNN) y estrategias de aprendizaje por transferencia para identificar imágenes de explotación sexual infantil (SEIC). Los autores presentan un modelo de dos niveles. En el primer nivel, una CNN se entrena inicialmente con un conjunto amplio de datos de imágenes de objetos cotidianos (clasificación general de objetos) y luego se ajusta utilizando imágenes de contenido para adultos. En el segundo nivel, esta red pre-entrenada se somete a un ajuste adicional con un conjunto más pequeño de imágenes específicas de SEIC y no SEIC. Los resultados del estudio muestran que este enfoque de dos niveles supera significativamente a los métodos existentes, incluidos los basados en la detección de piel, las bolsas de palabras visuales y el detector de pornografía de Yahoo! de última generación. En [6] propone un nuevo marco de trabajo basado en aprendizaje profundo para la detección de material de Abuso Sexual Infantil (CSA). El enfoque principal es descomponer el problema de detección de CSA en dos tareas más simples: detección de contenido pornográfico y clasificación de grupos de edad (menor o adulto) utilizando redes neuronales convolucionales (CNN). Para esto, los autores presentan una nueva arquitectura de CNN, llamada AttM-CNN, que incorpora un novedoso mecanismo de atención visual para enfocarse en las regiones relevantes de la imagen y un aprendizaje métrico para minimizar la variación intraclase y maximizar la distancia interclase. Además, introducen dos nuevos conjuntos de datos a gran escala: Pornographic-2M para detección de pornografía y Juvenile-80k para clasificación de edad. Las conclusiones demuestran que el marco de trabajo propuesto, que combina las salidas de los modelos de detección de pornografía y clasificación de edad, supera a los métodos existentes, incluido el software forense NuDetective, logrando una precisión del 92,72 % en la clasificación binaria de CSA y la capacidad de encontrar el 80 % de las imágenes de CSA en el 8,5 % superior de una lista clasificada por la puntuación de relevancia de CSA. El estudio destaca la efectividad del aprendizaje profundo y la importancia de crear conjuntos de datos a gran escala para abordar la escasez de datos etiquetados en la detección de CSA, proporcionando una herramienta valiosa para las fuerzas del orden en la lucha contra el abuso sexual infantil.

Otros estudios han refinado la detección de contenido adulto general. En [7] los autores propusieron una red neuronal convolucional (CNN) para la detección de contenido adulto en imágenes. Para ello, emplearon una técnica conocida como ≪tiempo limitado Gaussiano-Bernoulli≫, que se utilizó para extraer las características más significativas de las imágenes. Posteriormente, estas características fueron comprimidas mediante una máquina de Boltzmann, que permitió al modelo alcanzar una precisión del 99,16 %. Este trabajo evolucionó en [8] presenta un modelo para el reconocimiento de imágenes con contenido para adultos, basado en redes neuronales convolucionales profundas (CNN) integradas en una máquina de Boltzmann de Bernoulli-Gaussiana de tiempo limitado. El enfoque combina las CNN para la extracción y clasificación de características con la capacidad de la máquina de Boltzmann para resumir eficazmente dichas características. Este diseño permite un proceso más eficiente de extracción de características y mejora el rendimiento en la detección de contenido explícito. Los resultados obtenidos demuestran que el modelo propuesto supera en eficiencia y precisión a otros enfoques de vanguardia. Además, el modelo aborda el creciente desafío del contenido para adultos en Internet, destacando el uso de técnicas de aprendizaje profundo como una herramienta clave para filtrar y gestionar este tipo de material, especialmente con el objetivo de proteger a los menores de edad.

Los autores en [9] presenta una revisión de las redes neuronales convolucionales (CNN) en la visión por computadora, centrándose en la clasificación de imágenes, la detección de objetos y la predicción de video. El trabajo analiza la evolución de las arquitecturas de CNN, desde AlexNet hasta modelos más recientes como ResNet y MobileNet, destacando las mejoras en precisión y eficiencia. Las conclusiones enfatizan la capacidad de las CNN para lograr un rendimiento de vanguardia en tareas de visión por computadora y discuten los desafíos que aún enfrentan, como la interpretabilidad, los requisitos de datos y recursos computacionales, y la necesidad de una base teórica más sólida.

La literatura también explora técnicas de clasificación adaptables a diversos tipos de datos, con aplicaciones directas en la visión artificial. En [10], se presenta un sistema basado en MobileNet para detectar enfermedades en cultivos de manzano, priorizando la estabilidad y el bajo costo computacional. Los resultados demuestran que, aunque MobileNet, InceptionV3 y ResNet152 presentan precisiones similares, MobileNet destaca por su eficiencia operativa (0,22 s/imagen). Los autores concluyen que esta arquitectura es óptima para aplicaciones móviles además se proponen, para futuras investigaciones, el desarrollo de métricas de gradación de enfermedades basadas en conjuntos de datos más robustos.

En [11], se investiga la evaluación estética computacional

aplicada al diseño de arte ambiental, clasificando imágenes en categorías de estética alta así como baja. Los autores proponen el Análisis Discriminante Jerárquico de Computación de Borde Móvil (MECHDA) en conjunto con un método de evaluación basado en Redes Neuronales Convolucionales Profundas (DCNN) en paralelo. Los resultados experimentales indican que el algoritmo MECHDA optimiza la eficiencia en la transmisión de datos, mientras que la arquitectura DCNN paralela logra una mayor precisión en la clasificación. Este enfoque integral demuestra un rendimiento superior en la automatización del análisis estético, facilitando la categorización de piezas de diseño ambiental.

Existen diferentes trabajos relacionados con la implementación de de modelos de Machine Learning en dispositivos móviles en los que incluyen la optimización de recursos, seguridad, y en aplicaciones específicas. En [12] mencionan que, para mejorar la eficiencia móvil, los autores desarrollaron un nuevo método de poda de filtros basado en la inferencia de varianza para reducir el tiempo de ejecución en los dispositivos móviles sin comprometer la capacidad del modelo. Para proteger la privacidad del usuario, se diseñó un mecanismo de dos fases. La primera fase implica modificar la función de pérdida para que la salida intermedia de los dispositivos móviles solo contenga información para tareas normales, mientras que la información sensible se elimina. La segunda fase emplea un entrenamiento contradictorio para evitar que los atacantes reconstruyan los datos originales a partir de la salida intermedia. Los resultados mostraron que el sistema puede lograr una aceleración significativa en los dispositivos móviles sin comprometer la precisión del modelo ni la privacidad del usuario.

Los autores [13] presentan un algoritmo de entrenamiento en tiempo real para dispositivos móviles, enfocado en aplicaciones de reconocimiento facial. El objetivo principal es demostrar la viabilidad del aprendizaje en tiempo real de un gran volumen de muestras dentro de estos dispositivos. Para ello, el método propuesto emplea la extracción de características locales, una red neuronal secuencial, proyección aleatoria local (LRP) y un clasificador incremental aleatorio local (LRIC). Los resultados muestran que el algoritmo es capaz de entrenar nuevas muestras de manera incremental en tiempo real, sin problemas de memoria ni degradación de la precisión, utilizando un modelo reciente de teléfono móvil. Además, el enfoque busca equilibrar la eficiencia computacional y la precisión, optimizando su desempeño en dispositivos con recursos limitados. En [14] se presenta un sistema móvil ligero basado en encuestas para la identificación de alimentos. Este sistema está diseñado para integrarse completamente en un dispositivo móvil y es capaz de identificar y clasificar comidas que contienen múltiples tipos de alimentos en un mismo plato. La metodología emplea diversas características de textura y color, las cuales son procesadas mediante múltiples clasificadores. Los resultados obtenidos demuestran la efectividad del sistema en la identificación de alimentos, logrando un equilibrio entre costo computacional y rendimiento. Además, gracias al uso de un búfer de características, se consigue una reducción del 90 % en el tiempo de clasificación, manteniendo un desempeño óptimo en distintas condiciones de iluminación. [15] Se propone un sistema completo y totalmente automatizado de verificación facial en dispositivos móviles. Este sistema está compuesto por tres subsistemas principales: detección facial, alineación y verificación. Su diseño tiene como objetivo ofrecer un método natural y no intrusivo para garantizar la seguridad de los dispositivos móviles. Para optimizar su rendimiento, se analizan diversos algoritmos de reconocimiento facial basados en técnicas de subespacio, integrando en el sistema aquel que resulte más adecuado para su implementación en dispositivos móviles.

La revisión de la literatura evidencia una clara evolución de las técnicas tradicionales de análisis de piel y textura hacia arquitecturas complejas de aprendizaje profundo que maximizan la precisión en la detección de contenidos inapropiados. Dado que los delitos relacionados con el intercambio de estas imágenes representan un riesgo físico y psicológico para menores y adultos, se justifica la necesidad de herramientas de filtrado eficaces. Sin embargo, el despliegue de estos modelos en entornos móviles exige un equilibrio crítico entre la eficiencia operativa, la seguridad de los datos sensibles procesados y la protección de la privacidad del usuario. En este sentido, persiste la necesidad de equilibrar la alta precisión de los modelos con la eficiencia computacional, asegurando procesos no intrusivos que garanticen la seguridad del usuario sin degradar el rendimiento del hardware.

III. Detección de imágenes con MobileNet

El análisis de imágenes con contenido sexual almacenadas en el dispositivo se realizó con el modelo de MobileNet verificando su correcto funcionamiento para ser implementado en dispositivos móviles con la versión móvil del modelo. El proceso se realiza cuando se tiene una imagen y esta se envía al modelo para su análisis, el modelo realizara la predicción y si tiene contenido sexual la clasificara como imagen con contenido explicito.

El modelo se entrenó con un Dataset de contenido sexual obtenido de una fuente publica con 6.89GB, con un total de 33.096 de imágenes de contenido sexual, de las cuales 24.998 son imágenes de entrenamiento (12.499 imágenes de cada clase), 4.099 imágenes de validación (2.049 imágenes de la clase normal y 2.050 imágenes sexuales) y, por último, 3.999 imágenes de prueba (2.002 imágenes de la clase normal y 1.997 imágenes de la clase explícita).

Para la importación de las imágenes se utilizó la función flow_from_directory de la clase ImageDataGenerator. Esta clase está dentro del módulo TensorFlow Keras Preprocessing Image y nos permite cargar lotes de imágenes en un generador de salida. La función flow_from_directory lo que hace es: tomar la dirección de una única ruta, recorrer cada una de las carpetas dentro de esa ruta y, finalmente, guardar las imágenes en el generador de salida.

Seleccionar el modelo de MobileNet más adecuado para el reconocimiento de imágenes dentro de un dispositivo móvil se realizó el análisis de las versiones de MobileNetV1, MobileNetV2, MobileNetV3-Small y MobileNetV3-Large, para elegir cual es el más adecuado para el reconocimiento de imágenes sexuales. Se utilizo dos clases para su entrenamiento ”normalz ”porn”. Este análisis comparativo muestra el desempeño del modelo y muestra diferencias significativas en su desempeño al aplicar técnicas de Aumento de Datos (DA).

MobileNetV1, (las métricas se pueden observar en la Figura 1), muestra un bajo rendimiento, alcanzando solo un 61.10 % de precisión en entrenamiento y un 62.89 % en validación. Además, presenta 984 falsos negativos y 458 falsos positivos, fallando en aproximadamente un tercio de las imágenes de prueba. El modelo parte con una precisión muy baja en el conjunto de entrenamiento como en el de validación y mejora únicamente un 13 %, y parte con una perdida alta y no consigue reducirla en más de un 13 % por lo que su pérdida inicial es alta lo que indica problemas para generalizar y poca utilidad para implementar DA, por lo que esto hará que no pueda generalizar correctamente. Por otro lado, **MobileNetV2** (métricas del modelo en la Figura 2), mejora estos resultados, logrando un 83.91 % de precisión en entrenamiento y un 74.14 % en validación. Sin embargo, con 979 falsos negativos en un conjunto de prueba de 1,997 imágenes explícitas, evidencia dificultades para detectar contenido explícito. Las métricas de entrenamiento siguen siendo superiores a la de validación, lo que indica problemas de generalización similares a los de MobileNetV1.

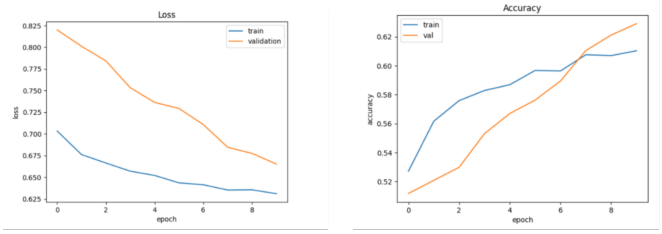


Fig. 1. Métricas de MobileNetV1.

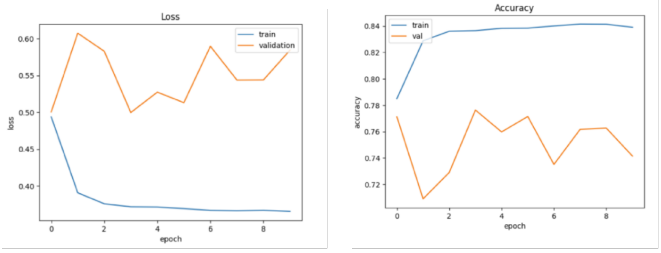


Fig. 2. Métricas de MobileNetV2.

A diferencia de **MobileNetV3-Small** (métricas del modelo en la Figura 3), destaca por su alto rendimiento y capacidad de generalización. Con una pérdida de entrenamiento del 15.28 % y de validación del 14.58 %, logra una precisión del 94.11 % en entrenamiento y del 94.44 % en validación. Este modelo solo clasifica incorrectamente 126 imágenes explícitas y 93 normales, demostrando, es decir, que consigue clasificar 3.779 imágenes de un total de 3.999 imágenes. Sus funciones de pérdida y precisión son consistentes y muestran una evolución de las funciones de coste y precisión que se dirigen hacia el punto mínimo (en el caso de la función de pérdida) y el punto máximo (en el caso de la función de precisión) y que, en todo momento, el conjunto de datos de validación consigue mejores métricas que el conjunto de datos de prueba. Con **MobileNetV3-Large** (métricas del modelo en la Figura 4), aunque también presenta buenos resultados (ya que se trata de una extensión del modelo MobileNetV3-Small), con un 92.64 % de precisión en entrenamiento y un 91.71 % en validación, junto con pérdidas del 18.68 % de entrenamiento, un 19,53 % de pérdida de validación, tiene un desempeño inferior al de MobileNetV3-Small. Además, presenta 76 falsos negativos y 239 falsos positivos, aunque las métricas sean buenas MobileNetV3-Small Tiene 3 millones de parámetros menos que MobileNetV3-Large. Esto significa que MobileNetV3-Small será más fácil implementar en un móvil, Table I.

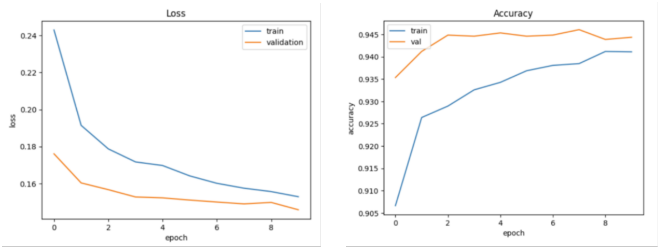


Fig. 3. Métricas MobileNetV3 small.

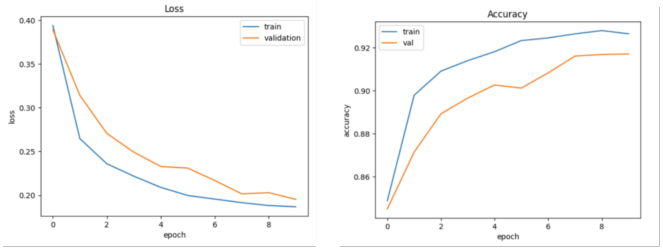


Fig. 4. Métricas MobileNetV3 Large.

En conclusión, MobileNetV3-Small es la opción más adecuada para trabajar con DA debido a su alta precisión, menor pérdida, excelente capacidad de generalización y eficiencia en términos de parámetros y rendimiento, especialmente en aplicaciones móviles donde la optimización de recursos es crucial.

III-A. Interpretabilidad MobileNet

Para comprender las decisiones internas del modelo MobileNetV3-Small, se utilizó la técnica de interpretabilidad Grad-CAM (Gradient-weighted Class Activation Mapping). La siguiente Figura 5 muestran los mapas de calor obtenidos, los cuales indican las regiones de la imagen en las que el modelo enfoca su atención para clasificar las clases como ”normal.º ”porn”. En imagen explícita presentada (Figura 5), el modelo destaca prominentemente la región superior del torso como el área de mayor relevancia con contenido sexual, mostrando la totalidad de la zona explícita de la imagen, indicando que el modelo considera todo el contenido visible en dicha área como determinante para la clasificación.

Para la clase ”normal”, los mapas de calor obtenidos mediante Grad-CAM también ofrecen información crucial sobre la explicabilidad del modelo en la confirmación de la ausencia de contenido explícito. En la Figura 6, el mapa de calor resalta las regiones de la imagen que, según el modelo, confirman la ausencia de contenido sexual explícito.

Tabla I
RENDIMIENTO DE LAS ARQUITECTURAS MOBILENET CON TÉCNICAS DA.

Modelo	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss	False Negatives	False Positives	True Positives	True Negatives	Observaciones
MobileNetV1	61.10 %	62.89 %	63.10 %	66.52 %	984	458	1015	1541	Bajo rendimiento, problemas de generalización, inadecuado para DA.
MobileNetV2	83.91 %	74.14 %	36.57 %	58.42 %	979	46	1020	1953	Mejor que MobileNetV1, pero con dificultades para generalizar y detectar contenido explícito.
MobileNetV3-Small	94.11 %	94.44 %	15.28 %	14.58 %	126	93	1873	1906	Excelente precisión y capacidad de generalización, ideal para DA.
MobileNetV3-Large	92.64 %	91.71 %	18.68 %	19.53 %	76	239	1923	1763	Buen rendimiento, pero menos eficiente que MobileNetV3-Small debido a su mayor número de parámetros.

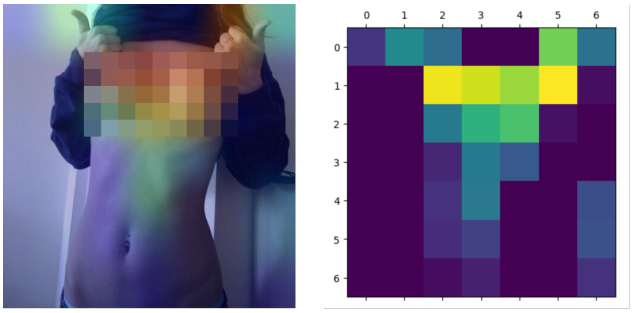


Fig. 5. Mapa de calor para la clase de contenido sexual.

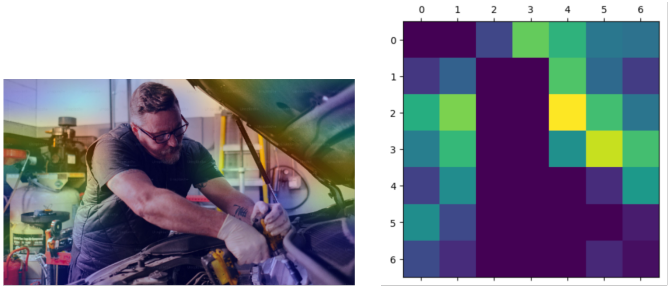


Fig. 6. Mapa de calor de la clase normal.

Es importante destacar que, para la clasificación ”normal”, el modelo se enfoca en los conjuntos de píxeles relevantes para confirmar esta categoría. Los mapas de calor generados se concentran en las zonas donde no existen elementos explícitos.

Esta evidencia permite concluir que el modelo realiza correctamente tanto la detección como la clasificación de las imágenes en ambas categorías (”porn ”normal”), ya que no solo identifica el contenido sexual, sino que también valida la ausencia de este.

IV. RENDIMIENTO DEL MODELO EN EL DISPOSITIVO MÓVIL

Una vez realizado el entrenamiento del modelo, se realizó la creación del modelo en versión móvil con extensión ”.tflite”. En el que se utilizó en un dispositivo móvil con Sistema Operativo Android versión 10, API 29.

Para asegurar la privacidad y seguridad de los datos analizados del dispositivo se realizó un análisis de las principales arquitecturas de implementación de modelos en los móviles [16]. Para garantizar la seguridad de los datos, se utilizó una arquitectura en la que tanto el entrenamiento como la prueba se realizan directamente en el dispositivo. Esta estrategia ofrece importantes ventajas, como un aumento en la privacidad, ya que los datos nunca salen del dispositivo. Además, no es necesario recurrir a una API externa para realizar el análisis ni para comunicarse con el modelo de aprendizaje automático. Al realizar tanto el entrenamiento como las inferencias de manera local, se preserva la privacidad del usuario y se permite que el modelo siga aprendiendo de los datos almacenados directamente en el dispositivo móvil.



Figura 7. Rendimiento de la CPU del modelo MobileNetV3-Small en dispositivos móviles.

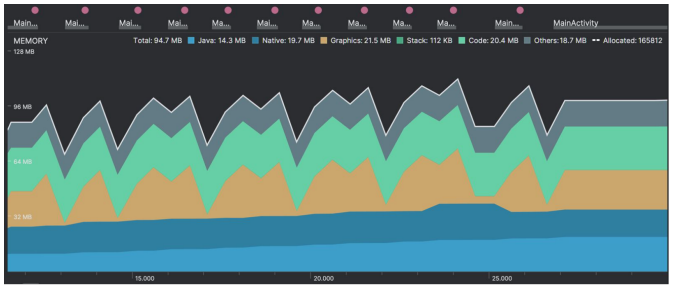


Figura 8. Rendimiento en memoria del modelo MobileNetV3-Small en dispositivos móviles.

Una vez definida la arquitectura, el modelo fue implementado en el dispositivo móvil. Se llevaron a cabo pruebas exhaustivas para analizar su rendimiento en el procesamiento de imágenes. Durante estas pruebas, la evaluación se centró en métricas clave de desempeño: el uso de la CPU y el consumo de memoria del dispositivo. Analizar estos aspectos

es fundamental para garantizar que el modelo opere de manera óptima. Esto asegura que el dispositivo no experimente problemas de rendimiento, indisponibilidad o un consumo excesivo de batería, permitiendo una integración eficiente en el smartphone. El rendimiento del modelo de MobileNetV3-Small al analizar una imagen o video la CPU alcanzaba un 12 % de uso, Figura 7, la memoria alcanza un pico máximo de 94.7MB, Figura 8. Hay que destacar que el funcionamiento del modelo es evaluado en un dispositivo con funcionamiento normal, refiriéndose a que está en uso con todos los servicios móviles disponibles y aplicaciones restantes en funcionamiento.

V. Conclusiones

La inteligencia artificial constituye una innovación significativa en la tecnología de la información, revolucionando la forma en que los usuarios se relacionan con los dispositivos inteligentes. No obstante, uno de los retos más significativos al trabajar con inteligencia artificial es asegurar la salvaguarda de la información personal, particularmente en el sector de la tecnología móvil. Este tema adquiere mayor relevancia considerando que muchos usuarios, utilizan móviles en el día a día manteniendo comunicación con su círculo cercano de conocidos de amigos o familiares en diferentes plataformas de comunicación como en el uso de mensajería instantánea. Por lo cual, para facilitar el análisis de la información compartida o almacenada en dispositivos móviles, específicamente en imágenes, se implementó el modelo de machine learning MobileNetV3-Small. El análisis se centró en clasificar el contenido en dos categorías: sexual y normal (sin contenido sexual). Previamente, se realizó un preprocesamiento en las imágenes, recortándolas a dimensiones de 224x224.

El modelo demostró una alta precisión y logra clasificar correctamente las imágenes. MobileNetV3-Small fue identificada como el modelo más adecuado para su implementación en dispositivos móviles, logrando un accuracy del 94.11 %. Una vez completado el entrenamiento, se llevó a cabo un análisis de interpretabilidad para identificar las áreas de las imágenes en las que el modelo se basa para realizar la clasificación. Este análisis confirmó que MobileNetV3-Small destaca adecuadamente las regiones consideradas sexuales, lo que respalda su capacidad para clasificar de manera precisa y confiable. Se utilizo la arquitectura ”Training and Testing on the Devices”para implementar el modelo en el móvil garantizando y priorizando la seguridad de los datos del usuario y con ello evitar ataques informáticos.

Los resultados obtenidos demuestran que el rendimiento del modelo al clasificar imágenes en el dispositivo fue satisfactorio. El consumo de CPU y memoria fue bajo, lo que permitió un funcionamiento fluido del dispositivo sin problemas de sobrecalentamiento, consumo excesivo de batería o interrupciones en el servicio. Esto contribuye a garantizar la seguridad y protección de los datos de los usuarios, así como de la información que se comparte o almacena en el dispositivo, reforzando la confianza en la implementación local de modelos de Machine Learning.

Agradecimientos

Este trabajo se ha realizado en el marco de los fondos del Plan de Recuperación, Transformacionales y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra “Ciberseguridad para la Innovación y la Protección y la Protección Digital” (Cátedra INCIBE-UCM). Asimismo, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, Proyecto CIRMA-CM (TEC-2024/COM-404).

Referencias

[1] A. E. D. Zambrano, A. N. S. Cedeño, M. C. F. López, M. J. E. Cedeño, G. A. S. Sardi *et al.*, “Fundamentación teórica de la inteligencia artificial en el desarrollo de aplicaciones móviles en el instituto de admisión y nivelación de la universidad técnica de manabí,” *Tesla Revista Científica*, vol. 3, no. 2, pp. e223–e223, 2023.

[2] M. M. Fleck, D. A. Forsyth, and C. Bregler, “Finding naked people,” in *Computer Vision—ECCV’96: 4th European Conference on Computer Vision Cambridge, UK, April 15–18, 1996 Proceedings Volume II 4*. Springer, 1996, pp. 593–602.

[3] T. Deselaers, L. Pimenidis, and H. Ney, “Bag-of-visual-words models for adult image classification and filtering,” in *2008 19th International Conference on Pattern Recognition*. IEEE, 2008, pp. 1–4.

[4] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual categorization with bags of keypoints,” in *Workshop on statistical learning in computer vision, ECCV*, vol. 1, no. 1-22. Prague, 2004, pp. 1–2.

[5] P. Vitorino, S. Avila, M. Perez, and A. Rocha, “Leveraging deep neural networks to fight child pornography in the age of social media,” *Journal of Visual Communication and Image Representation*, vol. 50, pp. 303–313, 2018.

[6] A. Gangwar, V. González-Castro, E. Alegre, and E. Fidalgo, “Attm-cnn: Attention and metric learning based cnn for pornography, age and child sexual abuse (csa) detection in images,” *Neurocomputing*, vol. 445, pp. 81–104, 2021.

[7] S. Karamizadeh, S. Shojae Chaeikar, and A. Jolfaei, “Adult content image recognition by boltzmann machine limited and deep learning,” *Evolutionary Intelligence*, pp. 1–10, 2022.

[8] ——, “Adult content image recognition by boltzmann machine limited and deep learning,” *Evolutionary Intelligence*, vol. 16, no. 4, pp. 1185–1194, 2023.

[9] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, “A review of convolutional neural networks in computer vision,” *Artificial Intelligence Review*, vol. 57, no. 4, p. 99, 2024.

[10] C. Bi, J. Wang, Y. Duan, B. Fu, J.-R. Kang, and Y. Shi, “Mobilenet based apple leaf diseases identification,” *Mobile Networks and Applications*, pp. 1–9, 2022.

[11] Y. Yang and Y. C. Ko, “Application of mobile edge computing combined with convolutional neural network deep learning in image analysis,” *International Journal of System Assurance Engineering and Management*, vol. 13, no. Suppl 3, pp. 1186–1195, 2022.

[12] P. Zheng, Z. Su, Y. Xue, F. Zhou, and C. Peng, “An efficient and privacy-preserving framework for cloud-assisted mobile deep learning with filter pruning and adversarial mechanisms,” *Available at SSRN 5070623*, 2024.

[13] K. Choi, K.-A. Toh, and H. Byun, “Realtime training on mobile devices for face recognition applications,” *Pattern recognition*, vol. 44, no. 2, pp. 386–400, 2011.

[14] L. Oliveira, V. Costa, G. Neves, T. Oliveira, E. Jorge, and M. Lizarraga, “A Mobile, Lightweight, Poll-based Food Identification System,” *Pattern Recognition*, vol. 47, no. 5, pp. 1941–1952, 2014.

[15] J. Ren, X. Jiang, and J. Yuan, “A Complete and Fully Automated Face Verification System on Mobile Devices,” *Pattern Recognition*, vol. 46, no. 1, pp. 45–56, 2013.

[16] S. Pérez Arteaga, A. L. Sandoval Orozco, and L. J. García Villalba, “Analysis of machine learning techniques for information classification in mobile applications,” *Applied Sciences*, vol. 13, no. 9, p. 5438, 2023.

Viabilidad de un intercambio de clave ML-KEM autenticado en entornos IoT mediante LoRa

Rodolfo Leonardo Sumoza Matos [1,2], Esteban Eduardo Mocskos [1,3] y Raúl Durán Díaz [4]
[1]Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires (UBA), Argentina
{rsumoza, emocskos}@dc.uba.ar
[2]Departamento de Informática, Facultad de Ingeniería, Universidad Austral, Pilar, Buenos Aires, Argentina
[3]Centro de Simulación Computacional (CSC-CONICET), Buenos Aires, Argentina
[4]Escuela Politécnica Superior, Universidad de Alcalá (UAH), Alcalá de Henares, España
raul.duran@uah.es

***Resumen*—Se presenta una caracterización experimental detallada de un protocolo de *intercambio de claves poscuántico autenticado* para entornos IoT basados en tecnología LoRa. El esquema se fundamenta en el mecanismo de encapsulación de clave ML-KEM-512 (FIPS 203) y emplea autenticación simétrica con una clave precompartida (PSK) mediante un MAC ligero instanciado con Ascon-MAC, con el objetivo de mantener baja sobrecarga computacional y energética.**

El protocolo se implementó y ejecutó sobre dos pasarelas RAK7248 en modo *Gateway-to-Gateway* (G2G) lo que les permite actuar directamente como nodos *fog* activos. Se evaluó su desempeño en ese modo de comunicación, que es de baja tasa de transmisión y alta restricción energética y se midieron el consumo energético, el tiempo en aire y la tasa de error de paquetes (PER) del intercambio de claves bajo distintos factores de dispersión y en dos escenarios de propagación: interior *Non-Line-of-Sight* (NLoS, con obstrucciones entre nodos) y exterior *Line-of-Sight* (LoS, con visión directa).

Los resultados muestran que el costo computacional del KEM es bajo en hardware de nivel pasarela, mientras que el costo total del intercambio está dominado por la transmisión de fragmentos. Tanto el tiempo en aire como la energía consumida crecen exponencialmente con el factor de dispersión, y la PER se mantiene por debajo del 5 % en todos los casos. Estos hallazgos confirman la viabilidad de integrar mecanismos poscuánticos en sistemas IoT basados en LoRa, siempre que se combinen con autenticación simétrica ligera y estrategias eficientes de fragmentación y compresión.

***Palabras clave*—ML-KEM, autenticación ligera, criptografía poscuántica, LoRa/LoRaWAN, IoT, consumo energético.**

I. Introducción

El avance de la computación cuántica amenaza los fundamentos de la criptografía de clave pública clásica [1], haciendo necesario adoptar mecanismos poscuánticos basados en problemas matemáticos que sigan siendo inabordables incluso ante computadores cuánticos. Entre estos mecanismos, el esquema de encapsulación de clave *Module-Lattice-Based Key-Encapsulation Mechanism* (ML-KEM), recientemente estandarizado por el NIST como FIPS 203 [2], constituye el sucesor directo de CRYSTALS-Kyber [3] y uno de los pilares de la transición hacia la criptografía resistente a la computación cuántica.

La incorporación de ML-KEM en entornos de *Internet of Things* (IoT) plantea, sin embargo, desafíos prácticos significativos. Los dispositivos y pasarelas (*gateways*) de baja potencia, en particular aquellos que emplean tecnologías LPWAN como LoRa/LoRaWAN, operan con anchos de banda reducidos, restricciones de tiempo de servicio (*duty cycle*) y disponibilidad energética muy limitada. En este contexto, los tamaños de clave pública y criptograma de ML-KEM (del orden de 0.8 a 1.5 kB) exceden ampliamente la carga útil máxima típica de LoRa (200–230 B), lo que obliga a fragmentar los mensajes y eleva tanto el tiempo en aire como el consumo energético [4], [5]. La viabilidad real de la criptografía poscuántica en sistemas IoT debe, por tanto, evaluarse empíricamente, considerando no sólo la solidez criptográfica sino también el impacto físico sobre el canal.

En un trabajo previo [6] se propuso un protocolo de *intercambio de claves poscuántico autenticado* diseñado específicamente para LoRa/LoRaWAN, combinando un KEM tipo Kyber con un código de autenticación de mensajes (MAC) derivado de una clave precompartida (PSK), que permitía la autenticación al tiempo que evitaba el uso de firmas poscuánticas de gran tamaño. Permanecía, sin embargo, abierta la cuestión de su viabilidad práctica.

Desde el punto de vista teórico, ML-KEM se basa en el problema de *Module Learning with Errors* (MLWE), que extiende las construcciones sobre retículos. Proporciona seguridad en tres niveles: ML-KEM-512, ML-KEM-768 y ML-KEM-1024, que ofrecen aproximadamente 128, 192 y 256 bits de seguridad, respectivamente. Para entornos IoT, el nivel 512 representa un equilibrio adecuado entre seguridad y tamaño de los objetos criptográficos, permitiendo un análisis realista de fragmentación, tiempo en aire y consumo energético.

La autenticación simétrica se implementa mediante Ascon-MAC. Ascon es una familia de primitivas ligeras que, sin el componente MAC, ha sido recientemente estandarizada por el NIST en la serie SP 800 [7]. Combina eficiencia computacional y alta resistencia criptográfica, convirtiéndose así en una opción idónea para dispositivos IoT con recursos limitados. Ascon-MAC, sin estar aún estandarizado, puede, sin embargo, ser usado para generar códigos de autenticación de peso ligero, como explican sus diseñadores en [8].

La Tabla I muestra que en un intercambio de clave poscuántico ML-KEM, en que las partes terminan compartiendo una clave común K, tanto la clave pública pk como el criptograma ct superan el tamaño máximo de carga útil de LoRa, que oscila entre 51 B y 222 B según la tasa de datos, el factor de dispersión y la región de operación [9], [10], [11]. Aparece así la necesidad de fragmentar los datagramas lo que implica más tiempo en aire, más consumo de energía y mayor probabilidad de retransmisión.

Sobre esta base, el presente trabajo muestra experimentalmente la viabilidad de un intercambio de clave (KEM)

Tabla I
TAMAÑOS DEFINIDOS EN FIPS 203 PARA ML-KEM

Esquema	$\|pk\|$ (B)	$\|ct\|$ (B)	$\|K\|$ (B)
ML-KEM-512	800	768	32
ML-KEM-768	1184	1088	32
ML-KEM-1024	1568	1568	32

autenticado basado en ML-KEM-512 y Ascon-MAC sobre pasarelas RAK7248 conectadas por LoRa en modo *Gateway-to-Gateway* (G2G) lo que les permite actuar como nodos *activos*, evitando así explícitamente la sobrecarga impuesta por el protocolo LoRaWAN y la necesidad de conexiones hacia servidores de red.

II. METODOLOGÍA

Para lograr este objetivo, se implementó un escenario Pasarela-a-Pasarela o *Gateway-to-Gateway* (G2G) en el que todos los objetos criptográficos se transmiten vía LoRa de manera fragmentada. El entorno reproduce un caso práctico en el que dos pasarelas IoT establecen una clave efímera segura sobre un canal de baja tasa binaria, sin infraestructura de clave pública (PKI) y con autenticación simétrica ligera mediante Ascon-MAC en donde, siguiendo el patrón propuesto en [6], los nodos disponen de una clave precompartida. Por razones de espacio, no podemos tratar en este trabajo la gestión administrativa de dicha clave (provisionado, rotación, etc.).

II-A. Escenario experimental y rol de los nodos

Los experimentos se realizaron entre dos pasarelas modelo RAK7248 que ejecutan Linux ARM64 y la implementación de referencia de ML-KEM-512, compilada con `-O3` para una arquitectura ARMv8-A. El intercambio involucra las tres operaciones del KEM (`KeyGen`, `Encaps`, `Decaps`) y las dos transmisiones LoRa fragmentadas.

Una pasarela, identificada como Nodo A, inicia el protocolo enviando un mensaje M_1 al Nodo B. Este verifica el mensaje y, si es válido, ejecuta la función de encapsulación, enviando un mensaje de vuelta, M_2, al Nodo A. Este lo verifica y, si es válido, acepta la clave intercambiada.

Los tamaños de M_1 y M_2 (aprox. 800–900 B) obligan a su transmisión en f_1 y f_2 fragmentos LoRa con CRC físico, incluso en condiciones ideales. En ambos sentidos, la fragmentación de M_1 y M_2 se realiza a nivel de aplicación asignando a cada trama LoRa un *número de secuencia* y el *número total de fragmentos* del mensaje. El receptor reordena los fragmentos recibidos, descarta duplicados y considera fallido el intercambio si detecta huecos en la numeración tras expirar un temporizador. Este mecanismo implementa explícitamente el "tratamiento cuidadoso de numeración de fragmentos" al que se hace referencia en la Sección IV.

II-B. Flujo del intercambio KEM autenticado

El flujo completo del intercambio, sobre el cual se registran todas las mediciones temporales y energéticas, se resume a continuación. Cada paso corresponde a una operación efectivamente ejecutada en los nodos RAK7248.

La naturaleza del protocolo de intercambio de claves garantiza que, al finalizar, el Nodo A está compartiendo la misma clave K con el Nodo B; además, si todas las verificaciones resultaron exitosas, quedan aseguradas la integridad y autenticación extremo a extremo sin que se haga necesario hacer uso de certificados digitales.

1. **Nodo A: generación de claves y autenticación inicial**
$$(pk, sk) \leftarrow \mathsf{ML\text{-}KEM.KeyGen}(),$$
$$T_1 \leftarrow \mathsf{Ascon\text{-}MAC}_{\mathsf{PSK}}(pk),$$
$$M_1 = pk \,\|\, T_1.$$
2. **Nodo A → Nodo B: transmisión de M_1**
Este se fragmenta en f_1 tramas LoRa:
$$M_1[1], \ldots, M_1[f_1] \;\leftarrow\; M_1.$$
3. **Nodo B: verificación y encapsulación**
El Nodo B reensambla M_1 y verifica el MAC:
$$pk \,\|\, T_1 \leftarrow M_1[1], \ldots, M_1[f_1],$$
$$b \leftarrow \mathsf{Ascon\text{-}VERIF}_{\mathsf{PSK}}(pk, T_1).$$
Si $b = 1$, es decir, la verificación es correcta:
$$(ct, K) \leftarrow \mathsf{ML\text{-}KEM.Encaps}(pk),$$
$$T_2 \leftarrow \mathsf{Ascon\text{-}MAC}_{\mathsf{PSK}}(pk \,\|\, ct),$$
$$M_2 = ct \,\|\, T_2.$$
4. **Nodo B → Nodo A: transmisión de M_2**
Este se fragmenta en f_2 tramas LoRa:
$$M_2[1], \ldots, M_2[f_2] \;\leftarrow\; M_2.$$
5. **Nodo A: verificación y desencapsulación**
El Nodo A reensambla M_2 y verifica el MAC:
$$ct \,\|\, T_2 \leftarrow M_2[1], \ldots, M_2[f_2],$$
$$b \leftarrow \mathsf{Ascon\text{-}VERIF}_{\mathsf{PSK}}(pk \,\|\, ct, T_2).$$
Si $b = 1$, se ejecuta la desencapsulación:
$$K \leftarrow \mathsf{ML\text{-}KEM.Decaps}(ct, sk).$$

II-C. Hardware y configuración de LoRa

Cada RAK7248 integra un Raspberry Pi Compute Module 4 (ARM Cortex-A72, 2 GiB RAM) y un concentrador Semtech SX1302. El consumo energético se midió mediante un INA226 con muestreo a 1 kHz.

Parámetros del enlace LoRa:

- Frecuencia central: 915 MHz.
- Ancho de banda: 125 kHz.
- Codificación: 4/5.
- Potencia de transmisión: 14 dBm.
- Factores de dispersión: SF7, SF9 y SF12.

Escenarios de propagación:

1. **NLoS interior:** 10 m con dos paredes de hormigón.
2. **LoS exterior:** 100 m a cielo abierto.

II-D. Métricas y procedimiento experimental

Para caracterizar el intercambio de clave se definió como *intercambio completo* al intervalo que va desde el inicio de la llamada a `KeyGen` en el Nodo A hasta la verificación exitosa de la clave compartida en el mismo nodo. Sobre este intervalo se midieron tres métricas principales:

1. **Tiempo en aire (ToA):** suma del tiempo en aire de todos los fragmentos de M_1 y M_2 (incluyendo posibles retransmisiones).

2. **Energía consumida:** energía eléctrica total consumida por el Nodo A durante el intercambio completo, obtenida por integración de la potencia instantánea.
3. **PER:** tasa de error de paquetes (*Packet Error Rate*), definida como la proporción de fragmentos perdidos o corruptos respecto del total de fragmentos transmitidos.

Cada configuración *(SF, entorno)* se repitió 30 veces. Sobre las 30 realizaciones se eliminaron valores atípicos mayores a 1,5×IQR (rango intercuartílico) y se calculó la mediana y los intervalos correspondientes para las magnitudes representadas en las figuras.

En cuanto a la obtención de las métricas:

- El **tiempo en aire** se obtuvo *mediante cálculo analítico*, usando las fórmulas oficiales de Semtech a partir de los parámetros físicos de LoRa y del tamaño de cada fragmento.
- La **energía** se obtuvo mediante *medición directa* sobre la línea de alimentación del Nodo A, usando un shunt de baja resistencia y un sensor INA226.
- La **PER** se calculó a partir de los registros de aplicación, que anotan cuántos fragmentos se intentan enviar y cuántos se reciben correctamente (tras reensamblar y verificar numeración y CRC).

Para reducir la variabilidad y facilitar la reproducibilidad, todos los experimentos se automatizaron mediante *scripts* que implementan la siguiente secuencia:

1. Colocar el Nodo A en estado inactivo (sin tráfico LoRa).
2. Configurar el INA226 para muestrear tensión y corriente a 1 kHz y comenzar el registro de $V_{\text{shunt}}(t)$ y $V_{\text{bus}}(t)$.
3. Disparar el intercambio KEM autenticado (generación de claves, transmisión de M_1, recepción de M_2 y verificación final de la clave).
4. Detener el registro del INA226 cuando finaliza el intercambio.
5. Usar las marcas de tiempo de los registros de aplicación para identificar el intervalo exacto $[t_0, t_{\text{fin}}]$ en el que el intercambio estuvo activo, y recortar la traza eléctrica a esa ventana antes de integrar.

Por claridad y reproducibilidad, el código fuente del protocolo, así como los *scripts* de automatización y procesamiento de datos, están disponibles en un repositorio público de GitHub (https://github.com/rsumoza/mlkem-lora-recsi2026).

Obtención del tiempo en aire: El tiempo en aire (T_{air}) de cada fragmento no se midió mediante instrumentación externa, sino que se *calculó* a partir de los parámetros físicos de la modulación LoRa y del tamaño de la carga útil del fragmento (PL), usando las expresiones recomendadas por Semtech [12]. Los valores representados en las Figuras 2 y 3 corresponden exactamente a estos tiempos calculados.

Los parámetros físicos empleados se resumen en la Tabla II. Las expresiones empleadas son:

$$T_{\text{sym}} = \frac{2^{\text{SF}}}{\text{BW}}, \qquad T_{\text{preamble}} = (N_{\text{pre}} + 4.25)T_{\text{sym}}.$$

$$n_{\text{int}} = \left\lceil \frac{8\,\text{PL} - 4\,\text{SF} + 28 + 16\,\text{CRC} - 20\,H}{4(\text{SF} - 2\text{DE})} \right\rceil,$$

$$N_{\text{payload}} = 8 + \max\big(n_{\text{int}}(\text{CR} + 4), 0\big),$$

$$T_{\text{payload}} = N_{\text{payload}} T_{\text{sym}}, \qquad T_{\text{air}} = T_{\text{preamble}} + T_{\text{payload}}.$$

Tabla II
PARÁMETROS FÍSICOS USADOS EN EL CÁLCULO DEL TIEMPO EN AIRE

Parámetro	**Valor**
BW	125 kHz
N_{pre}	8 símbolos
H	0 (cabecera explícita)
CRC	1
CR	1 (codificación 4/5)
DE	1 para SF $\geq$ 11
PL	tamaño del fragmento (en bytes)

Dado que las longitudes de los fragmentos PL y los parámetros físicos (BW, SF, CR, etc.) son conocidos para cada configuración, T_{air} se obtiene determinísticamente. El *tiempo en aire total del intercambio* es la suma del T_{air} de todos los fragmentos de M_1 y M_2 presentes en esa ejecución (incluyendo retransmisiones si las hay). En un subconjunto de pruebas se comprobó que estos tiempos calculados coinciden con los intervalos observados a nivel de aplicación con diferencias despreciables a la escala de las figuras.

Obtención de la energía: A diferencia del tiempo en aire, el consumo energético se obtuvo mediante *medición directa* sobre la línea de alimentación del Nodo A. Para ello se utilizó un shunt de baja resistencia (1 mΩ–100 mΩ) en serie con la fuente, junto con un sensor INA226 muestreado a 1 kHz. Toda la corriente que alimenta el dispositivo fluye a través del shunt, generando una caída de tensión proporcional (típicamente del orden de milivoltios) que puede medirse con precisión por el INA226. Con este montaje, el INA226 obtiene:

- la caída de tensión en el shunt, $V_{\text{shunt}}(t)$, y
- la tensión del bus de alimentación, $V_{\text{bus}}(t)$.

En este contexto, $V_{\text{bus}}(t)$ representa la tensión de entrada del RAK7248 (aprox. 5 V). A partir de esas mediciones se derivan los valores

$$I(t) = \frac{V_{\text{shunt}}(t)}{R_{\text{shunt}}}, \qquad P(t) = V_{\text{bus}}(t)\, I(t),$$

donde $I(t)$ es la corriente instantánea y $P(t)$ la potencia consumida en cada instante. La energía total utilizada por un intercambio se obtiene integrando la potencia sobre la ventana temporal en que el protocolo estuvo activo:

$$E = \sum_i P(t_i)\,\Delta t, \qquad \Delta t = 1\,\text{ms}.$$

El procedimiento experimental para cada ejecución fue:

1. Iniciar el registro de $V_{\text{shunt}}(t)$ y $V_{\text{bus}}(t)$.
2. Ejecutar el intercambio completo (`KeyGen`, transmisión de M_1, recepción de M_2 y verificación).
3. Detener el registro.
4. Identificar la ventana $[t_0, t_{\text{fin}}]$ correspondiente a la actividad real del protocolo mediante marcas de tiempo.
5. Integrar $P(t)$ únicamente sobre esa ventana.

Para aislar el *consumo del cómputo criptográfico* (Fig. 7), se repite el procedimiento con la radio desactivada y se recorta la traza usando las marcas de tiempo asociadas a `KeyGen`, `Encaps` y `Decaps`.

La Figura 1 resume el montaje experimental y la ventana temporal usada para la integración. Los valores obtenidos por este método son los que se muestran en las Figuras 4 y 5, y sirven como base para la correlación mostrada en la Figura 6.

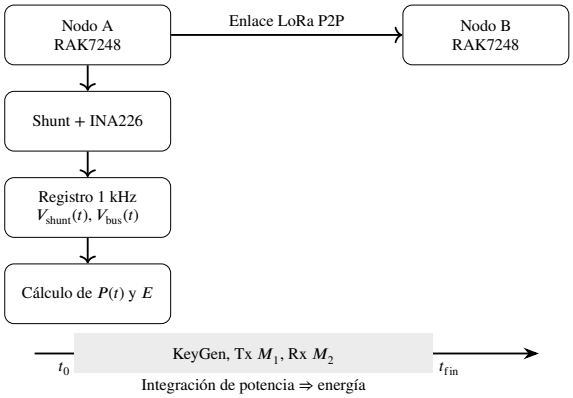


Figura 1. Procedimiento experimental para medir tiempo en aire y energía.

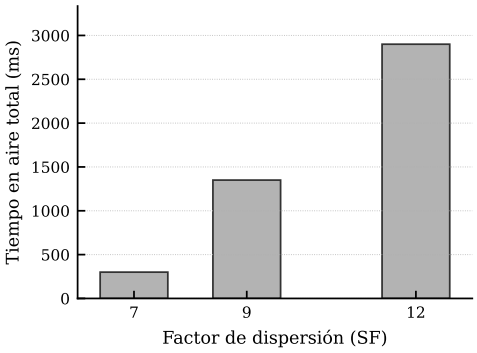


Figura 2. Tiempo en aire total por intercambio ML-KEM-512 en entorno NLoS.

III. Resultados experimentales

Los resultados permiten caracterizar cuantitativamente el comportamiento temporal y energético del intercambio autenticado basado en ML-KEM-512 sobre enlaces LoRa punto a punto.

III-A. Tiempo en aire

Las Figuras 2 y 3 muestran el *tiempo total en aire medido y calculado* por intercambio en escenarios NLoS y LoS. En ambos casos, el tiempo en aire crece exponencialmente con el SF, debido al aumento del *tiempo de símbolo* y a la fragmentación requerida para M_1 y M_2. Al pasar de SF7 a SF12, la duración promedio aumenta en más de un orden de magnitud, confirmando que la transmisión de fragmentos es el principal componente del retardo. Tanto el caso medido como el calculado, dieron los mismo resultados. Las figuras muestran los valores medidos, que se obtienen al medir el tiempo desde que llega el primer bit del paquete, hasta que llega el último bit, incluyendo el preámbulo, el encabezado, los bits útiles (payload), y los usados para CRC.

III-B. Energía consumida

Las Figuras 4 y 5 presentan la *energía total* consumida por intercambio, obtenida al integrar la potencia medida por el INA226. La tendencia replica la del tiempo en aire: el consumo aumenta con el SF y está fuertemente correlacionado con la duración de la actividad de radio. El entorno NLoS muestra valores levemente superiores, atribuibles a variaciones del canal y posibles retransmisiones.

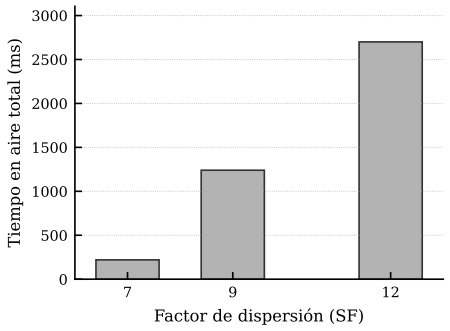


Figura 3. Tiempo en aire total por intercambio ML-KEM-512 en entorno LoS.

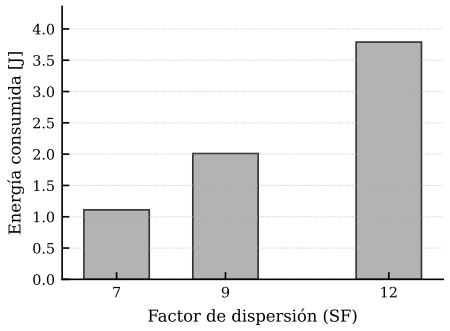


Figura 4. Energía total por intercambio ML-KEM-512 en entorno NLoS.

Para contextualizar, una batería 18650 de 3.6 V y 2400 mAh almacena ≈31 000 J, y una CR2032 alrededor de 2160 J, las cuales tienen diferentes costos y tamaños para adecuarse a distintos tipos de soluciones. Dado que cada intercambio completo requiere sólo unos pocos julios, un nodo IoT podría ejecutar entre 10^3 y 10^4 intercambios PQC antes de agotar su energía (bajo supuestos ideales), mostrando la compatibilidad energética del esquema.

III-C. Correlación tiempo-energía

La Figura 6 muestra la correlación entre el tiempo en aire y la energía por intercambio. La relación es prácticamente lineal en ambos entornos, lo que indica una potencia media casi constante y refuerza que el tiempo de radio efectivamente usado es el factor dominante del consumo total.

III-D. Consumo del cómputo criptográfico

La Figura 7 muestra el *consumo energético promedio* exclusivamente asociado a las operaciones `KeyGen`, `Encaps` y `Decaps` de ML-KEM-512. Para cada fase se promediaron las mediciones de energía obtenidas en los dos escenarios de propagación (NLoS interior y LoS exterior), de modo que el resultado presentado recoge ambos entornos físicos.

Los valores promedio son inferiores a 0.2 J en todas las fases y representan menos del 5 % del consumo total del intercambio, confirmando que el costo de ML-KEM-512 en el procesador es marginal frente a la transmisión LoRa.

III-E. Fiabilidad del enlace

La Figura 8 resume la PER en ambos entornos y para todos los SF. En todos los casos, se mantiene por debajo del 5 %, incluso con fragmentación intensiva, lo que demuestra que

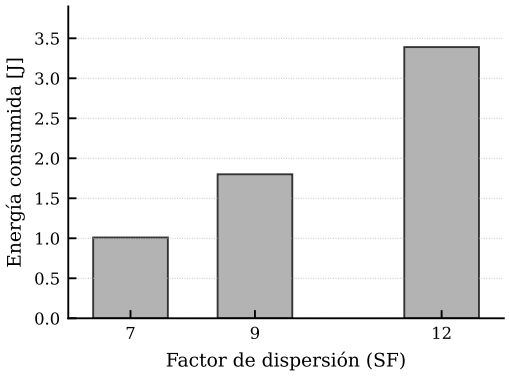


Figura 5. Energía total por intercambio ML-KEM-512 en entorno LoS.

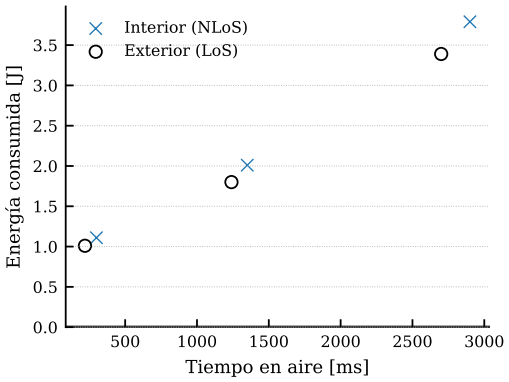


Figura 6. Correlación entre tiempo en aire y energía total por intercambio. Los puntos circulares corresponden al entorno LoS y las cruces al entorno NLoS, de modo que la diferencia se mantiene visible en impresiones en escala de grises.

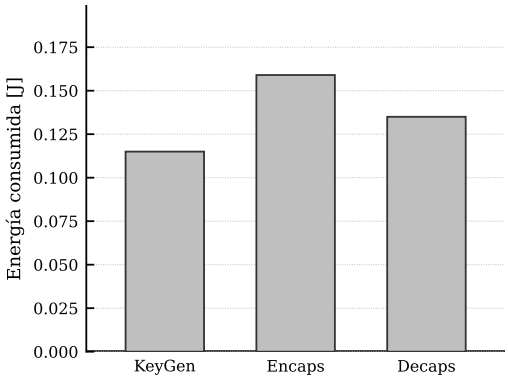


Figura 7. Consumo energético promedio del cómputo de ML-KEM-512 por fase (`KeyGen`, `Encaps`, `Decaps`), promediado entre los entornos NLoS y LoS.

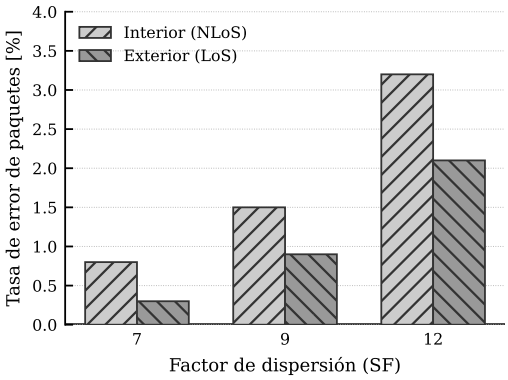


Figura 8. Tasa de error de paquetes (PER) en entornos NLoS y LoS.

el canal LoRa es lo suficientemente fiable para completar el intercambio autenticado sin mecanismos adicionales de corrección de errores.

IV. Análisis criptográfico y discusión

Los resultados experimentales permiten extraer conclusiones concretas sobre la viabilidad y las implicaciones de desplegar mecanismos poscuánticos en canales LoRa de baja capacidad. A partir de las mediciones de tiempo, energía y fiabilidad presentadas en las figuras, se discuten los siguientes aspectos clave:

- **Ejecutabilidad del KEM en hardware IoT:** El algoritmo ML-KEM-512 se ejecuta eficientemente en plataformas *gateway* como la RAK7248, mostrando tiempos de `Encaps()` y `Decaps()` del orden de pocos milisegundos. La Figura 7 evidencia que el costo energético del cómputo criptográfico representa menos del 5 % del total del intercambio, confirmando que el cuello de botella práctico no reside en la complejidad de la KEM sino en la capa física, donde el tiempo en aire crece exponencialmente con el factor de dispersión (SF).
- **Superficie de ataque y exposición temporal por fragmentación:** Aunque la seguridad formal IND-CCA2 de ML-KEM se mantiene intacta, la fragmentación impuesta por las limitaciones de carga útil amplía la *ventana de exposición temporal* del canal. Cuantos más fragmentos se requieran para transmitir los mensajes M_1 y M_2 que, recordemos, engloban pk, ct y los códigos de autenticación correspondientes, mayor es la posibilidad de que un adversario intente inyectar, reordenar o repetir tramas. Estas acciones no comprometen la confidencialidad ni la indistinguibilidad del esquema, pero requieren un tratamiento cuidadoso de numeración de fragmentos e integridad extremo a extremo.
  En el prototipo implementado, cada fragmento incluye un número de secuencia y el número total de fragmentos del mensaje, y el receptor reensambla, reordena y descarta duplicados antes de verificar el MAC global de M_1 o M_2. Este diseño limita el impacto de ataques de reordenamiento o repetición sobre un intercambio individual, aunque en despliegues masivos podrían requerirse mecanismos adicionales a nivel de red (ventanas anti-repetición, límites de frecuencia, etc.).
- **Validez del enfoque de autenticación ligera (PSK + Ascon-MAC):** El diseño autenticado propuesto en [6] se confirma como eficaz: tanto la clave pública pk como el criptograma ct se protegen mediante un MAC derivado de una PSK. La autenticación se aplica al objeto lógico completo (pk o $pk\|ct$), mientras que cada fragmento físico se transmite con CRC y numeración. Este enfoque evita firmas poscuánticas de gran tamaño y mantiene autenticación e integridad extremo a extremo sin necesidad de PKI.

- **Impacto de la capa de comunicación y líneas de optimización:** Las mediciones de energía y tiempo demuestran que las optimizaciones más relevantes se concentran en la capa de enlace. La correlación lineal entre el tiempo en aire y la energía consumida, mostrada en la Figura 6, evidencia que cualquier reducción del tiempo de radioencendido se traduce directamente en ahorro energético. Mecanismos como *Static Context Header Compression* (SCHC), compresión adaptativa de contexto o agregación de fragmentos pueden reducir el ToA sin modificar las propiedades criptográficas del KEM.
- **Resiliencia y secreto a futuro:** La combinación de seguridad IND-CCA2 con autenticación simétrica ligera preserva el *secreto a futuro* (*Perfect Forward Secrecy*, PFS). La exposición de la PSK o de una clave efímera no compromete sesiones anteriores, siempre que las claves se renueven por sesión. Esto es clave en arquitecturas IoT de borde, donde los nodos pueden operar desconectados y deben mantener autonomía criptográfica.

En conjunto, el análisis confirma que ML-KEM-512 es plenamente viable en entornos LoRa reales, siempre que la seguridad criptográfica se complemente con autenticación ligera basada en PSK y una gestión eficiente de la fragmentación. La Figura 7 demuestra que el cómputo del KEM representa una fracción marginal del costo total, mientras que la Figura 6 confirma que el consumo energético es proporcional al tiempo en aire. La fragmentación no constituye una debilidad del KEM, sino un desafío operativo que puede mitigarse mediante estrategias de transporte adaptativas y compresión de cabeceras.

V. Conclusiones y trabajos futuros

Este trabajo presenta una caracterización integral del comportamiento criptográfico, temporal y energético de ML-KEM-512 (FIPS 203) ejecutado sobre pasarelas LoRa en modo punto a punto. Los resultados muestran que el intercambio KEM autenticado es plenamente viable en plataformas ARM Cortex-A72 y que el costo total está dominado por la transmisión fragmentada, no por el cómputo criptográfico.

En coherencia con [6], se confirma que un esquema poscuántico basado en ML-KEM-512 puede integrarse en sistemas IoT de borde sin requerir PKI ni firmas digitales de gran tamaño. La combinación de encapsulación IND-CCA2 con autenticación simétrica ligera mediante PSK y Ascon-MAC permite mantener PFS y garantizar integridad en canales LoRa de baja tasa y alta latencia, con un consumo compatible con nodos alimentados por batería.

Como líneas futuras se propone:

- Integrar mecanismos de fragmentación y compresión seguros basados en SCHC [13], [14] para reducir tiempo en aire y energía.
- Extender la evaluación a escenarios LoRaWAN *multi-hop* y arquitecturas con *network server*, analizando la gestión de claves efímeras PQC entre pasarelas y modelos híbridos clásico-PQC.
- Realizar estudios comparativos con otros KEM PQC estandarizados para explorar el equilibrio entre nivel de seguridad, tamaño de mensajes y consumo energético.
- Estudiar la posibilidad de ataques de canal lateral en pasarelas LoRa. El análisis presentado se centra en métricas de tiempo en aire y consumo energético *agregado*, y no constituye un estudio de resistencia frente a ataques de canal lateral.

En conjunto, los resultados proporcionan una base empírica y reproducible para la adopción realista de criptografía poscuántica en redes IoT de baja potencia, contribuyendo a cerrar la brecha entre los estándares del NIST y las restricciones físicas de las tecnologías LPWAN.

Agradecimientos

Este trabajo se enmarca en la línea de investigación conjunta entre la Universidad de Buenos Aires, la Universidad Austral y la Universidad de Alcalá, y continúa la propuesta iniciada en RECSI 2024. Los autores agradecen el apoyo del CSC-CONICET y del proyecto P2QProMeTe (PID2020-112586RB-I00).

Referencias

[1] P. W. Shor, "Algorithms for Quantum Computation: Discrete Logarithms and Factoring," in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE Computer Society, 1994, pp. 124–134. doi:10.1109/SFCS.1994.365700.

[2] National Institute of Standards and Technology (NIST), "Module-Lattice-Based Key-Encapsulation Mechanism (ML-KEM)," FIPS PUB 203, August 2024, U.S. Department of Commerce, Gaithersburg, MD. doi:10.6028/NIST.FIPS.203.

[3] J. W. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé, "CRYSTALS – Kyber: A CCA-Secure Module-Lattice-Based KEM," in *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2018, pp. 353–367. doi:10.1109/EuroSP.2018.00032.

[4] F. Adelantado, X. Vilajosana, P. Tuset-Peiró, B. Martinez, J. Melià-Seguí, and T. Watteyne, "Understanding the Limits of LoRaWAN," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, 2017. doi:10.1109/MCOM.2017.1600613.

[5] L. Casals, B. Mir, R. Vidal, and C. Gómez, "Modeling the Energy Performance of LoRaWAN," *Sensors*, vol. 17, no. 10, p. 2364, 2017. doi:10.3390/s17102364.

[6] R. Sumoza Matos, E. E. Mocskos, and R. Durán Díaz, "Intercambio de clave poscuántico autenticado en entornos IoT mediante LoRa/LoRaWAN," in *Actas de la XVIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2024)*, A. Suárez Corona, Ed. León, España: Servicio de Publicaciones de la Universidad de León, 2024, pp. 349–354.

[7] National Institute of Standards and Technology (NIST), "Ascon-Based Lightweight Cryptography Standards for Constrained Devices," NIST SP 800-232, August 2025, U.S. Department of Commerce, Gaithersburg, MD. doi:10.6028/NIST.SP.800-232.

[8] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, "Ascon MAC, PRF, and Short-Input PRF," in *Topics in Cryptology – CT-RSA 2024*, E. Oswald, Ed. Cham: Springer Nature Switzerland, 2024, pp. 381–403. doi:10.1007/978-3-031-58868-6_15.

[9] Actility. (2022) LoRaWAN EU863–870 MHz Band Specifications. Accessed: 2025-10-30. Available: https://www.actility.com/lorawan-eu863-870-mhz-band/

[10] The Things Network Community. (2017) Fair Use Policy Explained. Accessed: 2025-10-30. Available: https://www.thethingsnetwork.org/forum/t/fair-use-policy-explained/1300

[11] Amazon Web Services, "Implementing LPWAN Solutions with AWS: LoRaWAN," Amazon Web Services, Tech. Rep., 2023, whitepaper, Accessed: 2025-10-30. Available: https://docs.aws.amazon.com/whitepapers/latest/implementing-lpwan-solutions-with-aws/lorawan.html

[12] S. Corporation, "AN1200.22 – LoRa Modem Designer's Guide," Semtech Corporation, Tech. Rep., 2015, application Note, Accessed: 2025-10-30. Available: https://lora-developers.semtech.com/documentation/tech-papers-and-guides/an1200.22-lora-modem-designers-guide/

[13] C. Bormann and L. Toutain, "SCHC: Generic Framework for Static Context Header Compression and Fragmentation," RFC 8724, 2020. Available: https://www.rfc-editor.org/rfc/rfc8724.html

[14] T. Gourmelon and A. Minaburo, "Static Context Header Compression and Fragmentation (SCHC) over LoRaWAN," RFC 9011, 2021. Available: https://www.rfc-editor.org/info/rfc9011

Multi-Solution Grover Search on Small Reversible Hash Functions: Simulation and Noisy Quantum Hardware Experiments

Jon Hernández-Bueno, Julen Bernabé-Rodriguez
TECNALIA, Basque Research and Technology Alliance (BRTA),
Bizkaia Science and Technology Park, Building 700, E-48160 Derio, Bizkaia, Spain.
Department of Communications Engineering, University of the Basque Country (UPV/EHU). Faculty of Engineering of Bilbao,
Plaza Ingeniero Torres Quevedo, n.1, Bilbao, 48013, Spain.
jon.hernandez@tecnalia.com

***Abstract*—We present an experimental study of Grover's quantum search algorithm in the multi-solution regime motivated by hash preimage and collision search. We design four families of small reversible toy hash functions with different Boolean structure and controlled numbers of preimages M over domains up to N = 2^{10}. One family is fully linear, two are non-linear three-input constructions, and one uses five input bits per output, increasing the non-linearity and circuit depth. For each family we build an explicit Qiskit oracle and run Grover search both on an ideal simulator and on a superconducting quantum processor (the IBM ibm_basquecountry backend), sweeping the number of iterations k around the standard optimal value and measuring the empirical success probabilities. Across all N,M pairs and hash families, the simulated behaviour matches the textbook prediction within statistical error, while on hardware the same circuits yield nearly flat success probabilities of a few percent, with the ideal oscillatory pattern washed out by noise and circuit depth. This provides a compact and reproducible laboratory for teaching and benchmarking multi-solution Grover search on hash-like problems, and simultaneously illustrates the limitations of current noisy quantum devices for implementing even small reversible hash oracles.**

***Index Terms*—Grover search, quantum cryptanalysis, hash functions, reversible oracles, quantum hardware**

I. Introduction

The prospect of large-scale quantum computers has triggered a systematic re-examination of the security of most deployed cryptographic primitives. While public-key schemes based on integer factorisation or discrete logarithms are known to be broken by Shor's algorithm [1], symmetric primitives and hash functions are also affected by quantum speed-ups. In particular, Grover's quantum search algorithm [2] provides a quadratic improvement over classical exhaustive search, so whenever the security of a scheme relies on the hardness of inverting or finding collisions for a hash function, Grover-type algorithms directly reduce the effective security level.

For an n-bit ideal hash, generic classical preimage search requires on the order of 2^n evaluations, while Grover reduces the query complexity to $O(2^{n/2})$ [3], [4]. For collision search, the classical $O(2^{n/2})$ query complexity can be further reduced by quantum collision-finding algorithms down to $O(2^{n/3})$ in the random-oracle model [5], [6]. These asymptotic results have been integrated into post-quantum cryptographic recommendations [7], [8] and underpin the common heuristic of increasing hash and key sizes in the presence of quantum adversaries.

Most of the existing literature on quantum attacks against hashes is, however, asymptotic and query-complexity oriented. Explicit small-scale implementations of hash-like oracles for Grover search, suitable as laboratories to understand multi-solution behaviour, finite-size effects and practical issues such as circuit depth and reversibility, have received comparatively less attention. Implementing a full standard hash such as SHA-2 or SHA-3 as a reversible circuit is far beyond current hardware and simulator capabilities, and even toy instances require substantial engineering effort.

Pedagogically, many introductions to Grover's algorithm focus on the single-solution case. In contrast, realistic cryptanalytic scenarios almost always involve multiple solutions: truncated outputs admit many preimages and any fixed output has exponentially many collisions. In this multi-solution regime, the success probability depends on both the database size N and the number of marked elements M, and the optimal number of iterations changes accordingly [11], [12]. Although this behaviour is well understood analytically, there are relatively few explicit, reproducible experiments exploring it systematically for hash-like oracles.

In this work we address these gaps by building and studying a family of small, reversible "toy" hash functions that approximate the structure of hash oracles while remaining tractable on current simulators and hardware. Each family has a different local Boolean structure, including a fully linear construction, two non-linear three-input maps and a more complex five-input non-linear variant, and for every output value we can efficiently compute the number of preimages M over domains up to $N = 2^{10}$. For each oracle we construct an explicit Qiskit implementation and use it to drive Grover search in the multi-solution setting relevant to hash preimage and collision search. We sweep the number of Grover iterations and compare the empirical success probability on an ideal simulator with the standard amplitude-amplification model, and we repeat the same experiments on a superconducting quantum processor to highlight the impact of noise and circuit depth.

Our goal is not to propose new cryptanalytic attacks, but to provide a compact, transparent and reusable framework for experimenting with multi-solution Grover search on hash-like

problems, and to illustrate how current hardware decoherence severely limits the achievable success probability for such oracles.

II. Background

A. Quantum Computing and Grover's Algorithm

In the circuit model of quantum computation [9], information is encoded in qubits, which may exist in coherent superpositions and entangled states, manipulated by unitary gates and finally measured in the computational basis. Frameworks such as Qiskit [10] provide high-level tools to design and simulate such circuits and form the basis of our implementation.

Grover's algorithm [2] addresses the unstructured search problem: given black-box access to a Boolean function $f : \{0,1\}^n \to \{0,1\}$, find an x such that $f(x) = 1$. Classically, the expected query complexity is $O(2^n)$; Grover reduces this to $O(2^{n/2})$ by iteratively amplifying the amplitudes of marked states through a sequence of reflections. The dynamics can be analysed in the two-dimensional subspace spanned by the uniform superpositions of marked and unmarked states [11], [12]. Writing $N = 2^n$ for the total number of states and M for the number of marked ones, and defining $\sin^2\theta = M/N$, after k iterations the success probability is

$$P_{\mathrm{succ}}(k) = \sin^2\left((2k+1)\theta\right), \tag{1}$$

with optimal $k_{\mathrm{opt}} \approx \frac{\pi}{4}\sqrt{N/M}$. This multi-solution analysis is the theoretical model we test against our simulations.

Grover's algorithm can be understood as a special case of amplitude amplification [12], which boosts the success probability of a generic quantum subroutine. In our case the subroutine is uniform sampling over the domain, and the marked set consists of all preimages of a chosen hash output.

B. Hash Functions and Quantum Attacks

Cryptographic hash functions are deterministic mappings $H : \{0,1\}^* \to \{0,1\}^n$ that compress arbitrary-length inputs into fixed-length digests and are expected to satisfy preimage, second-preimage and collision resistance [13]. Modern constructions, such as Merkle–Damgård and sponge designs, underlie SHA-2 and SHA-3 [14], [15]. In the idealised random-oracle model, truncating inputs to m bits yields roughly 2^{m-n} preimages per output on average [16].

Quantum algorithms modify the generic security baseline. Applying Grover search to the preimage problem reduces the complexity from $O(2^n)$ to $O(2^{n/2})$ queries to H [3], [4]. For collisions, Brassard, Høyer and Tapp [5] and subsequent works [6], [17] reach $O(2^{n/3})$ queries in the random-oracle model. These results, while not immediately yielding full attacks on concrete standards, clearly motivate increased hash output lengths and have been incorporated into post-quantum recommendations [7], [8].

C. Reversible Oracles and Toy Hashes

To apply Grover's algorithm to hash-based problems, one must implement the hash function as a reversible quantum oracle, i.e., a unitary U_H mapping $|x\rangle|0\rangle$ to $|x\rangle|H(x)\rangle$. For real-world hash standards such as SHA-2 or SHA-3, constructing such an oracle requires many qubits, deep circuits and careful ancilla management [18], [19], quickly exceeding the capabilities of current devices and challenging simulators.

Consequently, small-scale experimental studies and pedagogical expositions often resort to simplified "toy" oracles that capture key structural aspects of hash functions (bitwise mixing, non-linearity, diffusion) while remaining tractable at modest input sizes [20], [21]. Our work follows this philosophy: we design several families of small reversible toy hashes with controllable numbers of preimages M for each output and use them as testbeds to study the quantitative agreement between the amplitude amplification model and concrete simulations.

III. Approach

We now describe the design of our toy hash families, their reversible implementation in Qiskit, and the experimental procedure used to compare the simulated probabilities $P_{\mathrm{sim}}(k)$ (and, when available, the hardware probabilities $P_{\mathrm{hw}}(k)$) with $P_{\mathrm{th}}(k)$.

A. Families of Toy Hash Functions

We consider functions

$$H : \{0,1\}^{n_{\mathrm{in}}} \to \{0,1\}^{n_{\mathrm{out}}},$$

with $n_{\mathrm{in}} \in \{6, 8, 10\}$ and $n_{\mathrm{out}} \in \{3, 4, 5\}$ so that the full input space can be exhaustively analysed classically. Each output bit depends on a small subset of input bits (three for hash1, hash2 and hash4, and five for hash3) with indices taken modulo n_{in}. The four families are:

- **Hash1:** Baseline non-linear construction where each output bit is $A \oplus (B \wedge \neg C)$.
- **Hash2:** Linear XOR-based variant with output bits $A \oplus B \oplus C$.
- **Hash3:** Five-input non-linear function $A \oplus (B \wedge C) \oplus (D \wedge E)$, which increases algebraic degree and circuit depth.
- **Hash4:** Majority-based hash with output bits $\mathsf{maj}(A, B, C)$.

These choices provide distinct algebraic properties (linearity, degree, symmetry) while keeping circuit complexity modest. For each family and $(n_{\mathrm{in}}, n_{\mathrm{out}})$ we classically compute the preimage multiplicities

$$M_y = |\{x \in \{0,1\}^{n_{\mathrm{in}}} : H(x) = y\}|$$

for all outputs y, and select targets with representative M.

For hash3 we use five input positions per output bit, which leads to higher non-linearity and a larger number of Toffoli gates in the reversible implementation.

B. Reversible Oracle Construction in Qiskit

Each family is lifted to a reversible quantum circuit acting on three registers: input X of size n_{in}, output F of size n_{out} initialised to $|0\rangle$, and a small ancilla register A (we fix $n_{\mathrm{anc}} = 2$) also initialised to $|0\rangle$. The circuit realises a unitary

$$U_H : |x\rangle|0\rangle|0_{\mathrm{anc}}\rangle \mapsto |x\rangle|H(x)\rangle|0_{\mathrm{anc}}\rangle,$$

using only CNOT and Toffoli gates with explicit uncomputation of temporary values. For hash1, hash2 and hash4 each output bit depends on three input lines, whereas hash3 uses five controls per output, which increases the Toffoli count but keeps the ancilla footprint bounded.

The Grover phase oracle O for a given target output $y^\star$ is implemented via the standard compute–mark–uncompute pattern: (i) apply U_H to compute $H(x)$ into F; (ii) apply a multi-controlled phase flip conditioned on $F = y^\star$, implemented by mapping $y^\star$ to an all-ones pattern with X gates, performing an $(n_{\text{out}}-1)$-controlled Z, and undoing the mapping; and (iii) apply $U_H^\dagger$ to erase $H(x)$ and reset ancillae. The diffuser on the input register is implemented in the usual way as a reflection about the uniform superposition, using Hadamards, X gates and a multi-controlled Z.

C. Experimental Design and Metrics

For each hash family and each $(n_{\text{in}}, n_{\text{out}}) \in \{(6,3),(8,4),(10,5)\}$ we proceed as follows:

1) Sample T random inputs x_0 and define targets $y^\star = H(x_0)$.
2) For each $y^\star$, enumerate all classical preimages
$$S = \{x \in \{0,1\}^{n_{\text{in}}} : H(x) = y^\star\}$$
and record $M = |S|$.
3) Compute $N = 2^{n_{\text{in}}}$ and $k_{\text{opt}} \approx \left\lfloor \frac{\pi}{4}\sqrt{N/M} + \frac{1}{2} \right\rfloor$.
4) For each k in the window $[k_{\text{opt}} - 3, k_{\text{opt}} + 3]$ (clipped at $k \geq 0$), build a Grover circuit with an input register prepared in uniform superposition, k repetitions of the oracle O followed by the diffuser, and measurement of the input register.
5) Execute the circuit with multiple shots and define the success probability $P_{\text{sim}}(k)$ as the fraction of shots corresponding to $x \in S$.

For each configuration we also evaluate the theoretical success probability

$$P_{\text{th}}(k) = \sin^2\big((2k+1)\theta\big), \qquad \theta = \arcsin\sqrt{M/N},$$

and estimate binomial standard deviations

$$\sigma_{\text{sim}}(k) = \sqrt{\frac{P_{\text{sim}}(k)\big(1 - P_{\text{sim}}(k)\big)}{5\,000}},$$

which we use as error bars in the plots. For a subset of configurations we also execute the same transpiled circuits on the IBM ibm_basquecountry device using the Qiskit Runtime Sampler interface, from which we obtain hardware estimates $P_{\text{hw}}(k)$ by applying the same aggregation over classical preimages.

IV. Implementation

Our experiments are implemented in Python using Qiskit. The code is organised into four layers. At the lowest level we implement the classical toy hash families as bitstring-to-bitstring maps. On top of this we build reversible quantum circuits for each family using only X, CNOT and Toffoli gates, which are then wrapped into Grover oracles and diffusers. Finally, we provide driver routines that generate complete Grover circuits, transpile them with Qiskit's preset pass managers, and execute them either on the AerSimulator backend or on the IBM ibm_basquecountry device via the Qiskit Runtime Sampler.

Below we summarise the core routines in pseudocode form, which correspond directly to the public functions of the reference implementation.

A. Classical Toy Hash Families

The four hash families are implemented as simple Boolean maps on bit strings in Algorithm 1. Each takes as input a classical bit string $x \in \{0,1\}^{n_{\text{in}}}$ and outputs n_{out} bits, where each output bit depends on a small subset of input positions (three for hash1, hash2 and hash4, and five for hash3) with indices wrapped modulo n_{in}.

Algorithm 1 ToyHashFamily

```
procedure ToyHash(x, n_out, hash_name)
    n_in ← length of x
    y ← empty bit string
    for i = 0 to n_out − 1 do
        a ← i mod n_in
        b, c ← offsets depending on hash_name
        A ← x[a], B ← x[b], C ← x[c]
        if hash_name = hash1 then
            f_i ← A ⊕ (B ∧ ¬C)
        else if hash_name = hash2 then
            f_i ← A ⊕ B ⊕ C
        else if hash_name = hash3 then
            d, e ← additional offsets depending on hash_name
            D ← x[d], E ← x[e]
            f_i ← A ⊕ (B ∧ C) ⊕ (D ∧ E)
        else if hash_name = hash4 then
            f_i ← 1 if A + B + C ≥ 2 else 0
        end if
        Append f_i to y
    end for
    return y
end procedure
```

This classical layer is used both to generate target outputs $y^\star = H(x_0)$ and to enumerate all classical preimages $\{x : H(x) = y^\star\}$ when computing multiplicities M and empirical success probabilities.

B. Reversible Hash Oracles in Qiskit

Each hash family is lifted to a reversible quantum gate using CNOT and Toffoli primitives, ensuring that the ancilla register returns to $|0\rangle$ after computation and uncomputation, as shown in Algorithm 2.

Based on this routine, the phase oracle for Grover search is implemented via the standard compute–mark–uncompute pattern in Algorithm 3.

From a resource-estimation viewpoint, this construction makes explicit the overhead imposed by reversibility. All hash families are implemented using only X, CNOT and Toffoli gates, so the resulting unitary U_H can be directly mapped to Clifford+T circuits if a more fine-grained cost model is desired. The ancilla register is deliberately kept small and reused across all output bits, which bounds the total width of the oracle and simplifies its deployment on devices with limited qubit counts. At the same time, the circuit structure closely mirrors the original Boolean definitions, making it easy to inspect and modify individual components.

The diffuser is implemented in the standard way as a reflection about the uniform superposition in the n_{in}-qubit input space, using one multi-controlled *MCX* gate.

Algorithm 2 BUILDREVERSIBLEHASHCIRCUIT

procedure BUILDREVERSIBLEHASHCIRCUIT(n_{in}, n_{out}, n_{anc}, `hash_name`)
  Allocate input register X with n_{in} qubits
  Allocate hash register F with n_{out} qubits (initialised to $|0\rangle$)
  Allocate ancilla register A with n_{anc} qubits (initialised to $|0\rangle$)
  $U_H \leftarrow$ empty QuantumCircuit on (X, F, A)
  for $i = 0$ to $n_{\text{out}} - 1$ **do**
    $a, b, c \leftarrow$ input indices for output bit i (depend on `hash_name`)
    if `hash_name` $=$ *hash1* **then**
      Apply CNOT from X_a to F_i ▷ $F_i \oplus= A$
      Apply X on X_c
      Apply Toffoli with controls (X_b, X_c) and target F_i ▷ $F_i \oplus= B \wedge \neg C$
      Apply X on X_c ▷ uncompute temporary negation
    else if `hash_name` $=$ *hash2* **then**
      Apply CNOT from X_a to F_i
      Apply CNOT from X_b to F_i
      Apply CNOT from X_c to F_i
    else if `hash_name` $=$ *hash3* **then**
      $a, b, c, d, e \leftarrow$ input indices for output bit i (depend on `hash_name`)
      Apply Toffoli with controls (X_b, X_c) and target A_0 ▷ $A_0 \leftarrow B \wedge C$
      Apply Toffoli with controls (X_d, X_e) and target A_1 ▷ $A_1 \leftarrow D \wedge E$
      Apply CNOT from X_a to F_i ▷ $F_i \oplus= A$
      Apply CNOT from A_0 to F_i ▷ $F_i \oplus= B \wedge C$
      Apply CNOT from A_1 to F_i ▷ $F_i \oplus= D \wedge E$
      Uncompute A_1 with Toffoli $(X_d, X_e) \rightarrow A_1$
      Uncompute A_0 with Toffoli $(X_b, X_c) \rightarrow A_0$
    else if `hash_name` $=$ *hash4* **then**
      Use Toffoli gates to write $(A \wedge B)$ and $(A \wedge C)$ into A_0 and A_1
      Use a Toffoli on (B, C) to update $F_i \oplus= B \wedge C$
      Apply CNOTs from A_0 and A_1 to F_i ▷ XOR of pairwise ANDs
      Uncompute A_1 and A_0 with inverse Toffolis
    end if
  end for
  return U_H as a named gate
end procedure

Algorithm 3 BUILDGROVERORACLE

procedure BUILDGROVERORACLE(n_{in}, n_{out}, n_{anc}, $y^\star$, `hash_name`)
  Build reversible hash gate $U_H \leftarrow$ BUILDREVERSIBLEHASHCIRCUIT($n_{\text{in}}, n_{\text{out}}, n_{\text{anc}}$, `hash_name`)
  Allocate registers: X (n_{in}), F (n_{out}), A (n_{anc})
  $O \leftarrow$ empty QuantumCircuit on (X, F, A)
  // 1. Compute hash
  Append U_H on (X, F, A)
  // 2. Phase flip on matching hash value
  for $i = 0$ to $n_{\text{out}} - 1$ **do**
    if $y_i^\star = 0$ **then**
      Apply X on F_i ▷ map $y^\star$ to all-ones
    end if
  end for
  Apply H on $F_{n_{\text{out}}-1}$
  Apply multi-controlled X (*MCX*) with controls $F_0, \ldots, F_{n_{\text{out}}-2}$ and target $F_{n_{\text{out}}-1}$
  Apply H on $F_{n_{\text{out}}-1}$
  for $i = 0$ to $n_{\text{out}} - 1$ **do**
    if $y_i^\star = 0$ **then**
      Apply X on F_i ▷ undo mapping
    end if
  end for
  // 3. Uncompute hash
  Append $U_H^\dagger$ on (X, F, A)
  return O as a named oracle gate
end procedure

C. Grover Circuit Construction and Sweep Drivers

The core experimental routine builds a complete Grover circuit for a fixed configuration $(\texttt{hash_name}, n_{\text{in}}, n_{\text{out}}, y^\star, k)$ and executes it, as in Algorithm 4. The empirical success probability is obtained by aggregating counts corresponding to true preimages of $y^\star$.

The sweep over different iteration counts is organised so as to balance statistical accuracy and total runtime. For each configuration we restrict to a small window around k_{opt}, which is where the dependence of $P_{\text{succ}}(k)$ is most relevant from a cryptanalytic perspective. The number of shots is fixed globally, so that variations in the simulated success probability $P_{\text{sim}}(k)$ across hash families and sizes can be attributed to the underlying dynamics rather than to changes in sampling accuracy. In addition, all random choices (such as the selection of targets x_0) are controlled by an explicit seed to guarantee full reproducibility of the experiments.

Algorithm 4 RUNGROVERFORTARGET

procedure RUNGROVERFORTARGET(`hash_name`, n_{in}, n_{out}, $y^\star$, k, shots)
  $N \leftarrow 2^{n_{\text{in}}}$
  // 1. Enumerate classical preimages
  $S \leftarrow \{x \in \{0,1\}^{n_{\text{in}}} : \text{TOYHASH}(x, n_{\text{out}}, \texttt{hash_name}) = y^\star\}$
  $M \leftarrow |S|$
  if $M = 0$ **then**
    return failure (no solutions in domain)
  end if
  // 2. Build Grover circuit
  Allocate quantum registers: X (n_{in}), F (n_{out}), A (n_{anc})
  Allocate classical register C of size n_{in}
  $G \leftarrow$ empty QuantumCircuit on (X, F, A, C)
  Apply H to all qubits in X ▷ prepare uniform superposition
  $O \leftarrow$ BUILDGROVERORACLE($n_{\text{in}}, n_{\text{out}}, n_{\text{anc}}, y^\star$, `hash_name`)
  $D \leftarrow$ standard diffuser on X
  for $i = 1$ to k **do**
    Append O on (X, F, A)
    Append D on X
  end for
  Measure X into C
  // 3. Simulation and statistics
  Transpile G for the AerSimulator backend or quantum computer
  Execute G with given number of shots; collect counts $c[\cdot]$
  // 4. Compute empirical success probability
  $s \leftarrow 0$
  for each bit string z in support of c **do**
    $x \leftarrow$ reverse-bit-order(z) ▷ convert from Qiskit convention
    if $x \in S$ **then**
      $s \leftarrow s + c[z]$
    end if
  end for
  $P_{\text{sim}} \leftarrow s/\text{shots}$
  $P_{\text{th}} \leftarrow \sin^2\big((2k+1)\arcsin\sqrt{M/N}\big)$
  return $(P_{\text{sim}}, P_{\text{th}}, M, N, S, c[\cdot])$
end procedure

Algorithm 5 SWEEPOVERITERATIONSFORTARGET

procedure SWEEPOVERITERATIONSFORTARGET(`hash_name`, n_{in}, n_{out}, $y^\star$, shots, k_{margin})
  Compute M and N as in RUNGROVERFORTARGET; abort if $M = 0$
  $k_{\text{opt}} \leftarrow \max\Big(1, \Big\lfloor \frac{\pi}{4}\sqrt{N/M} + \frac{1}{2} \Big\rfloor\Big)$
  $k_{\min} \leftarrow \max(0, k_{\text{opt}} - k_{\text{margin}})$
  $k_{\max} \leftarrow k_{\text{opt}} + k_{\text{margin}}$
  $\mathcal{R} \leftarrow$ empty list of result records
  for $k = k_{\min}$ to $k_{\max}$ **do**
    $(P_{\text{sim}}, P_{\text{th}}, M, N, S, c[\cdot]) \leftarrow$ RUNGROVERFORTARGET(`hash_name`, $n_{\text{in}}, n_{\text{out}}, y^\star, k$, shots)
    Append record $\{\texttt{hash_name}, n_{\text{in}}, n_{\text{out}}, y^\star, N, M, k_{\text{opt}}, k, P_{\text{sim}}, P_{\text{th}}\}$ to $\mathcal{R}$
  end for
  return $\mathcal{R}$
end procedure

Algorithm 6 GLOBALMULTISOLUTIONSWEEP

procedure GLOBALMULTISOLUTIONSWEEP(shots, k_{margin})
  Fix hash family list $\mathcal{H} = \{$*hash1*, *hash2*, *hash3*, *hash4*$\}$
  Fix size configurations $\mathcal{C} = \{(6,3), (8,4), (10,5)\}$
  $\mathcal{R}_{\text{all}} \leftarrow$ empty list
  for each `hash_name` $\in \mathcal{H}$ **do**
    for each $(n_{\text{in}}, n_{\text{out}}) \in \mathcal{C}$ **do**
      for $t = 1$ to T **do** ▷ T random targets per configuration
        Sample random input $x_0 \in \{0,1\}^{n_{\text{in}}}$
        $y^\star \leftarrow \text{TOYHASH}(x_0, n_{\text{out}}, \texttt{hash_name})$
        $\mathcal{R} \leftarrow$ SWEEPOVERITERATIONSFORTARGET(`hash_name`, $n_{\text{in}}, n_{\text{out}}, y^\star$, shots, k_{margin})
        Append all records in $\mathcal{R}$ to $\mathcal{R}_{\text{all}}$
      end for
    end for
  end for
  Export $\mathcal{R}_{\text{all}}$ as a CSV-like summary table
  return $\mathcal{R}_{\text{all}}$
end procedure

V. Results

We now report the behaviour of Grover's algorithm in the multi-solution regime for the four toy hash families. For each configuration $(n_{\text{in}}, n_{\text{out}}) \in \{(6,3),(8,4),(10,5)\}$ and several random targets $y^\star$, we run the Grover circuit for a small window of iteration counts k around the theoretical optimum, executing the same transpiled circuit both on the AerSimulator and, for selected cases, on the IBM ibm_basquecountry device. From 5 000 shots per run we obtain a simulated success probability $P_{\text{sim}}(k)$ and, when using real hardware, a corresponding estimate $P_{\text{hw}}(k)$, which we compare against $P_{\text{th}}(k)$ in Equation (1).

A. Per-Family Behaviour

For the balanced family hash1, multiplicities follow the expected pattern $M \approx 2^{n_{\text{in}}-n_{\text{out}}}$ (e.g., $M = 8$ for $N = 64$, $M = 16$ for $N = 256$, $M = 32$ for $N = 1024$). Figure 1 shows a representative example with $(n_{\text{in}}, n_{\text{out}}) = (8, 4)$ and $M = 16$: the simulated success probability $P_{\text{sim}}(k)$ closely follows the sinusoidal curve $P_{\text{th}}(k)$ across the full sweep, with a peak around $k = 3$ reaching $P_{\text{sim}} \approx 0.96$. In contrast, the hardware curve $P_{\text{hw}}(k)$ stays in the few-percent range for all $k \in [0, 6]$ (between approximately 0.06 and 0.13) and does not exhibit a clear oscillatory pattern, illustrating the impact of noise and circuit depth.

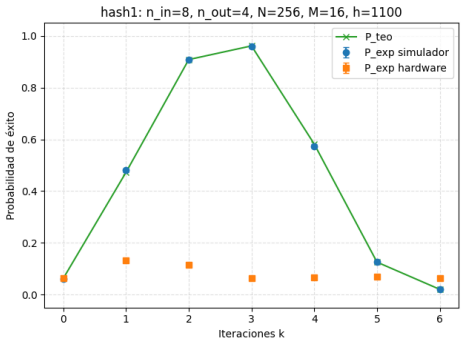


Figure 1. Success probability sweep for hash1 with $n_{\text{in}} = 8$, $n_{\text{out}} = 4$ and $M = 16$.

The XOR-based family hash2 has essentially the same multiplicity profile as hash1 but a fully linear structure. The global behaviour of the simulator is very similar: for all $(n_{\text{in}}, n_{\text{out}})$ and sampled targets, $P_{\text{sim}}(k)$ tracks $P_{\text{th}}(k)$ within the statistical error bars. Figure 2 illustrates a typical case with $(n_{\text{in}}, n_{\text{out}}) = (10, 5)$ and $M = 32$, where the peak around $k = 4$ is essentially saturated ($P_{\text{sim}} \approx 1$) and the side lobes are also well reproduced. On hardware, however, the corresponding success probabilities remain almost flat between approximately 0.03 and 0.05 for all sampled k, with no visible oscillations.

The non-linear family hash3 (with five input bits per output) leads to more skewed multiplicities, ranging from very small values (e.g., $M = 3$ for $N = 64$) to large ones (e.g., $M \gg 1$ for $N = 1024$). For small M, the sinusoid in k is more stretched and the optimal k increases; for large M the peak broadens. In both cases the simulated curves follow the theoretical envelope. Figure 3 shows a highly multi-solution regime with a broad maximum that matches the prediction

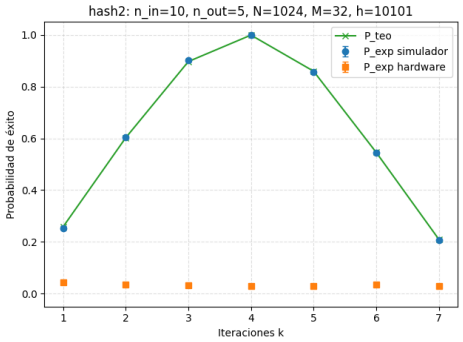


Figure 2. Success probability sweep for hash2 with $n_{\text{in}} = 10$, $n_{\text{out}} = 5$ and $M = 32$.

and a simulated peak near $P_{\text{sim}} \approx 1$ around $k = 4$. The hardware measurements for the same configuration remain again at a roughly constant level of about 3% across all k, without reproducing the ideal oscillatory structure.

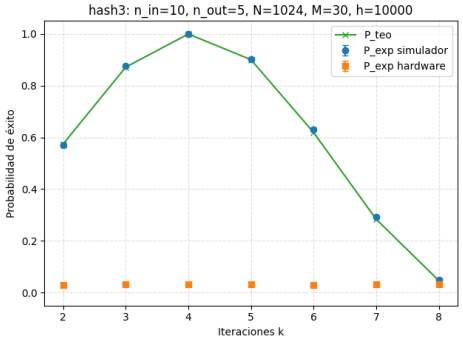


Figure 3. Success probability sweep for hash3 with $n_{\text{in}} = 10$, $n_{\text{out}} = 5$ and $M = 30$.

Finally, the majority-based family hash4 also exhibits non-uniform multiplicities (e.g., $M = 6$, 10, 32, 48 across different targets), but the resulting amplification patterns in the simulator remain well captured by $P_{\text{th}}(k)$. As an example, Figure 4 displays a configuration with $(n_{\text{in}}, n_{\text{out}}) = (10, 5)$ and $M = 48$, where the peak around $k = 3$–4 reaches a success probability near $P_{\text{sim}} \approx 1$ and the oscillatory decay for larger k matches the theory. On hardware, in contrast, the success probability stays in the range $P_{\text{hw}}(k) \approx 0.04$ for all k, again without a discernible sinusoidal pattern.

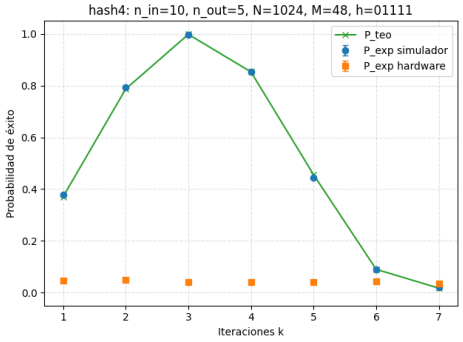


Figure 4. Success probability sweep for hash4 with $n_{\text{in}} = 10$, $n_{\text{out}} = 5$ and $M = 48$.

B. Error statistics

Across all hash families, input sizes and targets, the simulated success probabilities $P_{\mathrm{sim}}(k)$ closely follow the theoretical prediction $P_{\mathrm{th}}(k)$ in Equation (1). For the balanced families hash1 and hash2 the maximum deviation $|P_{\mathrm{sim}}(k) - P_{\mathrm{th}}(k)|$ over all configurations and k stays below ≈ 0.02, while for the more skewed families hash3 and hash4 it remains below ≈ 0.03. Since the binomial standard deviation for $5\,000$ shots is of order 10^{-2}, these differences are compatible with finite-sampling noise.

In contrast, the hardware success probabilities $P_{\mathrm{hw}}(k)$ differ substantially from $P_{\mathrm{th}}(k)$, with values in the few-percent range across all k and no clear sinusoidal structure. This behaviour reflects the fact that, on the chosen device, gate and measurement errors accumulated over deep circuits dominate over the ideal Grover dynamics, effectively limiting the observable success probability for our toy hash oracles.

This behaviour is summarised in Table I, where for one representative configuration of each family the simulated values P_{sim} essentially coincide with P_{th}, whereas the corresponding hardware values P_{hw} remain two orders of magnitude smaller.

Table I
REPRESENTATIVE VALUES OF THEORETICAL, SIMULATED AND HARDWARE SUCCESS PROBABILITIES FOR ONE CONFIGURATION OF EACH HASH FAMILY (ALL RUNS WITH 5 000 SHOTS).

Hash	n_{in}	n_{out}	k	P_{th}	P_{sim}	P_{hw}	$Gates_{hw}$
hash1	8	4	3	0.961	0.958	0.063	5425
hash2	10	5	4	0.999	0.999	0.030	8488
hash3	10	5	4	0.999	0.999	0.033	22524
hash4	10	5	3	0.998	0.998	0.039	19131

Table I also reports the total gate count after transpilation for the four representative configurations. Even for the smallest instance hash1, the transpiled circuits reach more than $5{,}000$ elementary gates, while for the larger non-linear families they exceed $19{,}000$ elementary gates, which is consistent with the accumulated error obtained.

C. Hardware setup

The hardware experiments were carried out on the IBM Quantum backend ibm_basquecountry. This device is a 156-qubit Heron r2 superconducting processor.

All circuits were constructed, transpiled and executed using Qiskit version 2.1.0, with access to IBM Quantum systems in November 2025.

VI. CONCLUSIONS

We have presented an experimental study of Grover's quantum search algorithm in the multi-solution regime using four families of small reversible hash functions with controlled preimage multiplicities. By explicitly constructing oracle circuits in Qiskit and running them on an ideal simulator, we swept the number of Grover iterations k for a variety of (N, M) combinations and observed excellent agreement between the simulated success probabilities $P_{\mathrm{sim}}(k)$ and the textbook prediction $P_{\mathrm{th}}(k) = \sin^2((2k+1)\theta)$ across all Boolean structures and multiplicity profiles.

We also executed selected configurations on the IBM ibm_basquecountry superconducting device. In this noisy hardware setting the coherent amplitude-amplification pattern is almost completely washed out: the measured success probabilities $P_{\mathrm{hw}}(k)$ remain in the few-percent range for all k, reflecting the combined effect of multi-controlled operations, circuit depth and device error rates. These observations support the standard intuition that, for generic Grover-type attacks on hash functions, the dominant parameters in the ideal model are the domain size N and the number of marked preimages M, but they also highlight that current noisy quantum processors are far from being able to realise such attacks even for very small toy instances.

REFERENCES

[1] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proc. 35th Annu. Symp. Foundations of Computer Science (FOCS)*, 1994, pp. 124–134.

[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. 28th Annu. ACM Symp. Theory of Computing (STOC)*, 1996, pp. 212–219.

[3] G. Brassard, P. Høyer, and A. Tapp, "Quantum counting," in *Proc. 25th Int. Colloq. Automata, Languages and Programming (ICALP)*, ser. LNCS, vol. 1443. Berlin, Heidelberg: Springer, 1998, pp. 820–831.

[4] A. Montanaro, "Quantum algorithms: An overview," *npj Quantum Information*, vol. 2, no. 15023, 2016.

[5] G. Brassard, P. Høyer, and A. Tapp, "Quantum cryptanalysis of hash and claw-free functions," in *Proc. 3rd Latin American Symp. Theoretical Informatics (LATIN)*, ser. LNCS, vol. 1380. Berlin, Heidelberg: Springer, 1998, pp. 163–169.

[6] A. Chailloux, M. Naya-Plasencia, and A. Schrottenloher, "An efficient quantum collision search algorithm and implications on symmetric cryptography," in *Advances in Cryptology – ASIACRYPT 2017*, ser. LNCS, vol. 10624. Cham: Springer, 2017, pp. 211–240.

[7] D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, pp. 188–194, 2017.

[8] National Institute of Standards and Technology, "Post-Quantum Cryptography: Call for Proposals," 2016. [Online]. Available: https://csrc.nist.gov/projects/post-quantum-cryptography

[9] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th anniv. ed. Cambridge University Press, 2010.

[10] H. Abraham *et al.*, "Qiskit: An Open-source Framework for Quantum Computing," 2019. [Online]. Available: https://qiskit.org

[11] M. Boyer, G. Brassard, P. Høyer, and A. Tapp, "Tight bounds on quantum searching," *Fortschritte der Physik*, vol. 46, nos. 4–5, pp. 493–505, 1998.

[12] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, "Quantum amplitude amplification and estimation," in *Quantum Computation and Information*, ser. Contemp. Math., vol. 305. Providence, RI: American Mathematical Society, 2002, pp. 53–74.

[13] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996.

[14] National Institute of Standards and Technology, "Secure Hash Standard (SHS)," FIPS Publication 180-4, Aug. 2015.

[15] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "The Keccak reference," Submission to NIST (Round 3 of the SHA-3 competition), 2011.

[16] P. Rogaway and T. Shrimpton, "Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance," in *Fast Software Encryption – FSE 2004*, ser. LNCS, vol. 3017. Berlin, Heidelberg: Springer, 2004, pp. 371–388.

[17] M. Zhandry, "Quantum lightning never strikes the same state twice," in *Advances in Cryptology – EUROCRYPT 2019*, ser. LNCS, vol. 11477. Cham: Springer, 2019, pp. 408–438.

[18] M. Amy et al., "Estimating the cost of generic quantum preimage attacks on SHA-2 and SHA-3," in *Selected Areas in Cryptography – SAC 2016*, ser. LNCS, vol. 10532. Cham: Springer, 2017, pp. 317–337.

[19] M. Grassl, B. Langenberg, M. Roetteler, and R. Steinwandt, "Applying Grover's algorithm to AES: quantum resource estimates," in *Post-Quantum Cryptography – PQCrypto 2016*, ser. LNCS, vol. 9606. Cham: Springer, 2016, pp. 29–43.

[20] S. Ramos-Calderer, E. Bellini, J. I. Latorre, M. Manzano, and V. Mateu, "Quantum search for scaled hash function preimages," *Quantum Information Processing*, vol. 20, no. 5, art. 180, May 2021.

[21] IBM Quantum Experience, "Grover's algorithm with toy oracles," Qiskit textbook tutorial. Available: https://qiskit.org/textbook/

Inventarios criptográficos para una migración postcuántica ágil y segura

Juan Banga Pardo
GRADIANT
Vigo, ES
jbanga@gradiant.org

Santiago de Vicente Poutás
GRADIANT
Vigo, ES
sdevicente@gradiant.org

Xavier Martinez Luaña
GRADIANT
Vigo, ES
xmartinez@gradiant.org

***Resumen*—Los avances en computación cuántica amenazan la criptografía de clave pública y han impulsado la aparición de estándares postcuánticos junto con calendarios de migración establecidos por organismos oficiales. En este contexto, la agilidad criptográfica se vuelve esencial, pero resulta inalcanzable sin una visibilidad completa del uso criptográfico existente. La elaboración de un inventario integral que identifique algoritmos, bibliotecas, claves, certificados y protocolos es por tanto un requisito fundamental. Este trabajo presenta un marco operativo para construir y normalizar dichos inventarios mediante *Cryptographic Bill of Materials* (CBOM), combinando descubrimiento automatizado basado en análisis estático, dinámico y monitorización pasiva con procesos de correlación y seguimiento continuo. El enfoque habilita remediaciones automatizadas guiadas por riesgo y sitúa el inventario como elemento central en la transición hacia la criptografía postcuántica.**

***Index Terms*—inventarios criptográficos, agilidad criptográfica, PQC, CBOM**

I. Introducción

Los notables avances en la computación cuántica plantean una amenaza existencial para la criptografía convencional de clave pública. Un ordenador cuántico suficientemente potente ejecutando el algoritmo de Shor podría romper esquemas de cifrado y firma digital de uso generalizado (*e.g.*, RSA y ECC) lo que comprometería la seguridad de las comunicaciones en Internet, las transacciones financieras y otros sistemas críticos [1]. Para abordar esta situación, la comunidad criptográfica se ha movilizado para desarrollar algoritmos de criptografía postcuántica capaces de resistir ataques cuánticos. En particular, el Instituto Nacional de Estándares y Tecnología (NIST) de EE. UU. publicó en 2024 los tres primeros estándares de *post-quantum cryptography* (PQC) finalizados basados en Kyber, Dilithium y SPHINCS+ (FIPS 203, 204 y 205 respectivamente), y mantiene en progreso un estándar basado en Falcon (FIPS 206) [2]. A comienzos de 2025, NIST anunció HQC como el único mecanismo de encapsulación de claves (KEM) adicional a estandarizar [3]. La hoja de ruta de transición de NIST (NIST IR 8547) [4] identifica los estándares criptográficos actualmente vulnerables a ataques cuánticos e introduce sustituciones resistentes para guiar el calendario de migración de las organizaciones. En resumen, el mundo se prepara para una profunda renovación criptográfica con el fin de preservar la seguridad en la era cuántica.

Entre estos preparativos, la agilidad criptográfica ha emergido como un paradigma defensivo clave. La agilidad criptográfica se refiere a la capacidad de los sistemas para adaptarse o cambiar con rapidez los algoritmos y las primitivas criptográficas con ninguna o con mínima interrupción [5]. Un diseño criptográficamente ágil abstrae las operaciones criptográficas detrás de interfaces modulares, lo que permite incorporar nuevos algoritmos sin cambios sustanciales en el código o la infraestructura. Por ejemplo, el NIST define la agilidad criptográfica como la capacidad de sustituir y adaptar algoritmos criptográficos en protocolos, *software*, *hardware* e infraestructuras sin interrumpir sistemas en ejecución, manteniendo así la resiliencia frente a amenazas en evolución [5]. El beneficio de esta agilidad es claro: cuando una vulnerabilidad (como un ataque cuántico viable) amenaza un criptosistema, una organización ágil puede hacer la transición hacia algoritmos más seguros con rapidez, evitando los prolongados retrasos que a menudo aquejan a las migraciones criptográficas, con el riesgo que estos implican [6]. En el contexto de PQC, la agilidad criptográfica significa estar preparado para pivotar de algoritmos clásicos a alternativas postcuánticas (e incluso entre algoritmos PQC), con la menor disrupción posible.

Sin embargo, una organización no puede migrar aquello que no ve o desconoce. La agilidad criptográfica fracasará en la práctica si se carece de visibilidad sobre los algoritmos, bibliotecas y claves criptográficas usados en un entorno. Antes de migrar la criptografía, es imprescindible descubrirla e inventariarla. De hecho, las recomendaciones recientes y la investigación señalan de manera contundente a los inventarios criptográficos como requisito previo de cualquier migración ágil [1], [7]. Un inventario criptográfico es un catálogo exhaustivo de los activos criptográficos de una organización y de su uso; *e.g.*, qué aplicaciones y protocolos emplean RSA o ECC, dónde se utilizan certificados de seguridad en la capa de transporte (TLS) y bibliotecas criptográficas, qué longitudes de clave y modos de cifrado están en uso, etc. Las organizaciones que carecen de esta visibilidad corren el riesgo de tener puntos ciegos en su migración: si se pasa por alto la criptografía de un sistema crítico, este podría permanecer como un punto vulnerable y explotable por adversarios cuánticos. La construcción de un inventario criptográfico preciso se considera con razón el primer paso hacia la agilidad criptográfica y la preparación para PQC [7].

Aunque la importancia de la agilidad algorítmica (*e.g.*, bibliotecas criptográficas adaptativas y mecanismos de negociación de protocolos) está bien reconocida [6], [8], existe una brecha de investigación y práctica en lo que respecta a la agilidad criptográfica a nivel empresarial. Los trabajos previos se han centrado mayoritariamente en la agilidad de bajo nivel; por ejemplo, modularizar más las API de las bibliotecas o habilitar que protocolos como TLS negocien nuevos conjuntos de cifrado [6]. Esto establece una base

necesaria, pero por sí misma no resuelve el desafío más amplio de actualizar el uso de criptografía dentro de toda una organización; las empresas carecen de marcos prácticos para coordinar y realizar actualizaciones en entornos complejos y distribuidos. Migrar unos pocos algoritmos de forma aislada es sencillo; gestionar sistemas diversos, infraestructura heredada y componentes de terceros es mucho más complejo. La investigación sobre este aspecto operativo de la agilidad criptográfica sigue siendo limitada, ya que abarca criptografía, ingeniería de *software* y gobernanza de TI. En consecuencia, las organizaciones dependen de procesos manuales y ad hoc que son propensos a errores e insostenibles a largo plazo, lo que evidencia la necesidad urgente de estrategias sistemáticas y marcos bien definidos que preparen para PQC y futuras transiciones criptográficas [6].

Este artículo aborda esa brecha y presenta un marco práctico en el que los inventarios criptográficos actúan como habilitadores de la agilidad criptográfica. A diferencia de trabajos previos centrados en la negociación de algoritmos o en API modulares, nuestra contribución se basa en un enfoque operativo que: **(i)** integra técnicas de descubrimiento automatizado, **(ii)** consolida los hallazgos en una *Cryptographic Bill of Materials* (CBOM), y **(iii)** vincula los datos del inventario con flujos de remediación basados en el riesgo. Esto permite a las organizaciones sustituir algoritmos vulnerables, rotar claves y monitorizar continuamente su uso de criptografía con rapidez. Asimismo, sienta las bases para transiciones ágiles, no solo hacia PQC, sino también ante futuros cambios criptográficos.

Las secciones restantes de este artículo se organizan del modo siguiente. La **Sección 2** describe el panorama de amenazas y la motivación para desarrollar agilidad criptográfica en preparación para la migración a PQC. La **Sección 3** detalla los distintos enfoques para construir un inventario criptográfico, incluyendo metodologías para descubrir activos, bibliotecas, protocolos y políticas, y para consolidar resultados mediante el estándar CBOM de CycloneDX. La **Sección 4** presenta una metodología dividida en fases para implementar inventarios integrales y enlazarlos con flujos de remediación automatizados. La **Sección 5** examina consideraciones de despliegue a escala empresarial y analiza los límites de la automatización, así como los retos a nivel de gobernanza e integración. La **Sección 6** concluye con un resumen de los hallazgos clave y las implicaciones más amplias de sostener una agilidad criptográfica *quantum-resistant*.

II. Marco normativo y panorama de amenazas

La urgencia de migrar hacia primitivas resistentes a ataques cuánticos se basa en estimaciones que apuntan en la misma dirección: en las próximas dos décadas podrían emerger ordenadores cuánticos con relevancia criptográfica capaces de romper los algoritmos de clave pública más empleados; existen análisis que incluso contemplan escenarios aún más cercanos [1], [9]. La respuesta institucional se ha materializado en mandatos concretos. En Estados Unidos, el Memorando de Seguridad Nacional 10 (NSM-10) y el memorando OMB M-23-02 instruyen a inventariar de manera prioritaria los sistemas criptográficos y a presentar planes anuales de migración. La Unión Europea, a través de la Hoja de Ruta de Implementación Coordinada para la Transición a la Criptografía Postcuántica de 2025, exige disponer antes de finales de 2026 de una gestión madura del inventario de activos criptográficos para acelerar la transición [10]; alineándose con la Recomendación de la Comisión (EU) 2024/1101 [11]. En paralelo, la hoja de ruta de transición del NIST (NIST IR 8547) identifica estándares vulnerables e introduce sustituciones resistentes para guiar calendarios de migración [4]. En el ámbito español, la guía CCN-TEC 009 refuerza la centralidad del inventario como punto de partida organizativo [12]. Estas directrices convierten la agilidad criptográfica en un requisito operativo, no en una opción.

La experiencia histórica con DES, MD5 o ciertas longitudes de RSA demuestra que la sustitución de algoritmos resulta lenta y compleja cuando faltan visibilidad y gobierno [13]. Por ello, las guías actuales insisten en iniciar la migración a PQC mediante un inventario criptográfico exhaustivo apoyado en herramientas de descubrimiento automatizado, que permita identificar usos efectivos, priorizar riesgos y planificar sustituciones. En este sentido, el NCCoE del NIST cuenta con una línea de trabajo específica de descubrimiento criptográfico dentro de su proyecto PQC y, en 2024, CISA publicó una estrategia en la que se insta a adoptar capacidades de descubrimiento e inventario criptográficos automatizados (ACDI), haciendo hincapié en la combinación de análisis automatizado y revisión manual para evitar puntos ciegos. [5], [14].

Para ser eficaz, la agilidad debe integrarse con la automatización y el control mediante políticas. La migración criptográfica ejecutada manualmente es costosa, lenta y propensa a errores; por tanto, es inadecuada frente a amenazas aceleradas. La guía del NIST sobre agilidad insiste en los mecanismos operativos que preservan la interoperabilidad durante los cambios de algoritmo: un inventario actualizado, alimentado de manera continua por múltiples fuentes de descubrimiento, indica los cambios necesarios, mientras que las herramientas automáticas implementan la forma de aplicarlos con garantías y registros auditables [5]. Planificar e invertir con antelación en estas capacidades consolida una postura proactiva: cuando sea necesario desplegar criptografía postcuántica o retirar una primitiva, el proceso será eficiente, verificable y controlado.

III. Descubrimiento de componentes criptográficos en sistemas heterogéneos

Lograr agilidad criptográfica y preparación postcuántica depende fundamentalmente de desarrollar primero un inventario criptográfico integral [15]. Este inventario debe proporcionar visibilidad sobre todos los recursos criptográficos en uso y actuar como requisito previo para adaptar con eficacia los sistemas cuando sea necesario sustituir algoritmos o implementaciones debido a vulnerabilidades emergentes o a amenazas cuánticas. Una organización no puede planificar una migración a una criptografía resistente a ataques cuánticos ni reemplazar con rapidez algoritmos débiles sin conocer antes *qué* criptografía utiliza y *dónde*. La construcción de un inventario criptográfico heterogéneo que incluya activos, algoritmos y bibliotecas, es esencial para apoyar cualquier esfuerzo de modernización criptográfica y para habilitar respuestas ágiles a medida que evolucionan los requisitos de seguridad.

En este trabajo, distinguimos cuatro tipos distintos de inventarios criptográficos, que se detallan en las subsecciones siguientes. **(i) Inventario de activos criptográficos (certificados y claves)**: cataloga claves criptográficas, certificados,

secretos y otras credenciales criptográficas en todo el entorno. **(ii) Inventario de bibliotecas criptográficas (componentes *software*)**: identifica bibliotecas, módulos y algoritmos criptográficos utilizados en aplicaciones y sistemas, incluidas las dependencias de terceros o de código abierto. **(iii) Inventario de protocolos criptográficos y uso de algoritmos**: mapea los protocolos y algoritmos criptográficos en uso (*e.g.*, versiones de TLS o SSH y conjuntos de cifrado) para comprender cómo se protege la información en tránsito y en reposo. **(iv) Inventario de políticas o configuraciones criptográficas**: enumera políticas y configuraciones a nivel de sistema y específicas de aplicaciones que controlen el comportamiento criptográfico (*e.g.*, algoritmos permitidos o no permitidos, conjuntos de cifrado, modos de operación, longitudes de clave y versiones de protocolo).

III-A. Inventario de activos criptográficos (claves y certificados)

Esta categoría engloba aquellos catálogos exhaustivos de claves criptográficas, certificados digitales, secretos y demás credenciales empleadas en el entorno. Incluye detalles como algoritmos y tamaños de clave, emisores, fechas de expiración, ubicaciones de almacenamiento, etc. Estos elementos pueden encontrarse distribuidos en múltiples almacenes y servicios: archivos de claves (*e.g.*, ficheros PEM), almacenes de certificados de los sistemas operativos (como el Windows Certificate Store o los Java KeyStores), gestores de claves o secretos (*e.g.*, GnuPG Keyring, Windows Credential Manager), módulos de *hardware* (HSM, tarjetas inteligentes) accesibles vía interfaces como PKCS#11, servicios en la nube, entre otros. Mantener inventariado este conjunto distribuido de claves y certificados es fundamental para gestionar su ciclo de vida y garantizar la sustitución oportuna de aquellos que queden obsoletos o se vean comprometidos.

III-B. Inventario de bibliotecas y módulos criptográficos

Identificación de todas las bibliotecas *software*, módulos criptográficos y algoritmos utilizados en las aplicaciones y sistemas de la organización, incluyendo dependencias de terceros o de código abierto. Esto abarca tanto librerías explícitas (*e.g.*, OpenSSL, Bouncy Castle, GNUTLS) como implementaciones de algoritmos incrustadas en el código. Registrar las versiones y tipos de estas bibliotecas es crucial, ya que sus vulnerabilidades (*e.g.* el bug Heartbleed en OpenSSL [16]) pueden afectar transversalmente a muchos sistemas. Asimismo, detectar qué implementaciones criptográficas se están usando (*i.e.*, si una aplicación usa su propio cifrado en lugar de una librería estándar) ayuda a evaluar riesgos y planificar la migración.

III-C. Inventario de protocolos y algoritmos en uso

Mapeo de los protocolos criptográficos (*e.g.*, TLS, SSH e IPsec) y de los parámetros y algoritmos que se están negociando o utilizando activamente en las comunicaciones. De este modo, es posible saber, por ejemplo, qué versiones de TLS se admiten, qué suites de cifrado se negocian y con qué algoritmos específicos (RSA, ECDHE, AES, etc.). Para obtener esta información suele ser necesaria la monitorización pasiva del tráfico o el análisis de registros de conexiones seguras. Técnicas de *fingerprinting* pasivo de protocolos (como el *fingerprinting* de *handshakes* TLS) facilitan identificar de forma no intrusiva los algoritmos y parámetros empleados por los clientes y servidores. Estudios previos han demostrado la efectividad de este enfoque para caracterizar el despliegue real de TLS en Internet [17], [18].

III-D. Inventario de políticas o configuraciones criptográficas

Recopilación de artefactos que gobiernan el comportamiento criptográfico de un sistema y sus componentes o aplicaciones. Su utilidad consiste en verificar y aplicar la conformidad: definir algoritmos permitidos, longitudes, versiones de protocolo, preferencias de conjuntos de cifrado o requisitos de certificación, y bloquear de manera global las prácticas no autorizadas. En la taxonomía de descubrimiento del NIST, estos artefactos se clasifican como ≪objetos de fichero no ejecutables≫ que deben ser descubiertos y catalogados al mismo nivel que claves o certificados, ya que la postura criptográfica depende tanto de las primitivas en uso como de las reglas que las rigen [7]. Los sistemas operativos modernos ofrecen gestión de políticas a escala del sistema, por ejemplo, los perfiles ≪DEFAULT≫, ≪LEGACY≫, ≪FUTURE≫ y ≪FIPS≫ en distribuciones como Red Hat Enterprise Linux, que permiten imponer, de manera transversal, algoritmos y versiones, y rechazar por defecto las configuraciones no conformes [19]. Integrar estas políticas en el inventario permite realizar comprobaciones automatizadas contra estándares aprobados y deshabilitar algoritmos de manera sistemática cuando surgen nuevas amenazas o criterios regulatorios.

IV. Metodología por fases para la implementación de un inventario criptográfico

La metodología presentada se estructura en fases y se orienta a dos objetivos transversales: (i) habilitar la agilidad criptográfica como capacidad operativa sostenida y (ii) preparar a la organización para una transición progresiva y verificable hacia la criptografía postcuántica. Todas las fases buscan proporcionar visibilidad, trazabilidad y capacidad de acción sobre el uso criptográfico real, facilitando la sustitución de algoritmos vulnerables y la adopción de primitivas postcuánticas con rapidez, control y mínimo impacto operativo.

IV-A. Fase 1: Delimitación y definición de objetivos

El proceso de inventariado se inicia con la delimitación del perímetro, que abarca los sistemas y entornos que se cubrirán, y alineando el esfuerzo con los objetivos generales de agilidad criptográfica y preparación postcuántica. Esta fase establece los criterios operativos que regirán el resto del ciclo, incluyendo el nivel de exhaustividad exigido, las prioridades de cobertura, las restricciones de acceso y la gobernanza del dato. El entregable principal de esta fase es un documento de alcance y objetivos que formaliza el perímetro, define los supuestos y exclusiones, fija métricas de completitud y calidad, y asigna responsabilidades de negocio, operación y seguridad sobre los activos a inventariar.

IV-B. Fase 2: Planificación y ejecución del descubrimiento en múltiples capas

El descubrimiento debe realizarse a través de capas complementarias con el fin de observar toda forma de uso de

criptografía sin interrumpir los sistemas en producción. El análisis estático de código fuente y de archivos binarios revela las primitivas incluidas y los enlaces a bibliotecas; los inventarios de *software* a nivel de sistema enumeran los activos desplegados; el trazado dinámico captura invocaciones de API criptográficas en tiempo de ejecución; y la inspección pasiva de los *handshakes* de protocolo revela las versiones negociadas y los conjuntos de cifrado. Dado que ninguna técnica ofrece cobertura completa por sí sola, la planificación debe priorizar una combinación de herramientas coherente con la heterogeneidad del entorno, maximizando la cobertura e integración. El resultado de esta fase es un plan de descubrimiento, compuesto por un mapa de fuentes a inventariar y una matriz de herramientas y métodos de observación por capa.

IV-C. Fase 3: Normalización de datos y canonicalización semántica

Se abordan de forma sistemática las divergencias de formato y nomenclatura entre las distintas herramientas de descubrimiento, que producen salidas heterogéneas e incompatibles. Para obtener una visión coherente del uso criptográfico es necesario traducir todos los hallazgos a una representación común, de modo que familias de algoritmos, atributos de claves, parámetros de protocolo, identidades de bibliotecas y reglas de configuración queden descritos con una semántica unificada. Este paso de normalización busca unificar alias de proveedor y variantes de parámetros; por ejemplo, mapear las cadenas ≪RSA-2048≫, ≪rsa2048≫ y el OID 1.2.840.113549.1.1.1 a un único identificador de algoritmo. Resultando finalmente en un inventario criptográfico completo pero aún no consolidado.

En este contexto, la adopción del formato CBOM de CycloneDX proporciona el esquema de referencia para dicha normalización. CBOM extiende el concepto de inventario de componentes de software con atributos específicos de criptografía y ofrece un modelo legible por máquina en el que representar algoritmos, claves, certificados, bibliotecas y sus relaciones de dependencia [20]. Su uso permite integrar resultados procedentes de fuentes heterogéneas en un único modelo de datos, habilita la verificación automática frente a políticas y buenas prácticas, y facilita la priorización de riesgos mediante campos como el nivel de seguridad cuántica o el grado de certificación asociado a cada componente. De este modo, la normalización basada en CBOM convierte un conjunto de salidas dispares en un inventario cohesionado y accionable, apto para ser consumido por procesos de análisis, remediación y gobernanza criptográfica.

IV-D. Fase 4. Correlación, deduplicación y mapeo de dependencias

Una vez que los datos recopilados se han normalizado en una representación común, el siguiente paso consiste en enlazar los hallazgos de los distintos métodos de descubrimiento para que describan el mismo elemento lógico en lugar de aparecer como registros separados. Este procedimiento implica unir los objetos identificados durante el análisis del código fuente, el escaneo de binarios, la observación en tiempo de ejecución, la enumeración de activos y la inspección de protocolos. Además, se deben eliminar los duplicados cuando varias técnicas detectan el mismo componente. El proceso también debe mapear las relaciones entre componentes, cómo una aplicación que depende de una biblioteca criptográfica, cómo esa biblioteca utiliza una clave o un certificado concretos y cómo el sistema resultante negocia parámetros de protocolo durante la operación. El resultado es un inventario CBOM consolidado, enriquecido con un grafo de dependencias y trazabilidad cruzada hacia las evidencias originales, de forma que el impacto de modificar o retirar un elemento pueda analizarse de manera fundada y verificable.

IV-E. Fase 5. Aseguramiento de la calidad y clasificación de riesgos

Con un inventario consolidado e interrelacionado, la prioridad pasa a ser confirmar la exactitud y evaluar el riesgo. Cuando la información recopilada no sea suficiente, la verificación dirigida o el muestreo pueden mejorar la fiabilidad. Cada elemento se somete a una evaluación integral de su postura de seguridad que tiene en cuenta la fortaleza del algoritmo, las debilidades conocidas de la implementación, los protocolos en los que se utiliza y su exposición dentro del sistema. Para apoyar la priorización, deben adjuntarse metadatos adicionales, como el nivel de seguridad cuántico o los niveles de certificación de cumplimiento normativo. Este sistema de clasificación permite identificar con rapidez los problemas más urgentes, como componentes vulnerables o algoritmos que quedarán obsoletos en un futuro próximo, y, al mismo tiempo, sustenta la planificación de transiciones criptográficas a largo plazo mediante la generación del entregable de la fase, consistente en una matriz de riesgos en la que cada elemento quede clasificado y asociado de forma explícita a su evidencia y justificación, habilitando así la priorización de acciones de remediación y seguimiento.

IV-F. Fase 6. Integración y automatizaciones

Tras la validación y la clasificación del inventario, su valor práctico depende de conectarlo con los sistemas capaces de actuar sobre sus hallazgos. Esta fase se ocupa de establecer interfaces con una variedad de herramientas y sistemas, entre ellos gestores de certificados, sistemas de gestión de claves (KMS), plataformas de gestión de configuración, *pipelines* de integración y entrega continuas (CI/CD) y autoridades de certificación (CA). Gracias a estas conexiones, las actualizaciones de elementos criptográficos pueden coordinarse de forma controlada y repetible. Por ejemplo, las claves y los certificados marcados para sustitución pueden rotarse masivamente, las bibliotecas criptográficas obsoletas pueden actualizarse durante el proceso de compilación y la configuración de los protocolos puede reforzarse en múltiples sistemas sin reconfiguración manual. Aunque estas integraciones permiten cambios rápidos y a gran escala, deben diseñarse con salvaguardas que garanticen la supervisión adecuada de las modificaciones de alto impacto, mientras que las actualizaciones previsibles y de bajo riesgo pueden ejecutarse de manera automática.

IV-G. Fase 7. Monitorización continua y gestión del ciclo de vida

La fase final garantiza que el inventario se mantenga preciso y pertinente con el paso del tiempo. Para ello se establecen escaneos periódicos y actualizaciones basadas en eventos que

se activan cuando se despliega un código nuevo, cuando se producen cambios de configuración o cuando se modifican activos criptográficos. Cualquier desviación respecto de las políticas criptográficas aprobadas puede detectarse con rapidez y es posible iniciar de inmediato acciones de remediación. El mantenimiento de registros detallados de cambios facilita los procesos de auditoría y el cumplimiento normativo, y la incorporación de los resultados de cada actualización o remediación al inventario mejoran su precisión y su completitud. Este bucle de retroalimentación continuo convierte el inventario en un registro dinámico de la postura criptográfica de la organización, lo que le permite adaptarse con rapidez cuando evolucionan los estándares, surgen nuevas amenazas o se introducen tecnologías criptográficas novedosas, como los algoritmos postcuánticos. En consecuencia, el proceso se consolida como un habilitador persistente de la agilidad criptográfica en toda la organización.

V. Retos de despliegue a gran escala y consideraciones prácticas

El despliegue de un inventario criptográfico a escala empresarial introduce desafíos sustanciales debido a la heterogeneidad y la complejidad de las infraestructuras modernas. Las organizaciones operan en entornos diversos, que incluyen plataformas nativas en la nube, *mainframes* heredados, sistemas empotrados y servicios ejecutados en contenedores. Cada uno presenta restricciones particulares que dificultan el uso de una única herramienta de descubrimiento o remediación. Los agentes automatizados pueden ser incompatibles con sistemas anticuados y las herramientas de análisis estático suelen no detectar operaciones criptográficas que se cargan dinámicamente o que están ofuscadas en tiempo de ejecución.

Tal como subrayan las guías sobre preparación cuántica, el descubrimiento automatizado a menudo debe complementarse con procesos manuales para asegurar una visibilidad completa [1]. Incluso cuando el uso criptográfico es técnicamente detectable, cuestiones operativas como una titularidad poco clara, una clasificación de activos incompleta o la ausencia de metadatos sobre el ciclo de vida pueden obstaculizar la gobernanza y retrasar los esfuerzos de remediación. Por tanto, la implantación de un inventario criptográfico de alcance empresarial no solo es una tarea técnica, sino también un desafío organizativo que requiere la estrecha colaboración de los equipos de seguridad, operaciones e ingeniería de *software*.

La capa de activos de un inventario criptográfico cumple un papel fundamental al identificar claves y certificados en almacenes del sistema operativo, en HSM y en sistemas de archivos no gestionados. Una vez poblado, este inventario se convierte en un punto de consulta para la activación de operaciones automatizadas del ciclo de vida de los activos mediante API bien definidas y motores de políticas. La integración de los inventarios criptográficos con infraestructuras de clave pública (PKI) y con sistemas de gestión de claves (KMS) es esencial para habilitar esta automatización. Cuando el inventario criptográfico está integrado con gestores de certificados, este permite preparar sustituciones resistentes a ataques cuánticos antes de que sea demasiado tarde. Del mismo modo, los clústeres de HSM pueden ejecutar migraciones de claves cuando detectan que los algoritmos resistentes a ataques cuánticos se han vuelto viables en términos de compatibilidad o se han convertido en requisito normativo.

El impacto de esta capacidad se aprecia con especial claridad a nivel de activo. Cuando un inventario revela un certificado autofirmado con un algoritmo desfasado, puede activarse un proceso de regeneración automatizada usando primitivas más robustas. Si el certificado fue emitido por una CA, un agente automatizado puede iniciar una reemisión segura y redistribuir certificados actualizados. Estas acciones reducen de forma sustancial el tiempo de remediación y la intervención manual de los administradores, factores críticos para cumplir los requisitos normativos a tiempo.

Conviene señalar que ciertos escenarios de remediación, como el recifrado de datos en reposo, no pueden automatizarse por completo. Estos procesos requieren descifrar y volver a cifrar los datos con claves nuevas, a menudo con ventanas temporales de mantenimiento estrictas y gran cantidad de lógica específica del sistema a migrar. Agentes con conocimiento del inventario criptográfico del sistema pueden ayudar a planificar estos pasos, pero no pueden ejecutarlos de forma independiente. Este límite entre automatización e intervención humana debe gestionarse con cuidado.

Las bibliotecas criptográficas, en especial las que se vinculan de forma dinámica y se distribuyen mediante gestores de paquetes, son candidatas idóneas para la remediación automatizada. Los *pipelines* de entrega continua pueden configurarse para sustituir módulos obsoletos por versiones actualizadas y preparadas para PQC. En cambio, las bibliotecas criptográficas enlazadas de forma estática o proporcionadas por un proveedor plantean mayores dificultades. Sin acceso al código fuente o sin la cooperación de terceros, los procedimientos de remediación deben abordarse de forma manual. Guiado por la información derivada del inventario criptográfico, un administrador puede identificar los algoritmos o primitivas que requieren sustitución y formular un plan de migración.

La telemetría a nivel de protocolo, que incorpora capturas pasivas de *handshakes* TLS o SSH y el trazado de funciones criptográficas, aporta una capa de información crítica. Estos registros revelan el uso real de algoritmos que los escaneos estáticos pueden pasar por alto. Cuando se combinan con componentes gestionados de forma centralizada, como *service meshes*, *reverse proxies* o API *gateways*, las políticas ligadas al inventario criptográfico pueden aplicar cambios en redes completas. Estos cambios pueden incluir la desaprobación de TLS 1.2 o la actualización de los conjuntos de cifrado admitidos. En sistemas menos centralizados o heredados, los cambios de protocolo pueden perturbar la lógica de operación, por lo que los flujos de remediación deben apoyarse en evaluaciones con contexto, como capturas de paquetes, *risk scoring* y grafos de dependencias.

Los *frameworks* o sistemas de gestión de políticas permiten imponer de forma declarativa y a gran escala el uso de algoritmos seguros. Cuando los inventarios señalan configuraciones desactualizadas, las herramientas automatizadas pueden actualizar miles de equipos desde perfiles ≪LEGACY≫ a ≪FUTURE≫ o ≪PQC-READY≫. No obstante, los administradores deben revisar y aprobar los cambios que puedan afectar a sistemas externos e integraciones.

A pesar del potencial para la automatización, las remedia-

ciones criptográficas se enfrentan a las limitaciones operativas del mundo real. Las migraciones de datos, los sistemas empotrados y una criptografía muy acoplada a la lógica de aplicación requieren criterio experto. Además, la ausencia de un contexto semántico suficiente en las herramientas puede llevar a clasificar erróneamente prácticas seguras o a proponer cambios arriesgados, con el consiguiente riesgo de falsos positivos o de un fallo del sistema. Un enfoque sistemático establece dos trayectorias: por un lado, flujos automatizados para cambios deterministas y de bajo riesgo; por otro lado, remediaciones dirigidas por personal experto, respaldadas por la información obtenida del inventario criptográfico, para escenarios complejos o de alto impacto.

Cada remediación, ya sea automatizada o manual, contribuye a la actualización continua del inventario y crea un registro de estado de la postura criptográfica que se mantiene al día y es auditable. Este bucle retroalimentado convierte al inventario en un sistema vivo, alineado con las normativas y con una adopción profunda de la agilidad criptográfica. Al integrar descubrimiento, evaluación, orquestación y gobernanza en un ciclo cerrado, las organizaciones pueden pasar de un enfoque reactivo, impulsado por situaciones de crisis, a una estrategia proactiva y escalable que favorezca la resiliencia cuántica a largo plazo con interrupciones mínimas.

VI. Conclusiones

La agilidad criptográfica ya no es una característica opcional, sino una necesidad para lograr una transición a PQC segura y oportuna. Aunque la negociación algorítmica y las API modulares constituyen la base técnica, no pueden tener éxito sin visibilidad sobre los activos criptográficos y sus dependencias a nivel de toda la organización. Esta visibilidad debe proporcionarla un inventario criptográfico integral.

El marco que proponemos en este trabajo utiliza CBOM para unificar el formato de agregación de salidas de descubrimiento heterogéneas en un modelo coherente. Al vincular un inventario criptográfico con automatizaciones en distintos servicios, como gestores de certificados, KMS, *pipelines* CI/CD o sistemas de gestión de políticas, las organizaciones podrán sustituir la criptografía vulnerable con mayor rapidez y con menos interrupciones. Esta metodología propuesta no solo respalda la migración a PQC, sino también futuros cambios criptográficos, puesto que resulta difícil pensar que los algoritmos actuales vayan a permanecer seguros indefinidamente.

Las mejoras futuras de este marco podrían centrarse en: **(i) integrar técnicas de detección basadas en IA** en el proceso de descubrimiento automatizado para identificar el uso criptográfico en binarios y código fuente, incluso bajo ofuscación o implementaciones no estándar; **(ii) diseñar mecanismos para inventarios criptográficos federados entre múltiples entidades**, garantizando la interoperabilidad segura y la gobernanza; y **(iii) ampliar el enfoque de remediación guiada por políticas** para adaptarse de forma dinámica a actualizaciones regulatorias, los marcos de cumplimiento normativo y a las necesidades de respuesta ante incidentes.

Referencias

[1] S. D. Young, "M-23-02 MEMORANDUM FOR THE HEADS OF EXECUTIVE DEPARTMENTS AND AGENCIES: Migrating to post-quantum cryptography." [Online]. Available: https://www.whitehouse.gov/wp-content/uploads/2022/11/M-23-02-M-Memo-on-Migrating-to-Post-Quantum-Cryptography.pdf

[2] G. Alagic, D. Apon, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, D. Smith-Tone, and Y.-K. Liu, "Status report on the third round of the NIST post-quantum cryptography standardization process." [Online]. Available: https://csrc.nist.gov/pubs/ir/8413/upd1/final

[3] G. Alagic, M. Bros, P. Ciadoux, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, H. Silberg, D. Smith-Tone, and N. Waller, "Status report on the fourth round of the NIST post-quantum cryptography standardization process." [Online]. Available: https://csrc.nist.gov/pubs/ir/8545/final

[4] D. Moody, R. Perlner, A. Regenscheid, A. Robinson, and D. Cooper, "Transition to post-quantum cryptography standards." [Online]. Available: https://csrc.nist.gov/pubs/ir/8547/ipd

[5] E. Barker, L. Chen, D. Cooper, D. Moody, A. Regenscheid, M. Souppaya, W. Newhouse, R. Housley, S. Turner, W. Barker, and K. Scarfone, "Considerations for achieving cryptographic agility: Strategies and practices (draft)." [Online]. Available: https://csrc.nist.gov/pubs/cswp/39/considerations-for-achieving-cryptographic-agility/2pd

[6] D. Ott, K. Paterson, and D. Moreau, "Where is the research on cryptographic transition and agility?" vol. 66, no. 4, pp. 29–32. [Online]. Available: https://dl.acm.org/doi/10.1145/3567825

[7] W. Newhouse, M. Souppaya, W. Barker, C. Brown, P. Kampanakis, M. Manzano, D. McGrew, A. Dames, V. Soukharev, P. Lafrance, A. Hu, D. Hook, R. Garcia, E. Gervis, E. Kim, and C. Lee, "Migration to post-quantum cryptography: Preparation for considering the implementation and adoption of quantum safe cryptography." [Online]. Available: https://csrc.nist.gov/pubs/sp/1800/38/iprd-(1)

[8] N. Alnahawi, N. Schmitt, A. Wiesmaier, A. Heinemann, and T. Grasmeyer, "On the state of crypto-agility." [Online]. Available: https://eprint.iacr.org/2023/487

[9] T. L. Scholten, C. J. Williams, D. Moody, M. Mosca, W. Hurley, W. J. Zeng, M. Troyer, and J. M. Gambetta, "Assessing the benefits and risks of quantum computers." [Online]. Available: http://arxiv.org/abs/2401.16317

[10] European Commission. A coordinated implementation roadmap for the transition to post-quantum cryptography. [Online]. Available: https://digital-strategy.ec.europa.eu/en/library/coordinated-implementation-roadmap-transition-post-quantum-cryptography

[11] European Comission. Comission recommendation (EU) 2024/1101 on a coordinated implementation roadmap for the transition to post-quantum cryptography. [Online]. Available: https://eur-lex.europa.eu/eli/reco/2024/1101/oj/eng

[12] Centro Criptológico Nacional. Recomendaciones para una transición postcuántica segura de CCN-PYTEC. [Online]. Available: https://www.ccn-cert.cni.es/es/seguridad-al-dia/novedades-ccn-cert/12234-recomendaciones-para-una-transicion-postcuantica-segura-de-ccn-pytec.html

[13] X. Wang and H. Yu, "How to break MD5 and other hash functions," in *Proceedings of the 24th annual international conference on Theory and Applications of Cryptographic Techniques*, ser. EUROCRYPT'05. Springer-Verlag, pp. 19–35. [Online]. Available: https://doi.org/10.1007/11426639_2

[14] Cybersecurity and Infrastructure Security Agency. Strategy for migrating to automated post-quantum cryptography discovery and inventory tools. [Online]. Available: https://www.cisa.gov/resources-tools/resources/strategy-migrating-automated-post-quantum-cryptography-discovery-and-inventory-tools

[15] N. Schmitt, J. Henrich, D. Heinz, N. Alnahawi, and A. Wiesmaier, "On criteria and tooling for cryptographic inventories."

[16] Heartbleed bug. [Online]. Available: https://heartbleed.com/

[17] M. Husák, M. Čermák, T. Jirsík, and P. Čeleda, "HTTPS traffic analysis and client identification using passive SSL/TLS fingerprinting," vol. 2016, no. 1, pp. 1–14, publisher: SpringerOpen. [Online]. Available: https://jis-eurasipjournals.springeropen.com/articles/10.1186/s13635-016-0030-7

[18] P. Kotzias, A. Razaghpanah, J. Amann, K. G. Paterson, N. Vallina-Rodriguez, and J. Caballero, "Coming of age: A longitudinal study of TLS deployment," in *Proceedings of the Internet Measurement Conference 2018*, ser. IMC '18. Association for Computing Machinery, pp. 415–428. [Online]. Available: https://dl.acm.org/doi/10.1145/3278532.3278568

[19] Red Hat. Using system-wide cryptographic policies. [Online]. Available: https://docs.redhat.com/en/documentation/red_hat_enterprise_linux/8/html/security_hardening/using-the-system-wide-cryptographic-policies_security-hardening

[20] OWASP Foundation. Cryptography bill of materials (CBOM). [Online]. Available: https://cyclonedx.org/capabilities/cbom/

The impact of aggregation algorithms on property inference attacks in federated learning

Adolf Felip Garcia*, Antonina Serebriakova†, Gorka Pineda Burgueño*, Cristina Pérez-Solà*, Victor Garcia-Font†
*Universitat Autònoma de Barcelona (UAB)
AdolfFelip.Garcia@autonoma.cat, Gorka.Pineda@uab.cat, Cristina.Perez@uab.cat
†Universitat Oberta de Catalunya (UOC)
aserebriakova@uoc.edu, vgarciafo@uoc.edu

***Abstract*—Federated learning enables multiple parties to collaboratively train a global model without transferring their raw data to a central server, which limits data exposure and strengthens privacy guarantees. However, users' privacy remains exposed to several potential attack vectors. Among these, property inference attacks aim to deduce hidden attributes of participants' local data by analyzing shared model updates. This paper examines how different aggregation algorithms (one of the components of the federated learning architecture) affect both the effectiveness of such attacks and the overall performance of the global model. Using TensorFlow Federated implementations of Federated Averaging (FedAvg), Federated Proximal (FedProx), and Federated Stochastic Gradient Descent (FedSGD), we empirically evaluate the relationship between the aggregation algorithm, property inference susceptibility, and global model performance. Our results show that the choice of aggregation strategy has a major impact on the risk of property inference attacks. In particular, the inclusion of a proximal term in FedProx improves resistance to inference attacks without degrading classification accuracy. Nonetheless, the results should be interpreted with caution, as the observed effects exhibit variability and are based on experiments conducted with a single dataset.**

***Index Terms*—Federated learning, privacy, property inference attack**

I. Introduction

Federated learning (FL) [1] has emerged as an innovative solution for training artificial intelligence models without the need to centralize data. This technique enables machine learning models to be trained from data distributed across multiple devices or servers, avoiding the need to transfer them to a single centralized server.

Although federated learning minimizes the risk of sensitive information leakage typically associated with centralized storage, it still presents notable privacy vulnerabilities [2]. One of the most significant threats is the possibility of conducting property inference attacks [3], [4], through which an attacker can attempt to deduce characteristics of clients' original data by analyzing the shared global models. These attacks can occur both passively (by observing global model updates) and actively (by manipulating the training process to improve attack effectiveness), compromising user privacy even without directly accessing their data.

This work focuses on the study of property inference attacks in the context of federated learning, analyzing how one of the key components of its architecture, the aggregation algorithm, influences both the effectiveness of the attack and the performance of the trained classifier. The main objective is to determine which aggregation algorithm offers greater privacy protection against this type of attack, while evaluating whether the choice of such an algorithm entails a trade-off in terms of the resulting model's performance. In doing so, this study seeks to contribute to a better understanding of the existing tensions between privacy and utility in federated learning systems.

The remainder of the paper is structured as follows. Section II provides the background on federated learning and describes the main aggregation algorithms analyzed in this study. Section III reviews previous research on privacy threats and property inference attacks in federated settings. Section IV details the proposed experimental setup, including the implementation of aggregation methods, attack configuration, and evaluation metrics. Section V presents and discusses the experimental results obtained under different attack scenarios. Finally, Section VI concludes the paper and outlines potential directions for future research.

II. Federated learning background

This section begins with an introduction to the federated learning paradigm, followed by a review of the main aggregation algorithms that have been proposed in the literature.

A. Federated Learning

Federated learning [2], [1] is a decentralized approach in which multiple clients collaboratively train a global model, sharing only model updates rather than raw data with a central server. Training proceeds over multiple communication rounds, where clients improve the global model by sending locally-computed parameters, while preserving the confidentiality of their datasets.

This paradigm offers distinct advantages: it improves data privacy by keeping sensitive information local, facilitates collaboration among resource-constrained or geographically distributed participants, and significantly reduces communication costs compared to traditional centralized approach. FL is increasingly adopted in domains where privacy, regulatory compliance, and efficient data integration are essential.

B. Aggregation Models

An aggregation model [5] is the algorithm used to combine the local models trained by different clients into a single global model. This process ensures privacy protection, as model parameters are aggregated instead of the raw training data.

Aggregation models in federated learning can generally be categorized into two types. In **parameter-based** aggregation, the server aggregates the trainable parameters of the models, such as the weights, shared by participating clients. In contrast, **output-based** aggregation focuses on combining the outputs produced by each client's model (such as intermediate representations or embeddings) instead of the model parameters themselves.

Below, some of the most relevant aggregation algorithms are described, all of which belong to the parameter-based category.

1) FedAvg: One of the most widely used aggregation methods is Federated Averaging (FedAvg) [6]. In this method, a group of clients is randomly selected during each training round. The central server then collects the parameters of the local models and averages them to obtain a global model. The algorithm proceeds as follows:

1) Client selection: In each communication round, the server randomly selects a group of clients to participate in the training process.
2) Local training: Each client performs training on its own model using its local data.
3) Model aggregation: Once clients have trained and sent their models, the server computes a weighted average of the local models. This average takes into account the size of each client's local dataset, so clients with more data have greater influence on the global model.
4) Global model update: The server uses the average to update the global model. This new model is then redistributed to the clients to start the next training cycle.

In the FedAvg algorithm, the global model is updated as a weighted average of the local models trained during round t:

$$w_{glob}^{t} = \sum_{k \in S_t} \frac{n_k}{n} w_k^t$$

where:

- t is the index of the federated learning round.
- w_{glob}^t is the global model after round t,
- S_t is the set of clients selected in round t,
- w_k^t is the model trained locally by client k during round t,
- n_k is the number of local samples for client k, and
- $n = \sum_{k \in S_t} n_k$ is the total number of samples from the clients participating in the round.

2) FedProx: Federated Proximal (FedProx) [7] is a variant of FedAvg developed to address the statistical heterogeneity of client data in FL. In realistic FL scenarios, client datasets are often distributed in a non-independent and non-identical (non-IID) manner, which poses challenges for stable and accurate global model convergence.

To mitigate these effects, FedProx modifies the local training objective by adding a proximal term that penalizes the difference between local model parameters and the global reference model. This regularization encourages each client's update to stay close to the shared global state, while still enabling adaptation to local data. The overall objective for client k becomes:

$$L_k(w_k) + \frac{\mu}{2} \|w_k - w_{glob}\|^2$$

where:

- $\mathcal{L}_k(w_k)$ is the local loss function of client k, which measures the error of the local model w_k with respect to the data of that client,
- w_k is the parameter vector of the local model trained by client k,
- w_{glob} is the parameter vector of the global model aggregated from all clients, and
- $\mu \geq 0$ is a parameter that controls the strength of the regularization, that is, how close the local model is required to be to the global model.

Despite this change in local training, the final aggregation at the central server remains a weighted average of the local weights, following the same formula as FedAvg. Therefore, the only difference between FedProx and FedAvg lies in the incorporation of the proximal term into the local optimization function, while the rest of the algorithm is identical in both cases.

3) FedSGD: Unlike FedAvg and its derivatives, which combine the *weights* of local models after training, Federated Stochastic Gradient Descent (FedSGD)- based aggregation [8] works directly with the *gradients* obtained during local optimization.

In this scheme, each client calculates the gradients of its loss function with respect to the model parameters and sends them to the central server. The server then updates the global model using an aggregated gradient descent step:

$$w^t = w^t - \eta \cdot \sum_{k \in S_t} \frac{n_k}{n} \nabla f_k(w^t)$$

where:

- w^t is the current global model at round t,
- η is the learning rate,
- $\nabla f_k(w^t)$ is the local gradient computed by client k, and
- n_k is the number of samples of client k,

This approach eliminates the need for full local model training at each client in every round, making the process faster and more suitable for devices with limited computational resources or strict latency requirements. However, because gradient updates do not leverage local data for multiple epochs, the method can be more susceptible to noisy updates and slows overall convergence, particularly in heterogeneous scenarios.

FedSGD is thus beneficial for quick, communication-efficient distributed optimization, but may require careful tuning and robust noise handling to achieve comparable stability and privacy as weight-based aggregation methods like FedAvg or FedProx.

III. State of art

Federated learning (FL) systems remain vulnerable to privacy attacks despite avoiding centralized data collection. Most attacks target individual data samples: membership inference attacks determine whether specific data was uses in training, while gradient inversion attacks reconstruct original data from

shared gradients [2]. Property inference attacks differ fundamentally: they do not target individual samples but instead infer general statistical properties of the entire training dataset by analyzing aggregated model updates.

A **property inference attack** [9] is a form of privacy attack in federated learning that attempts to infer characteristics of the training data that are irrelevant to the learning objective. In this type of attack, the adversary attempts to infer properties that are not directly necessary for the global model but that may reveal sensitive information about the training data. For example, in a federated learning system where the global model is trained to detect a specific disease, an attacker could attempt to infer whether a client's training data comes from a population with a high or low socioeconomic status. Although this information is not necessary for the model's main task, it could be indirectly deduced from the statistical patterns reflected in the model updates, thereby violating user privacy.

Property inference attacks can be carried out by either a passive adversary, who merely observes the training process without direct interaction, or an active adversary, who actively participates in the training procedure.

A. Passive Adversary

In a property inference attack with a passive adversary [10], the adversary does not participate directly in the model training process nor attempts to influence it. In this type of attack, the adversary only has access to the global models obtained after each training round. The adversary observes the global models generated by aggregating the local models from the different clients participating in training. By studying the global model, the adversary can infer properties of the clients' data without having direct access to them.

The phases of this type of attack are as follows. First, the attacker collects datasets with and without the property of interest (*Attack Preparation phase*) and trains shadow models on each, recording model updates during training (*Shadow Model Training phase*). These collected model updates are labeled by property and used to build a meta-classifier that learns to discriminate whether a weight update reflects presence of the property (*Meta-classifier training phase*).

When the real federated process runs, the attacker feeds differences between successive global models to the meta-classifier to detect the property's presence (*Inference During Federated Training phase*). If the attacker knows the set of participating clients per round, they can also link inferences to specific users.

Since the adversary does not interact with training and only observes public model updates, the attack is harder to detect compared to active approaches.

B. Active Adversary with Poisoning

An active adversary differs fundamentally by manipulating the training process [4], [3]. While following the same shadow model and meta-classifier construction as the passive attack, the active adversary injects poisoned updates during federated rounds to amplify the distinguishing characteristics of property P in the aggregated global model. This manipulation makes the presence or absence of P more clearly reflected in model updates, thereby increasing the inference accuracy of the meta-classifier.

IV. Methodology and Experimental Setup

This section describes the methodology followed to conduct the experiments, as well as the implementation and configuration of the different federated learning algorithms employed.

A. Aggregation algorithms

For the implementation of the federated learning methods, the TensorFlow Federated (TFF) library was used, leveraging the native functions and algorithms it provides. Specifically, the following algorithms from `tff.learning.algorithms` were evaluated:

- **FedSGD**: implemented using build_fed_sgd, which performs a global update based on the aggregated gradient.
- **Unweighted FedAvg**: implemented with build_unweighted_fed_avg, which computes a simple average of the local models.
- **Weighted FedAvg**: using build_weighted_fed_avg, which applies a weighted average according to the number of samples per client.
- **Unweighted FedProx**: implemented with build_unweighted_fed_prox, which introduces a proximity term to limit the deviation of local models from the global one.
- **Weighted FedProx**: using build_weighted_fed_prox, which combines the proximity penalty with a weighted averaging scheme.

B. Experimental Configuration

Experiments were conducted with the *LFW_deepfunneled* dataset [11], that contains facial images.

For training and evaluating the global model within the federated learning framework, the dataset was split into two parts: 90% of the samples were used to train the local classifiers of the participating clients, while the remaining 10% were reserved exclusively for testing the resulting global model. This partition ensures a fair assessment of the system's performance on data unseen during training.

The values chosen for the experimental configuration reflect a balance between computational efficiency and learning capacity. The target feature "Male" was used as the primary classification variable, while the feature "White" was selected as the property to be inferred by the attack. Both the global and local models employ a learning rate of 0.01, ensuring gradual convergence. Training was carried out over 100 federated rounds. For the attack, 10 shadow models were generated, and the inference model was trained for 1000 epochs, also with a learning rate of 0.01.

C. Evaluation Metrics

This study evaluates two distinct dimensions:

- **Federated model performance:** assessment of the classifier trained within the FL framework to determine its image classification capability.
- **Privacy:** analysis of the attack model's effectiveness in inferring a private feature. The goal is to measure to what extent the federated system exposes sensitive information to malicious clients.

To evaluate the performance of the implemented models, the following standard metrics for binary classification problems were used:

- **Case counts:**
  - *TP (True Pos.):* correctly classified positive cases.
  - *TN (True Neg.):* correctly classified negative cases.
  - *FP (False Pos.):* incorrectly classified positive cases.
  - *FN (False Neg.):* incorrectly classified negative cases.
- **Optimal threshold:** the decision threshold value that maximizes model performance in classification.
- **Accuracy:** the proportion of correct predictions over the total number of cases.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision:** the proportion of true positives among all instances predicted as positive.

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall or Sensitivity:** the proportion of true positives detected among all actual positives.

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **Specificity:** the proportion of true negatives detected among all actual negatives.

$$\text{Specificity} = \frac{TN}{TN + FP}$$

- **F1-score:** the harmonic mean between precision and recall, providing a balance between the two.

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Area Under the ROC Curve (AUC-ROC):** measures the model's ability to distinguish between positive and negative classes.
- **Matthews Correlation Coefficient (MCC):** summarizes classification quality by considering true and false positives and negatives, particularly useful for imbalanced datasets.
- **Composite metric: *Attack Effectiveness Score (AES)*** In certain cases, the existence of multiple metrics can hinder decision-making and comparisons. To address this situation, a composite metric called Attack Effectiveness Score (AES) is introduced, which provides a single, synthetic value that summarizes the attack's ability to correctly infer the target property. The AES is defined as the arithmetic mean of F1-score, AUC-ROC, and MCC.
- **Privacy Score** To evaluate the security of different aggregation methods against an attack, we define a new metric called Privacy Score, which is the inverse of the *Attack Effectiveness Score* (1 − AES).

D. Optimal Threshold Selection

To determine the optimal threshold in binary classification, one of the criteria used is *Youden's J statistic*, which is calculated as the difference between the true positive rate (TPR or sensitivity) and the false positive rate (FPR). In practical terms, for a dataset with true labels and predicted probabilities, the ROC curve is obtained using the `roc_curve` function, which provides the vectors TPR, FPR and the corresponding thresholds.

The optimal threshold is selected as the one that maximizes the difference:

$$J = \text{TPR} - \text{FPR}$$

This method allows finding a threshold point that maximizes sensitivity while keeping the false positive rate low.

In addition to this criterion, the following were also considered:

- The threshold that maximizes overall accuracy.
- The threshold that maximizes the F1-score.

These criteria offer different perspectives for selecting the best threshold according to the specific evaluation objective.

This procedure was systematically applied to all combinations of aggregation algorithms and attack configurations, ensuring that the conclusions regarding vulnerability are comparable and based on the best possible scenario for the attacker.

For better understanding and visual comparison, these results are graphically illustrated in Figures 1, 2, and 3, where the effects of selecting different thresholds can be observed for the same experiment.

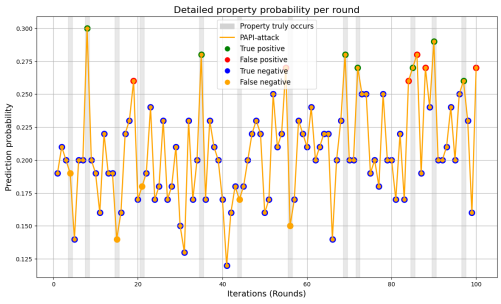


Fig. 1. Weighted FedAvg — Youden's J.

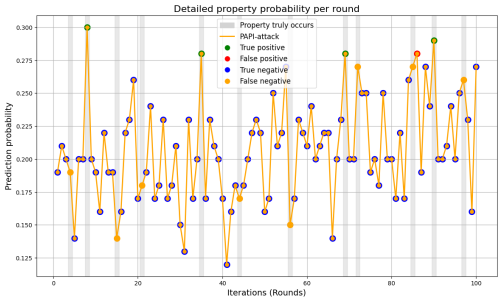


Fig. 2. Weighted FedAvg — Accuracy.

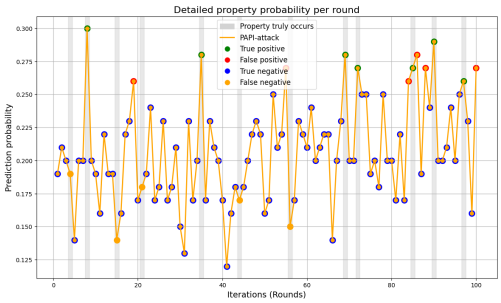


Fig. 3. Weighted FedAvg — F1-score.

V. Evaluation of Experimental Results

Tables I and II present the AES results for the different aggregation methods and threshold criteria under the active (with poisoning) and passive (without poisoning) adversaries, respectively.

Criterion	F1-score	AUC	MCC	AES
FedSGD				
Youden's J	0,727	0,954	0,716	**0,799**
Accuracy	0,400	0,954	0,476	**0,610**
F1-score	0,727	0,954	0,716	**0,799**
Unweighted FedAvg				
Youden's J	0,370	0,574	0,276	**0,406**
Accuracy	0,154	0,574	0,272	**0,333**
F1-score	0,370	0,574	0,276	**0,406**
Unweighted FedProx (proximal_strength = 0.1)				
Youden's J	0,571	0,806	0,529	**0,635**
Accuracy	0,400	0,806	0,476	**0,561**
F1-score	0,571	0,806	0,529	**0,635**
Unweighted FedProx (proximal_strength = 0.5)				
Youden's J	0,328	0,662	0,254	**0,415**
Accuracy	0,000	0,662	0,000	**0,221**
F1-score	0,328	0,662	0,254	**0,415**
Weighted FedAvg				
Youden's J	0,560	0,643	0,498	**0,567**
Accuracy	0,471	0,643	0,480	**0,531**
F1-score	0,560	0,643	0,498	**0,567**
Weighted FedProx (proximal_strength = 0.1)				
Youden's J	0,333	0,334	0,295	**0,321**
Accuracy	0,154	0,334	0,272	**0,253**
F1-score	0,333	0,334	0,295	**0,321**
Weighted FedProx (proximal_strength = 0.5)				
Youden's J	0,529	0,802	0,473	**0,601**
Accuracy	0,000	0,802	0,000	**0,267**
F1-score	0,529	0,802	0,473	**0,601**

TABLE I
CALCULATION OF THE *Attack Effectiveness Score* FOR EACH THRESHOLD CRITERION ACROSS DIFFERENT AGGREGATION METHODS WITH *poisoning*.

The obtained results reveal significant differences in the resilience of aggregation methods against property inference attacks, both in poisoned and non-poisoned scenarios.

In the **active adversary** setting (i.e. with poisoning), the most vulnerable method was FedSGD, which achieved the highest Attack Effectiveness Score (AES), with a value of 0.799. This indicates that the attacker can infer the protected feature with high accuracy. In contrast, the Weighted FedProx method with `proximal_strength = 0.1` obtained the highest Privacy Score (PS) (see Table III) with a value of 0.747, indicating substantial resistance to the attack. This suggests that an appropriate configuration of the proximal parameter can significantly enhance system privacy.

In the **passive adversary** setting (i.e. without poisoning), all methods exhibit a general reduction in attack effectiveness. However, a similar trend persists: methods based on FedProx with unweighted averaging, particularly with `proximal_strength = 0.1`, achieve the highest PS values (see Table IV), demonstrating a greater ability to preserve privacy. Conversely, methods such as Weighted FedAvg or FedSGD remain relatively vulnerable.

These observations suggest that the use of proximal regularization techniques and proper tuning of the proximity term strength can be key factors in enhancing the robustness of federated systems against inference attacks, especially in adversarial environments with malicious participants.

Criterion	F1-score	AUC	MCC	AES
FedSGD				
Youden's J	0,305	0,622	0,207	**0,378**
Accuracy	0,000	0,622	0,000	**0,207**
F1-score	0,305	0,622	0,207	**0,378**
Unweighted FedAvg				
Youden's J	0,154	0,269	0,272	**0,232**
Accuracy	0,154	0,269	0,272	**0,232**
F1-score	0,214	0,269	0,000	**0,161**
Unweighted FedProx (proximal_strength = 0.1)				
Youden's J	0,250	0,490	0,161	**0,295**
Accuracy	0,000	0,490	0,000	**0,147**
F1-score	0,250	0,490	0,161	**0,295**
Unweighted FedProx (proximal_strength = 0.5)				
Youden's J	0,316	0,702	0,223	**0,404**
Accuracy	0,000	0,702	0,000	**0,234**
F1-score	0,364	0,702	0,263	**0,443**
Weighted FedAvg				
Youden's J	0,286	0,582	0,163	**0,344**
Accuracy	0,000	0,582	0,000	**0,194**
F1-score	0,286	0,582	0,163	**0,344**
Weighted FedProx (proximal_strength = 0.1)				
Youden's J	0,444	0,650	0,425	**0,506**
Accuracy	0,444	0,650	0,425	**0,506**
F1-score	0,444	0,650	0,425	**0,506**
Weighted FedProx (proximal_strength = 0.5)				
Youden's J	0,379	0,758	0,338	**0,492**
Accuracy	0,154	0,758	0,272	**0,395**
F1-score	0,412	0,758	0,324	**0,498**

TABLE II
CALCULATION OF THE *Attack Effectiveness Score* FOR EACH THRESHOLD CRITERION ACROSS DIFFERENT AGGREGATION METHODS *without poisoning*.

Agregation method	PS
FedSGD	0,201
Unweighted FedAvg	0,594
Unweighted FedProx (proximal_strength = 0.1)	0,365
Unweighted FedProx (proximal_strength = 0.5)	0,579
Weighted FedAvg	0,469
Weighted FedProx (proximal_strength = 0.1)	**0,747**
Weighted FedProx (proximal_strength = 0.5)	0,399

TABLE III
MAXIMUM COMPLEMENTARY VALUES $1 - \text{AES}$ FOR EACH AGGREGATION METHOD WITH *poisoning*.

As shown in Table V, the final accuracy of the global model is very similar across the different aggregation methods. This indicates that, from a predictive performance perspective, all evaluated approaches are comparable and exhibit stable behavior in classifying the target feature.

These observations highlight that it is possible to improve the system's privacy without significantly compromising predictive performance.

Agregation method	PS
FedSGD	0,793
Unweighted FedAvg	0,839
Unweighted FedProx (proximal_strength = 0.1)	**0,853**
Unweighted FedProx (proximal_strength = 0.5)	0,766
Weighted FedAvg	0,806
Weighted FedProx (proximal_strength = 0.1)	0,494
Weighted FedProx (proximal_strength = 0.5)	0,606

TABLE IV
MAXIMUM COMPLEMENTARY VALUES $1 - \text{AES}$ FOR EACH AGGREGATION METHOD WITHOUT *poisoning*.

Agregation method	Precision
FedSGD	0,834
Unweighted FedAvg	0,833
Unweighted FedProx (proximal_str = 0.1)	0,835
Unweighted FedProx (proximal_str = 0.5)	0,833
Weighted FedAvg	0,833
Weighted FedProx (proximal_str = 0.1)	0,833
Weighted FedProx (proximal_str = 0.5)	0,833

TABLE V
FINAL ACCURACY OF THE GLOBAL MODEL FOR EACH AGGREGATION METHOD.

VI. CONCLUSIONS

In this paper we evaluated how different aggregation methods in federated learning (FedSGD, FedAvg, and FedProx, both weighted and unweighted) influence an attacker's ability to infer private client properties.

The results show that, although the final accuracy of the global model is almost identical across all methods (around 0.833), resistance to attack varies significantly depending on the type of aggregation and the presence or absence of poisoning. In particular, the introduction of poisoning notably increases the attacker's effectiveness.

In the poisoned scenario, the most vulnerable methods were FedSGD and Unweighted FedAvg, which achieved high AES values (up to 0.799), indicating that the attacker could infer the sensitive property with high accuracy. In the absence of poisoning, overall resistance improves, though differences between methods persist: Unweighted FedProx with `proximal_strength = 0.1` was the most robust while FedAvg and FedSGD remained relatively vulnerable.

These findings highlight that the global model's predictive accuracy does not necessarily reflect the level of privacy provided by the system.

This work presents the results of the analysis conducted on a single dataset. As an immediate direction for future work, the experimental phase should be expanded to verify that the obtained results generalize well to other scenarios and, consequently, that the conclusions can be reliably extrapolated to the broader context of federated learning.

ACKNOWLEDGMENTS

This work is funded by the projects PID2021-125962OB-C33 SECURING/NET, PID2021-125962OB-C31 SECURING/CYBER, PID2024-156914OB-C43 SAFE-BLOCKCHAIN, and PID2024-156914OB-C41 SAFE-CYBER, funded by the Ministerio de Ciencia e Innovación, la Agencia Estatal de Investigación and the European Regional Development Fund (ERDF); the ARTEMISA International Chair of Cybersecurity and the DANGER Strategic Project of Cybersecurity C062/23, both funded by the Spanish National Institute of Cybersecurity through the European Union - NextGenerationEU and the Recovery, Transformation and Resilience Plan; and the Catalan Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) grants SGR2021-00643 and SGR2021-01508.

REFERENCES

[1] J. Sahoo, M. Ouaissa, and A. K. Nair, *Federated Learning: Principles, Paradigms, and Applications*. Apple Academic Press, 2024.

[2] J. C. Zhao, S. Bagchi, S. Avestimehr, K. S. Chan, S. Chaterji, D. Dimitriadis, J. Li, N. Li, A. Nourian, and H. R. Roth, "Federated learning privacy: Attacks, defenses, applications, and policy landscape — a survey," *arXiv preprint arXiv:2405.03636*, 2024. [Online]. Available: https://arxiv.org/abs/2405.03636

[3] Z. Wang, Y. Huang, M. Song, L. Wu, F. Xue, and K. Ren, "Poisoning-assisted property inference attack against federated learning," *IEEE Transactions on Dependable and Secure Computing*, 2022.

[4] A. Serebriakova, G. P. Burgueño, V. Garcia-Font, and C. Pérez-Solà, "Unveiling privacy risks: Property inference attacks in federated learning environments," in *XVIII Reunión Española sobre Criptología y Seguridad de la Información. RECSI 2024*. Universidad de León, 2025, pp. 331–336.

[5] P. Qi, D. Chiaro, A. Guzzo, M. Ianni, G. Fortino, and F. Piccialli, "Model aggregation techniques in federated learning: A comprehensive survey," *Future Generation Computer Systems*, vol. 150, pp. 272–293, 2024.

[6] Y. Liu and J. J. Yu, "Privacy-preserving traffic flow prediction: A federated learning approach," https://www.researchgate.net/publication/344381999, 2020.

[7] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[8] S. Singh, "Fedsgd," https://shreyansh26.github.io/post/2021-12-18-federated-optimization-fedavg/, 2021.

[9] T. Wang, "Property inference attacks on neural networks using dimension reduction representations," Ph.D. dissertation, Harvard University, 2019.

[10] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning," *arXiv preprint arXiv:1812.00910*, 2020. [Online]. Available: https://arxiv.org/abs/1812.00910

[11] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," in *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, 2007.

QUILLAQUA: Arquitectura por capas y Huella Digital de Seguridad en IoT agrícola

Diego F. Bustamante V., Luis E. Sánchez, Antonio Santos-Olmo
David G. Rosado, Eduardo Fernández-Medina
GSyA Research Group, UCLM, Ciudad Real
diegofernando.bustamante@alu.uclm.es,
{luise.sanchez, david.grosado, antonio.santosolmo, eduardo.fdezmedina}@uclm.es

***Resumen*—La agricultura digital combina superficies de ataque crecientes, dispositivos de baja potencia y conectividad irregular; proteger datos, servicios y actuadores sin perder autonomía energética es crítico. Este trabajo presenta QUILLAQUA, un marco *edge-first* de ciberseguridad sostenible para invernaderos, con defensa en profundidad, comunicaciones cifradas y supervisión local que minimizan la dependencia de la nube. El aporte central es un conjunto mínimo de indicadores con criterio energético —incluida la Huella Digital de Seguridad (HDS)—, más un método de instrumentación en pasarelas y controladores y un tablero de KRIs que integra conexiones cifradas, tiempos medio de detección/recuperación, porcentaje de OTA verificadas y dependencia de servicios remotos.**

***Index Terms*—IoT agrícola; ciberseguridad; edge computing; indicadores de sostenibilidad; Huella Digital de Seguridad.**

I. Introducción

La digitalización agrícola avanza hacia sistemas ciberfísicos (CPS) apoyados en IoT e Inteligencia Artificial (IA) para optimizar agua, energía e insumos y ganar resiliencia frente a un entorno cambiante. Bajo etiquetas como Agricultura 4.0 y agricultura inteligente, proliferan soluciones que combinan sensorización, comunicaciones y analítica en agricultura de campo, urbana y de ambiente controlado, con fuerte dependencia de la automatización y del procesamiento de datos [1], [2], [3], [4], [5]. Estos CPS se despliegan sobre hardware de bajo consumo y conectividad intermitente, mientras la expansión de dispositivos y servicios conectados viene acompañada de brechas recurrentes: despliegues sin cifrado extremo a extremo ni control de acceso robusto, fuerte dependencia de la nube y escasa consideración de los riesgos específicos de los CPS agrícolas [6], [7], [8], [9]. Aunque existen marcos y arquitecturas para mejorar la eficiencia energética y la seguridad mediante diseños por niveles, cifrado ligero y procesamiento en el borde [10], [11], [12], son aún escasos los trabajos que integran de forma explícita ciberseguridad, IA y sostenibilidad, cuantificando el coste energético de los controles y conectándolo con principios de *green tech* y resiliencia climática [13], [5].

Ante este contexto, QUILLAQUA se plantea como un marco por capas, *edge-first*, para CPS agrícolas que combina ciberseguridad, inteligencia y sostenibilidad operativa [1]. El marco se apoya en la familia MARISMA para el análisis y gestión continua del riesgo [14], [15] y en buenas prácticas alineadas con el NIST Cybersecurity Framework [16]. Operativamente, prioriza comunicaciones cifradas y autenticadas, control de acceso y monitoreo en el borde, e incorpora métricas de *green security* y la Huella Digital de Seguridad (HDS) para atribuir consumo energético a los controles y vincularlo con la toma de decisiones.

II. Background

La agricultura digital opera como un ecosistema CPS–IoT multicapa donde sensores, actuadores, pasarelas y servicios cooperan para monitorizar y controlar procesos críticos del invernadero. Proteger datos, servicios y actuadores exige mitigar amenazas como manipulación de sensores, ataques de denegación de servicio (DDoS) o secuestro de comunicaciones mediante acceso robusto, cifrado en tránsito y mecanismos de supervisión como IDS/IPS [7]. La integración de IA con IoT mejora la detección de anomalías, diagnóstico y predicción agronómica, pero persisten brechas en calidad de datos, interoperabilidad y seguridad [17], lo que refuerza la importancia de enfoques *security-by-design* y de redes 5G.

Desde la perspectiva de sostenibilidad, la Ciberseguridad con Criterio Ambiental (CRGT) propone cuantificar el coste energético de ataques y defensas para alinear protección y consumo energético, fomentando controles ligeros y esquemas de autenticación eficientes [18], [13]. En redes de sensores inalámbricos (WSN) y en el Internet de las Cosas Robóticas (IoRT), esta lógica se traduce en protocolos y topologías conscientes de la energía y en un conjunto mínimo de controles —cifrado en tránsito y reposo, autenticación y control de acceso, IDS/IPS, actualización segura de firmware y copias cifradas— que reducen la exposición y prolongan la vida útil del sistema [9], [19], [20], [12]. Defensas basadas en *edge* y *fog computing*, junto con IDS federados (FELIDS), blockchain y PUF, permiten además acercar la detección de intrusiones al origen de los datos y limitar la exfiltración de información sensible.

Para gobernar el riesgo en este tipo de entornos, MARISMA [14] y su extensión MARISMA-CPS [15], apoyadas por la herramienta eMARISMA, proporcionan un marco adaptable para modelar activos, amenazas y controles en CPS, calcular el riesgo de forma automatizada y mantener trazabilidad mediante paneles en tiempo real que facilitan priorizar mitigaciones según el contexto operativo y energético.

III. Marco QUILLAQUA

QUILLAQUA [1] adopta una arquitectura por capas orientada a CPS agrícolas con hardware de bajo consumo y conectividad intermitente. El diseño integra cifrado de extremo a extremo (TLS/SSL), control de acceso basado en roles (RBAC), autenticación reforzada y monitoreo con sistemas de detección de intrusiones (IDS/IPS), y prioriza la computación

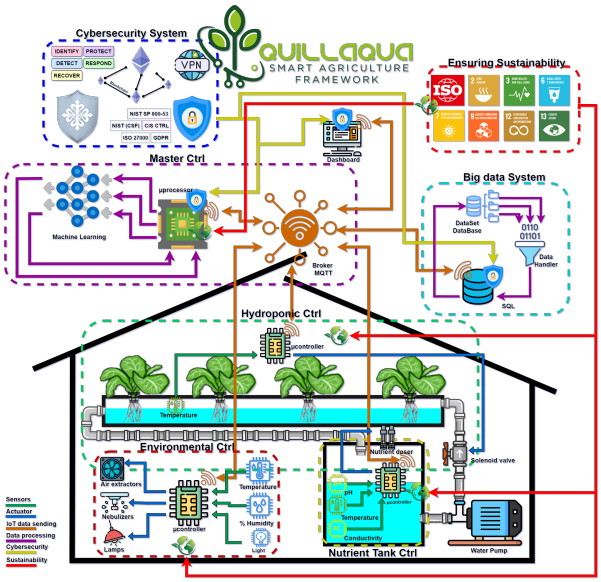


Figura 1. QUILLAQUA Framework-Arquitectura

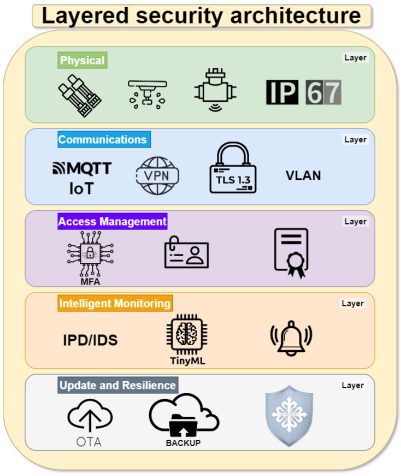


Figura 2. Layered security architecture

en el borde mediante patrones *edge/fog* y microservicios para reducir latencia y mantener datos sensibles en origen. La identidad de los dispositivos puede fortalecerse mediante Physical Unclonable Functions (PUF), y la trazabilidad ampliarse con blockchain cuando el caso de uso lo requiere. Además, QUILLAQUA incorpora la sostenibilidad como criterio técnico mediante métricas de coste energético y resiliencia [3], [21], [15], [14], [10], facilitando decisiones equilibradas entre seguridad, desempeño y consumo.

Sobre estos principios, QUILLAQUA se concreta en una arquitectura lógica organizada en cinco bloques funcionales: *Cybersecurity*, *Master Ctrl*, *Cybersecurity System*, *Ensuring Sustainability* y *Smart Greenhouse*. El bloque *Smart Greenhouse* agrupa tres sistemas embebidos principales: Environmental Control (E-Ctrl), Hydroponic Control (H-Ctrl) y Nutrient Tank Control (NT-Ctrl), responsables de la monitorización y actuación local sobre clima, solución nutritiva y depósitos. El bloque *Master Ctrl*, implementado en una Raspberry Pi 4, actúa como pasarela y orquestador: ejecuta el broker MQTT, consolida la telemetría en MariaDB y coordina flujos mediante Node-RED, con administración remota mediante SSH. El bloque *Cybersecurity* aplica los controles descritos en la Sección IV, mientras que *Cybersecurity System* y *Ensuring Sustainability* integran, respectivamente, la instrumentación de riesgos (MARISMA/eMARISMA) y los indicadores de *green security* —incluida la Huella Digital de Seguridad—. La Figura 1 resume esta organización y las interacciones entre bloques.

III-A. Versiones

La evolución avanza desde transmisiones abiertas (V1) a una arquitectura modular con seguridad en capas y gestión continua del riesgo: V2 elimina la transmisión para reducir superficie de ataque; V3 restablece conectividad e inicia gestión de riesgo; V4 consolida embebidos y refuerza controles; y V5 completa NT-Ctrl, incorpora modelos de IA con históricos y progresa hacia mayor autonomía energética, manteniendo en todas las versiones seguridad por diseño y seguimiento de indicadores que equilibran protección y consumo.

III-B. Embebidos

Los controladores organizan la operación en tres módulos: **H-Ctrl** (riego radicular), **E-Ctrl** (ventilación/extracción) y **NT-Ctrl** (dosificación y pH). La coordinación a través de *Master Ctrl* permite combinar modos manuales y automáticos, orquestar reglas y secuencias, y mantener operación local y registro coherente ante conectividad WAN variable, con orquestación del riesgo e indicadores en MARISMA/eMARISMA.

IV. Arquitectura de seguridad propuesta

La arquitectura de seguridad se organiza en capas para operar con baja potencia y conectividad limitada, alineada con QUILLAQUA y su gestión del riesgo con MARISMA/eMARISMA [15]. Cada capa aborda un tipo de problema distinto —*física*, *comunicaciones*, *gestión de accesos*, *monitoreo inteligente* y *actualización & resiliencia*— y define controles concretos. En conjunto materializan la defensa en profundidad y generan las señales que alimentan los indicadores de seguridad y energía. La Figura 2 resume esta organización.

IV-A. Capa física

coloreLa capa física reduce la exposición de los sensores y los actuadores y asegura que los nodos respeten el presupuesto energético ($< 0{,}5\,\mathrm{W}$). Considera los accesos oportunistas, la manipulación deliberada y las condiciones ambientales del invernadero dentro del ciclo de identificar–proteger–detectar–responder–recuperar. Se emplean el encapsulado del hardware, el *duty-cycling*, la comunicación frugal y el apagado de módulos no críticos para mantener la operación aun con energía limitada. En los controladores y las pasarelas se añaden el arranque verificado, el anti-*rollback*, el cierre de puertos de depuración y el cifrado de datos en tránsito (MQTT/TLS) y en reposo, con una gestión de claves integrada en QUILLAQUA. A futuro, se prevén sensores de intrusión (tapa, acelerómetro, contactos reed) que activen el bloqueo o el modo seguro y el registro forense, así como la correlación entre las alertas observadas por el IDS en el borde (cerca del sensor) y los eventos/telemetría registrados en las pasarelas o en la plataforma aguas arriba. La trazabilidad

entre el identificador físico, el certificado y la CMDB, junto con la medición del consumo asociada, alimenta los KRIs de seguridad–energía y la Huella Digital de Seguridad.

IV-B. Capa de Comunicaciones

La capa de comunicaciones garantiza confidencialidad, integridad y disponibilidad en un entorno de bajo consumo y enlaces inestables. Se emplea MQTT sobre TLS 1.2/1.3 con autenticación por certificados y QoS 2, de modo que datos y comandos viajan cifrados y con garantías de entrega, mientras que el acceso administrativo se encapsula en VPN y sesiones SSH endurecidas. Según el nivel de riesgo se activa autenticación mutua (mTLS) y se aplican ACL por tópico siguiendo mínimo privilegio. El aislamiento entre tráfico IoT y de gestión, la segmentación lógica, el resguardo de claves y los límites de sesión/cotas por identidad reducen la superficie de ataque. Los clientes utilizan reintentos controlados y *store-and-forward* ante fallos de la WAN, y un IDS analiza patrones de tráfico (reconexiones anómalas, publicaciones fuera de tópico, errores TLS), pudiendo combinarse con autenticación ligera basada en PUF. Así se mantiene el flujo cifrado, se controla quién se comunica con quién y se favorece un patrón *edge/fog* con menor dependencia de la nube.

IV-C. Capa de Gestión de Accesos

La capa de gestión de accesos organiza las identidades de personas, servicios y dispositivos, definiendo autenticación, permisos y registro de actividad. Busca aplicar mínimo privilegio y segregación de funciones para evitar que una cuenta comprometida controle todo el sistema. Para tareas administrativas y paneles críticos se contempla autenticación multifactor (MFA) y acceso remoto encapsulado (VPN más saltos SSH endurecidos); en el estado actual ya se dispone de VPN y MQTT/TLS, y segmentación lógica y MFA se incorporan como refuerzos planificados. Las comunicaciones entre servicios y dispositivos usan mTLS y ACL por rol, soportadas por una autoridad certificadora interna con raíz *offline*, certificados por identidad y distribución de CRL/OCSP en ventanas de conectividad. En nodos de campo se estudia el uso de PUF con *fuzzy extractors* para identidades resistentes a clonación sin almacenar claves directamente. Sobre el broker se aplican RBAC y ACL específicas, cuentas segregadas y políticas de rotación y revocación de credenciales. Indicadores como cobertura de MFA, porcentaje de conexiones mTLS válidas o proporción de tópicos con ACL explícita permiten seguir la salud de esta capa en eMARISMA.

IV-D. Capa de monitoreo inteligente

La capa de monitoreo inteligente detecta y contiene actividad anómala con bajo consumo y mínima exfiltración de datos. En el borde, sobre una Raspberry Pi 4, se despliegan IDS y, cuando el riesgo lo justifica, IPS para inspeccionar tráfico IoT y de administración. Estos sistemas correlacionan registros del broker MQTT y de la base de datos para identificar reconexiones inusuales, publicaciones fuera de política o consultas sospechosas, acortando el tiempo medio de detección (MTTD) y evitando depender de la nube para saber qué ocurre en el invernadero. En coherencia con Green Security, se estudian modelos ligeros (TinyML/TensorFlow Lite) para realizar parte del análisis en el borde, estimando tasas de publicación, latencia o consumo energético sin enviar datos crudos. Los indicadores incluyen tasas de detección y falsos positivos, tiempos de respuesta y energía de inferencia; las alertas, enriquecidas con información sobre activo, identidad y capa afectada, activan *playbooks* de contención y recuperación e impulsan el ciclo de mejora continua en eMARISMA.

IV-E. Capa de actualización y resiliencia

La capa de actualización y resiliencia mantiene firmware, software y configuraciones al día, asegurando continuidad aun con energía limitada y conectividad WAN intermitente. Se adopta una estrategia de actualizaciones OTA firmadas extremo a extremo: las imágenes se distribuyen por canales cifrados (TLS 1.2/1.3), se verifican criptográficamente en el dispositivo y se bloquea el retroceso mediante anti-*rollback*, organizando la distribución en oleadas y dentro de ventanas energéticas planificadas. La continuidad se refuerza con copias de seguridad cifradas, separación lógica de componentes y pruebas de restauración alineadas con objetivos RTO/RPO. En los canales de datos se emplea MQTT con QoS 2 y almacenamiento local con reenvío controlado, lo que permite operar en modo aislado y sincronizarse cuando reaparece la conectividad. Las tareas de mantenimiento se programan considerando consumo, *duty-cycling* y reposo de módulos, manteniendo siempre activos cifrado y autenticación básicos. Esta capa combina OTA firmada, respaldo verificado y administración remota segura, alimentando indicadores como % de OTA seguras, tiempo de recuperación y HDS.

V. Integración del enfoque Green Security

El enfoque *Green Security* se aplica como criterio transversal sobre la arquitectura QUILLAQUA, priorizando en el borde controles como MQTT sobre TLS 1.2/1.3 con certificados y administración encapsulada en VPN/SSH para reducir exposición y dependencia de la nube en entornos de bajo consumo [1].

La Huella Digital de Seguridad (HDS) sintetiza el coste energético de los controles y se utiliza como KRI en eMARISMA para adaptar políticas según energía disponible [13]; en zonas sin red eléctrica, los módulos de seguridad y comunicación pueden alimentarse con energías renovables manteniendo cifrado y autenticación en el borde [11]. Sobre esta base, la Sección VI formaliza los indicadores y la instrumentación necesaria.

VI. Indicadores y métricas de seguridad sostenible

El objetivo de esta sección es responder cuantitativamente a dos preguntas: (i) cuánta energía consume la ciberseguridad en QUILLAQUA y (ii) qué beneficio aporta en reducción de riesgo y continuidad de servicio. Para ello se define un conjunto compacto de indicadores que vinculan controles concretos (VPN, TLS/mTLS, IDS en el borde, OTA firmada, etc.) con consumo energético, tiempos de detección/recuperación y dependencia de la nube, a partir de la telemetría integrada en el tablero de KRIs de eMARISMA.

La propuesta combina métricas energéticas —energía diaria de seguridad, proporción de consumo dedicada a controles y Huella Digital de Seguridad (HDS)— y métricas operativas —capacidad de detección y recuperación (MTTD/MTTR),

Tabla I
COSTO-BENEFICIO ENERGÉTICO DE CONTROLES (VERSIÓN RESUMIDA)

Control	Capa	CIA	Costo HDS	Overh./Lat.	Política/Observaciones
MQTT sobre TLS 1.2/1.3	Coms.	C,I	mJ/msg	+25–60 B; +2–6 ms	*Siempre*. Suites actuales; QoS según carga.
mTLS (cert. cliente)	Coms.	C,I	$+\alpha$ mJ/HS; mJ/msg	+150–250 B (HS); +5–15 ms inicial	*Riesgo alto / admin*. PKI local; CRL/OCSP diferidos.
VPN (administración)	Gestión	C	mWh/h (sesión)	+ms salto	*Sólo tareas admin*. Plano admin. separado y auditado.
ACL en broker	Gestión	C,I	≈ 0	≈ 0	*Siempre*. `allow_anonymous=false`; roles por tópico.
IDS en borde	Monit.	I,D	mWh/h (inferencia)	+ms inspección	*Bajo energía* $\geq E_{\text{mín}}$ *y riesgo*. Modelos ligeros.
OTA firmada	Resil.	I	mJ/update	+KB/update	*Ventanas energéticas*. Anti-rollback; verificación al arranque.

Coms.=Comunicaciones; Monit.=Monitoreo; Resil.=Resiliencia; CIA=Confidencialidad, Integridad, Disponibilidad; HS=Handshake.

grado de dependencia *cloud* y porcentaje de actualizaciones OTA seguras. La Tabla I resume el compromiso energía–protección de los principales controles y orienta cuándo mantenerlos siempre activos o activarlos sólo bajo cierto riesgo o presupuesto energético.

VI-A. Consumo energético de seguridad (kWh/día).

El primer indicador responde a la pregunta "¿cuánta energía diaria dedica el invernadero a estar seguro?". Se define como la energía atribuible al *stack* de seguridad:

$$E_{\text{sec}}(d) = \int_{t_d}^{t_d+24\,\text{h}} P_{\text{sec}}(t)\,dt,$$

donde $P_{\text{sec}}(t)$ es la potencia instantánea asociada a tareas de seguridad.

En la práctica, $P_{\text{sec}}(t)$ se estima mediante mediciones de potencia en la RPi4 y en los controladores E-/H-/NT-Ctrl, junto con el registro de *duty cycles* de actividades de seguridad: handshakes TLS/mTLS, cifrado/descifrado, inferencias del IDS y verificaciones de integridad. Estas actividades se alinean con las funciones del ciclo Identify–Protect–Detect–Respond–Recover del NIST CSF [16]. Un protocolo A/B a carga constante (con y sin un control activo) permite aislar la parte de consumo atribuible a cada función (canal, cifrado, vigilancia, registro) usando telemetría existente (reconexiones, ACK, bytes cifrados, número de inferencias) [6].

Este indicador se utiliza para dimensionar la alimentación (por ejemplo, paneles solares/baterías) y para comparar el efecto energético de activar o desactivar un control: si al habilitar el IDS en el borde $E_{\text{sec}}(d)$ aumenta un 5 % pero el riesgo se reduce de forma significativa, la decisión puede considerarse aceptable; si el incremento es muy alto para un beneficio marginal, conviene revisar la arquitectura.

VI-B. Proporción de consumo por seguridad

Mientras que $E_{\text{sec}}(\text{day})$ mide la energía absoluta, la proporción de consumo por seguridad cuantifica qué parte del consumo total se dedica a ciberseguridad:

$$\%\text{Seguridad} = 100 \cdot \frac{E_{\text{cyber}}}{E_{\text{total}}}.$$

Aquí E_{total} es la energía diaria total del invernadero y E_{cyber} agrupa cuatro bloques medibles: (i) cómputo de seguridad (terminación TLS, IDS/IPS), (ii) comunicaciones seguras (handshakes, *keep-alive*, retransmisiones evitadas), (iii) gestión de identidades y claves (PKI, accesos administrativos encapsulados en VPN/SSH) y (iv) OTA firmada y cifrado en reposo. De nuevo, se utilizan protocolos A/B y normalización por carga (tasa de mensajes, tamaño medio, ciclos de control).

Este indicador permite fijar umbrales operativos, por ejemplo, "mantener %Seguridad por debajo del 15 % del consumo total". Si al incorporar un nuevo control el porcentaje se dispara, el tablero lo hace visible y facilita reconsiderar su diseño (por ejemplo, reducir frecuencia de escaneos o agrupar envíos a la nube).

VI-C. Tiempo medio de detección y respuesta

Los tiempos medio de detección (MTTD) y de respuesta (MTTR) cuantifican el valor operativo de tener IDS, VPN, registro y monitoreo local: indican cuánto tarda el sistema en darse cuenta de un problema y cuánto en recuperarse.

$$\text{MTTD} = \bar{X}(t_{\text{alerta}} - t_{\text{evento}}), \quad \text{MTTR} = \bar{X}(t_{\text{recuperación}} - t_{\text{alerta}}).$$

Las marcas de tiempo se extraen del IDS/IPS, del broker MQTT, de la base de datos y del plano de acceso. Se reportan medias y percentiles p50/p95 por clase de incidente y ventanas semanales / mensuales. El *store-and-forward* permite seguir registrando eventos y aplicar estos indicadores incluso ante caídas de la WAN, y el procesamiento en el borde reduce latencia y contribuye a disminuir el MTTD frente a soluciones centradas en la nube.

En el tablero, MTTD y MTTR se utilizan para evaluar si la combinación de VPN, IDS local y reglas de acceso está cumpliendo los objetivos de resiliencia: un aumento sostenido del MTTD desencadena revisión de firmas, umbrales o capacidades de correlación.

VI-D. Grado de dependencia cloud

Este indicador mide hasta qué punto la seguridad de QUILLAQUA depende de servicios remotos, algo clave en un marco *edge-first*:

$$\text{Dependencia}(\,\%) = 100 \times \frac{\text{operaciones } \texttt{cloud_required}}{\text{operaciones totales}}$$

donde cada evento o operación de seguridad se etiqueta con $\texttt{execution_scope} \in \{\texttt{local}, \texttt{cloud_required}\}$. Pueden definirse variantes por número de eventos, volumen de bytes o tiempo de computación.

Este indicador evidencia qué controles pueden seguir operando de forma local (por ejemplo, IDS en el borde, autenticación basada en certificados, VPN entre sedes) y cuáles requieren inevitablemente servicios externos (por ejemplo, resolución de CRL/OCSP en la nube). En el tablero se utiliza para priorizar la migración de funciones críticas hacia el borde y reducir la probabilidad de que una caída de la WAN deje al invernadero sin capacidad de protegerse.

VI-E. Actualizaciones OTA seguras

Las actualizaciones OTA seguras responden a la pregunta "¿qué porcentaje de cambios de firmware/software se realizan con garantías de integridad y autenticidad?":

$$\%\text{OTA seguras} = 100 \times \frac{N_{\text{OTA seguras}}}{N_{\text{OTA totales}}},$$

donde $N_{\text{OTA seguras}}$ contabiliza únicamente aquellas operaciones en las que se ha verificado la firma, uso del canal TLS

1.2/1.3 y se ha aplicado anti-*rollback*. La instrumentación se apoya en registros del servidor de distribución, acuses del dispositivo y evidencias del canal. La combinación de despliegue gradual (réplica/*canary*/oleadas) y administración encapsulada (VPN/SSH endurecido) reduce el riesgo asociado.

En eMARISMA, este porcentaje se interpreta como un KRI de higiene de actualización: valores cercanos al 100 % indican que el proceso está bajo control; descensos repetidos señalan costuras en la cadena de aprovisionamiento o en la operación remota que requieren medidas correctivas.

VI-F. Huella digital de seguridad (kWh/dispositivo·día)

La Huella Digital de Seguridad (HDS) sintetiza, por dispositivo y por día, cuánta energía se dedica a controles de ciberseguridad y permite comparar configuraciones de forma homogénea:

$$\mathrm{HDS} = \frac{1}{N}\sum_{i=1}^{N} \frac{E_{\mathrm{sec},i}(T)}{T_{\text{días}}} \Big[\mathrm{kWh}/(\text{dispositivo}\cdot\text{día})\Big]$$

donde N es el número de dispositivos y $E_{\mathrm{sec},i}(T)$ la energía de seguridad acumulada por el dispositivo i en el periodo $T_{\text{días}}$. Valores altos indican configuraciones más "blindadas" y costosas; valores muy bajos pueden revelar controles insuficientes.

La Figura 3 detalla el mapa de instrumentación que alimenta la HDS a lo largo del flujo CPS → broker → pasarela → base de datos → nube. A la izquierda se sitúan los CPS (E-/H-/NT-Ctrl), donde se mide el coste de los *handshakes* TLS y del cifrado por mensaje útil (`TLS HS`, mJ/evento; `Cifrado`, mJ/msg y bytes adicionales), que se traducen en el indicador `HDS_msg`. En el broker MQTT/TLS se observan mensajes extra por QoS/reintentos y eventos bloqueados por ACL (`QoS/retry`, `ACL`), de los que se derivan `Overhead_bytes` y `Lat_Δ`. Sobre el Master Control (RPi4) se despliega el IDS en el borde, cuya energía por hora, %CPU e inferencias/hora (`IDS local`) aportan a `HDS_hora`; la base de datos MariaDB introduce el coste por operación de persistencia. El plano de administración (VPN/SSH) incorpora el consumo asociado a sesiones de gestión, mientras que el bloque de nube (opcional) cuantifica la diferencia entre enviar datos crudos o agregados (`Agregados vs crudos`, B/intervalo, mJ/MB). En la parte inferior, el panel HDS resume estos KPIs en métricas operativas (`HDS_msg`, `HDS_hora`, `Overhead_bytes`, `Lat_Δ`), que se visualizan en eMARISMA para ajustar profundidad y frecuencia de inspección sin perder garantías de seguridad [13].

Para que otros despliegues de QUILLAQUA reproduzcan las mediciones sin instrumentar todo el sistema, se define una matriz mínima en la Tabla II con tres puntos de medida —broker, IDS en borde y enlace opcional a la nube—, indicando señal/unidad, fuente o método y el KPI asociado (`HDS_msg`, `HDS_hora`, `Overhead_bytes`, `Lat_Δ`). La nota de la tabla recuerda normalizar por carga (mensajes/s, tamaño medio, ciclos de control) y emplear *store-and-forward* ante intermitencia WAN para mantener comparabilidad entre laboratorio y campo.

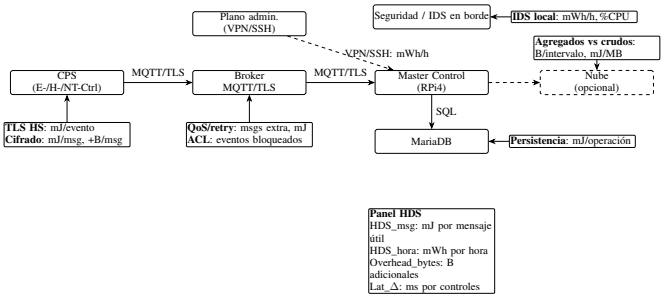


Figura 3. Mapa instrumentación HDS: puntos de medida y KPIs de salida.

Tabla II
TRAZABILIDAD DE INSTRUMENTACIÓN HDS (VERSIÓN MÍNIMA).

Punto de medida	Señal / Unidad	Fuente / Método	Vinculación KPI
CPS → Broker	Handshake [mJ/evento]; cifrado [mJ/msg]; +B/msg	Medidor de potencia en CPS; logs MQTT (bytes, QoS)	`HDS_msg`, `Overhead_bytes`, `Lat_Δ`
IDS en borde (RPi4)	mWh/h; %CPU; inferencias/h	Telemetría OS; potencia RPi4; contador de inferencias	`HDS_hora`
Enlace a nube (opcional)	B/intervalo; mJ/MB	Contadores de envío; potencia RPi4	`HDS_hora`, `Overhead_bytes`

Nota: normalizar por carga (msgs/s, tamaño medio, ciclos de control) y usar *store-and-forward* ante intermitencia WAN.

VII. GOBERNANZA Y MEJORA CONTINUA

La gobernanza de QUILLAQUA se articula mediante un ciclo PDCA (Plan–Do–Check–Act) integrado con MARISMA-CPS, de modo que el riesgo se evalúa de forma continua y los KRIs se siguen casi en tiempo real, conectando lo que ocurre en el invernadero con decisiones sobre segmentación, cifrado, monitoreo y dependencia de la nube. En *Plan* se fijan objetivos de seguridad y sostenibilidad, inventario A–T–C, umbrales y políticas base (cifrado, acceso, monitoreo, OTA segura); en *Do* se despliegan VPN para acceso remoto, MQTT/TLS, IDS en el borde y el ciclo NIST (identificar–proteger–detectar–responder–recuperar) con registro en broker y base de datos. En *Check*, eMARISMA consolida métricas de eficacia y energía (MTTD/MTTR, %TLS/SSL, %OTA seguras, HDS y proporción de consumo por seguridad) frente a los umbrales para detectar desviaciones y priorizar mitigaciones, mientras que en *Act* se ajustan segmentación, autenticación (MFA), perfiles IDS, QoS/ACL y procedimientos de OTA, recalibrando KRIs y objetivos energéticos.

La Figura 4 resume esta gobernanza: eMARISMA actúa como núcleo de riesgo continuo y tablero de KRIs, las fases PDCA se mapean al marco NIST y un anillo exterior representa *Green Security* como condición transversal —HDS, proporción de consumo por seguridad y baja dependencia de la nube— en el ciclo de mejora continua.

VIII. CONCLUSIONES

QUILLAQUA es un marco *edge-first* que integra seguridad por diseño y sostenibilidad operacional. En su estado actual mantiene defensa en profundidad con acceso remoto por VPN, mensajería MQTT sobre TLS (QoS 2) y consultas bajo TLS/SSL, con una hoja de ruta hacia segmentación lógica y autenticación multifactor sin comprometer la automatización del invernadero. La gobernanza se estructura con PDCA y riesgo continuo soportado por MARISMA-CPS/eMARISMA, cuyo tablero consolida métricas de eficacia y sostenibilidad

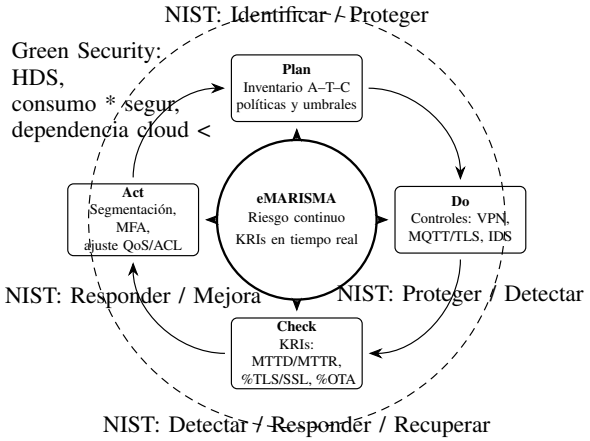


Figura 4. Gobernanza de QUILLAQUA.

(MTTD/MTTR, porcentaje de conexiones TLS/SSL, tasa de OTA firmadas) para priorizar mitigaciones.

Como aporte central se incorporan indicadores de "green security", en particular la Huella Digital de Seguridad (HDS) para atribuir consumo energético a controles de ciberseguridad. Se define un protocolo reproducible (handshakes TLS/mTLS, cifrado/descifrado, vigilancia IDS, verificaciones) y se mapea su instrumentación a lo largo del flujo CPS→broker→ pasarela→base de datos→nube, generando KPIs que permiten comparar laboratorio y campo. En comunicaciones se combinan TLS/mTLS, ACL por tópico y aislamiento del plano administrativo con VPN/SSH, junto con límites de sesión/QoS, *backoff* y *store-and-forward* para robustecer disponibilidad y contener el coste energético, ajustando parámetros en campo según cobertura/SNR y presupuesto.

Líneas de mejora incluyen: (i) desplegar el tablero de KRIs en eMARISMA cerrando el PDCA con métricas energéticas (HDS) y de eficacia (MTTD/MTTR); (ii) completar segmentación lógica y MFA; (iii) adoptar OTA firmadas con anti-*rollback* y su seguimiento como indicador; y (iv) evaluar analítica ligera en el borde (IDS/TinyML) para reducir exfiltración y latencia manteniendo trazabilidad de controles.

Agradecimientos

Esta publicación no habría sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MICIU/AEI/10.13039/501100011033 y la Unión Europea (Chist-Era Program); KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MICIU/AEI/10.13039/501100011033 /FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la JCCM, y FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; y Grant RED2024-154240-T financiado por MICIU/AEI/10.13039/501100011033.

Referencias

[1] D. BustamanteV, L. E. Sánchez, D. G. Rosado, A. Santos-Olmo, and E. Fernández-Medina, "Towards a sustainable cybersecurity framework for agriculture 4.0 based on a systematic analysis of proposals," *Computers & Security*, vol. 159, p. 104650, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404825003396

[2] A. Mitra, S. L. T. Vangipuram, A. K. Bapatla, V. K. V. V. Bathalapalli, S. P. Mohanty, E. Kougianos, and C. Ray, "Smart agriculture: A comprehensive overview," *SN Computer Science*, vol. 5, no. 8, p. 969, Oct. 2024. [Online]. Available: https://doi.org/10.1007/s42979-024-03319-w

[3] G. Singh, N. Kalra, N. Yadav, A. Sharma, and M. Saini, "Smart agriculture: A review," *Siberian Journal of Life Sciences and Agriculture*, vol. 14, no. 6, pp. 423–454, DIC. 2022. [Online]. Available: 10.12731/2658-6649-2022-14-6-423-454

[4] S. Qazi, B. A. Khawaja, and Q. U. Farooq, "Iot-equipped and ai-enabled next generation smart agriculture: A critical review, current challenges and future trends," *IEEE Access*, vol. 10, pp. 21 219–21 235, 2022. [Online]. Available: 10.1109/ACCESS.2022.3152544

[5] G. Büyüközkan and D. Uztürk, "Integrated design framework for smart agriculture: Bridging the gap between digitalization and sustainability," *Journal of Cleaner Production*, vol. 449, p. 141572, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0959652624010205

[6] N. Carvalho, T. Adão, R. Morais, A. R. Costa, and E. Peres, "Cybersecurity in precision agriculture: a short review and a practical status assessment over mysense iot-based platform," *Procedia Computer Science*, vol. 256, pp. 255–266, 2025. [Online]. Available: https://doi.org/10.1016/j.procs.2025.02.119

[7] A. R. Mahlous, "Security analysis in smart agriculture: Insights from a cyber-physical system application," *Computers, Materials & Continua*, vol. 79, no. 3, pp. 4781–4803, 2024. [Online]. Available: http://www.techscience.com/cmc/v79n3/57134

[8] S. Neethirajan, "Safeguarding digital livestock farming - a comprehensive cybersecurity roadmap for dairy and poultry industries," *Frontiers in Big Data*, vol. Volume 8 - 2025, 2025. [Online]. Available: https://www.frontiersin.org/journals/big-data/articles/10.3389/fdata.2025.1556157

[9] M. M. Moslehi, "Exploring coverage and security challenges in wireless sensor networks: A survey," *Computer Networks*, vol. 260, p. 111096, 2025. [Online]. Available: https://doi.org/10.1016/j.comnet.2025.111096

[10] J. Miao, D. Rajasekhar, S. Mishra, S. K. Nayak, and R. Yadav, "A microservice-based smart agriculture system to detect animal intrusion at the edge," *Future Internet*, vol. 16, no. 8, 2024. [Online]. Available: 10.3390/fi16080296

[11] M. T. Mohammad, H. A. Mahmood, and Q. I. Ali, "A self-powered IoT platform with security mechanisms for smart agriculture," *Ingénierie des Systèmes d'Information*, vol. 28, no. 6, pp. 1525–1532, 2023. [Online]. Available: https://doi.org/10.18280/isi.280609

[12] P. V. Rao, N. Varshney, N. Sundaram, B. Kishore, S. O. Husain, and N. Kanys, "Developing a scalable iot architecture for precision agriculture using edge computing," in *2024 International Conference on IoT, Communication and Automation Technology (ICICAT)*, 2024, pp. 976–980. [Online]. Available: 10.1109/ICICAT62666.2024.10923487

[13] I. O. Ibraheem and L. Bello-Ahmed, "Green tech, safe networks: the role of cybersecurity in combating climate change," *Discover Internet of Things*, vol. 5, no. 1, p. 58, May 2025. [Online]. Available: https://doi.org/10.1007/s43926-025-00125-5

[14] L. E. Sánchez, A. Santos-Olmo, D. G. Rosado, C. Blanco, M. A. Serrano, H. Mouratidis, and E. Fernández-Medina, "Marisma: A modern and context-aware framework for assessing and managing information cybersecurity risks," *Computer Standards & Interfaces*, vol. 92, p. 103935, 2025. [Online]. Available: https://doi.org/10.1016/j.csi.2024.103935

[15] D. G. Rosado, A. Santos-Olmo, L. E. Sánchez, M. A. Serrano, C. Blanco, H. Mouratidis, and E. Fernández-Medina, "Managing cybersecurity risks of cyber-physical systems: The marisma-cps pattern," *Computers in Industry*, vol. 142, p. 103715, 2022. [Online]. Available: https://doi.org/10.1016/j.compind.2022.103715

[16] NIST, "The NIST Cybersecurity Framework (CSF) 2.0," National Institute of Standards and Technology, Tech. Rep., feb 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf

[17] J. S. Prashanth, G. B. Krishna, A. V. K. Prasad, and P. R. Rao, "Smart farming revolution: A cutting-edge review of deep learning and IoT innovations in agriculture," *Operations Research Forum*, vol. 6, no. 1, p. 35, Mar. 2025. [Online]. Available: https://doi.org/10.1007/s43069-025-00434-z

[18] V. K. V. V. Bathalapalli, S. P. Mohanty, E. Kougianos, V. P. Yanambaka, B. K. Baniya, and B. Rout, "A puf-based approach for sustainable cybersecurity in smart agriculture," in *2021 19th OITS International Conference on Information Technology (OCIT)*, 2021, pp. 375–380. [Online]. Available: 10.1109/OCIT53463.2021.00080

[19] P. Gangwani, A. Perez-Pons, and H. Upadhyay, "Evaluating trust management frameworks for wireless sensor networks," *Sensors*,

vol. 24, no. 9, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/9/2852

[20] J. Krejčí, M. Babiuch, J. Suder, V. Krys, and Z. Bobovský, "Internet of robotic things: Current technologies, challenges, applications, and future research topics," *Sensors*, vol. 25, no. 3, 2025. [Online]. Available: https://www.mdpi.com/1424-8220/25/3/765

[21] P. B, B. Othman, G. Trivedi, N. Manan, R. S. Pawar, and D. P. Singh, "The application of the internet of things (iot) to establish a technologically advanced industry 4.0 for long-term growth and development," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2022, pp. 1927–1932.

Extendiendo X-Road con Validación de Código de Metaservicios: Una Contribución Orientada a la Seguridad

Víctor Figueroa[1,2]
[1]Grupo de Seguridad y Auditoría (GSyA)
Universidad de Castilla-La Mancha (UCLM), Ciudad Real, España
[2]Universidad Siglo 21, Argentina
romualdovictor.figueroa@alu.uclm.es

Antonio SANTOS-OLMO
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
Antonio.Santosolmo@uclm.es

Luis Enrique SÁNCHEZ
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
Luise.Sanchez@uclm.es

David G. ROSADO
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
David.GRosado@uclm.es

Eduardo FERNÁNDEZ-MEDINA
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
Eduardo.Fdezmedina@uclm.es

***Abstract*—X-Road es un framework de interoperabilidad segura ampliamente adoptado en ecosistemas de e-Gobierno para el intercambio confiable de datos entre organismos públicos. Se identificó una vulnerabilidad en su mecanismo de registro de servicios, que permitía a los usuarios, de forma inadvertida o intencionada, utilizar códigos reservados destinados a los metaservicios internos. Este artículo presenta una contribución de software que extiende X-Road con un mecanismo de validación para prevenir dicha vulnerabilidad. Implementada como parte de GAUCHO, un marco holístico de ciberseguridad para e-Gobierno, esta mejora refuerza la integridad del framework, evita conflictos funcionales y fortalece la ciberseguridad general. La solución se integra nativamente en el código fuente de X-Road, respeta sus convenciones arquitectónicas y ha sido incorporada al repositorio oficial mediante un Pull Request formal.**

***Index Terms*—Ciberseguridad, e-Gobierno, X-Road, Interoperabilidad Segura, Validación de Código, Software de Código Abierto**

I. Introducción

X-Road es una plataforma middleware de código abierto, madura y diseñada para facilitar el intercambio de datos seguro y estandarizado entre organizaciones, particularmente en entornos del sector público [1], [2], [3]. Su adopción en ecosistemas de e-Gobierno europeos y latinoamericanos ha avanzado significativamente la interoperabilidad interinstitucional, la integridad de los datos y la prestación de servicios centrados en el ciudadano [2], [4], [5]. Sin embargo, como ocurre con cualquier infraestructura ampliamente desplegada, mantener la seguridad a largo plazo requiere una evaluación continua de riesgos sobre sus mecanismos arquitectónicos y operativos.

Una vulnerabilidad crítica reside en el manejo de los identificadores de metaservicios. X-Road incluye un conjunto de operaciones internas —tales como *listClients*, *getWsdl*, y *allowedMethods*— utilizadas para soportar el descubrimiento de servicios, el descubrimiento de federaciones y otras funcionalidades centrales [6], [7]. Sin embargo, antes de la contribución propuesta, no existía un mecanismo para evitar que los usuarios registraran servicios utilizando estos identificadores reservados [8]. Esta vulnerabilidad abre la puerta a configuraciones erróneas accidentales, resoluciones de espacios de nombres en conflicto y ataques deliberados que podrían interrumpir las capacidades de descubrimiento de servicios o engañar a los consumidores de servicios [8], [9].

La motivación de esta contribución es formalizar un mecanismo de validación que asegure la integridad del registro de servicios. Nuestra solución introduce una lista configurable de códigos de metaservicios protegidos, aplicada a nivel de la interfaz de usuario, evitando así su uso indebido. Este enfoque es coherente con principios clásicos de seguridad, tales como configuraciones seguras por defecto, controles robustos de validación de datos y la aplicación estricta del principio de privilegios mínimos [10], [11]. Fue conceptualizado dentro del marco GAUCHO, un marco holístico de ciberseguridad adaptado para infraestructuras de Gobierno digital resilientes [12].

GAUCHO (Government Adaptive Unified Cybersecurity Holistic Orchestration) es un marco holístico de ciberseguridad para e-Gobierno que articula tres componentes principales: un módulo de gestión de riesgos de la información, una capa de interoperabilidad segura y un componente de gobernanza estratégica de la ciberseguridad. En este trabajo, X-Road se sitúa dentro de la capa de interoperabilidad segura, donde actúa como framework de intercambio confiable de datos sobre el que se despliegan controles ligeros y reutilizables, como el mecanismo de validación de códigos de metaservicio presentado en este artículo.

La literatura reciente refuerza la necesidad de mejoras continuas de seguridad y protecciones proactivas en sistemas

críticos. En este sentido, el mecanismo de validación de código aquí presentado fortalece proactivamente el ecosistema X-Road al eliminar un posible vector de ataque antes de que pueda ser explotado [13].

Investigaciones recientes enfatizan que la actualización y el refuerzo continuo de las medidas de seguridad son esenciales para mantener la resiliencia de cualquier infraestructura digital [14]. En línea con este principio, nuestra solución extiende la plataforma X-Road con un mecanismo de validación adicional diseñado para fortalecer proactivamente su postura de seguridad y salvaguardar la integridad del registro de servicios.

De manera similar, la importancia de implementar controles de ciberseguridad preventivos en sistemas conectados, en alusión a plataformas como X-Road, ha sido subrayada en otros dominios. Tales medidas pueden robustecer eficazmente las infraestructuras críticas contra amenazas dirigidas [15]. En el contexto del e-Gobierno, el mecanismo de validación propuesto contribuye con este objetivo al asegurar que los códigos de servicio reservados dentro de X-Road no puedan ser mal utilizados, mejorando así la confianza y la integridad operativa.

El objetivo de este trabajo no es realizar un análisis exhaustivo de amenazas ni una evaluación empírica del impacto de vulnerabilidades en plataformas de interoperabilidad, sino introducir y validar una mejora concreta a nivel de software que refuerza la integridad operativa del framework. La contribución se centra en la prevención temprana de configuraciones incorrectas o abusos potenciales mediante un control ligero, alineado con principios de seguridad preventiva y diseño seguro por defecto.

El resto de este artículo se organiza como sigue: la Sección 2 detalla el diseño del software; la Sección 3 presenta ejemplos ilustrativos; la Sección 4 discute el impacto y la reutilización; la Sección 5 describe las limitaciones y el trabajo futuro; y la Sección 6 concluye el artículo.

II. X-Road dentro del marco GAUCHO

GAUCHO es un marco de ciberseguridad holístico orientado a infraestructuras de Gobierno digital resilientes. Se estructura en tres componentes interrelacionados:

i. Un módulo de gestión de riesgos de la información, que permite analizar y priorizar amenazas sobre servicios y activos críticos.
ii. Una capa de interoperabilidad segura, responsable de garantizar que el intercambio de datos entre organismos se realice de forma confiable y verificable.
iii. Un componente de gobernanza estratégica de la ciberseguridad, que proporciona una visión global, métricas y mecanismos de toma de decisiones.

En este contexto, X-Road se ubica dentro de la capa de interoperabilidad segura como framework de intercambio de datos sobre el que se articulan servicios de e-Gobierno. La contribución presentada en este trabajo ilustra cómo controles ligeros y reutilizables pueden incorporarse en esta capa para reducir riesgos operativos, reforzar la protección de las funcionalidades internas del framework y ofrecer mejores garantías a los organismos que participan en federaciones de servicios digitales.

Asimismo, al integrarse dentro de la visión de GAUCHO, esta mejora facilita que los eventos relacionados con intentos de uso indebido de códigos reservados puedan ser tenidos en cuenta en los procesos de gestión de riesgos y en las actividades de gobernanza de la ciberseguridad, sin necesidad de modificar el diseño global de X-Road.

Desde esta perspectiva, la validación de códigos de metaservicio constituye un ejemplo de cómo controles específicos, integrados directamente en la capa de interoperabilidad, pueden contribuir a la reducción de riesgos operativos y funcionales sin alterar la arquitectura global del sistema. Este enfoque favorece una integración progresiva entre mecanismos técnicos de protección y procesos más amplios de gestión y supervisión de la ciberseguridad.

III. Descripción del Software

X-Road es una plataforma de interoperabilidad de código abierto diseñada para permitir un intercambio de datos seguro y estandarizado, particularmente en entornos de e-Gobierno [1], [2], [3]. Cada servicio se identifica mediante un código propio, algunos de los cuales están reservados para los metaservicios internos del framework (por ejemplo, listMethods o getWsdl). Estos identificadores son críticos para las operaciones del framework y no deben ser utilizados por los miembros al registrar sus servicios, ya que ello podría suplantar metaservicios, generar conflictos operativos o funcionales y, en última instancia, introducir riesgos de seguridad.

La mejora propuesta extiende la API de la interfaz de usuario del componente Proxy de X-Road al incorporar un mecanismo de validación que salvaguarda el uso de identificadores de metaservicios durante el registro de servicios (ver Figura 1). La solución consta de tres componentes coordinados que garantizan colectivamente que los códigos reservados no puedan ser reutilizados, ya sea de forma inadvertida o intencionada.

1) **ReservedServiceCodesProvider**: Clase responsable de cargar una lista configurable de identificadores reservados desde el archivo de propiedades del sistema, a saber, *common-application.yml*. La lista se gestiona a través de la clave *proxy-ui-api.reserved-service-codes*, que los administradores pueden ajustar según sea necesario [16].
2) **Integración en la capa ServiceDescriptionService**: La clase de backend responsable de registrar nuevos servicios encapsula la lógica de validación. Antes de persistir un objeto de servicio, el validador comprueba el código del servicio contra la lista de códigos reservados y lanza una excepción de validación si se encuentra una coincidencia. Integrar esta lógica dentro del ciclo de vida del servicio existente preserva la cohesión arquitectónica.
3) **Validación en ServiceService**: Para evitar la actualización de un servicio existente con un código reservado, la clase *ServiceService* —que gestiona

las actualizaciones a nivel de la entidad Service—también fue extendida. La misma lógica de validación de *ReservedServiceCodesProvider* se incorporó en el método *updateService()*. El validador también emite registros estructurados para la auditabilidad, asegurando que los intentos bloqueados se registren de forma transparente y puedan ser revisados cuando sea necesario [17].

Juntos, estos tres componentes forman un mecanismo de validación cohesivo que intercepta tanto las operaciones de registro como la actualización de servicios, asegurando que los códigos de metaservicios reservados no puedan ser reutilizados en ningún punto del ciclo de vida del servicio. La implementación se adhiere a los principios de diseño modular de X-Road y aprovecha sus mecanismos de inyección de dependencias existentes, permitiendo una integración perfecta sin alterar la lógica central. La característica es completamente comprobable, con pruebas unitarias que validan su comportamiento en diversos escenarios, incluyendo intentos de uso de códigos reservados conocidos como *listClients*, *getWsdl*, y *getOpenAPI* [18]. Además, el mecanismo está encapsulado y es opcional, permitiendo a los administradores habilitarlo o deshabilitarlo mediante simples cambios en los archivos de configuración.

La Figura 1 define los componentes principales:

- ReservedServiceCodesProvider: Mantiene la lista de códigos de servicio reservados.
- ServiceDescriptionService: Gestiona el registro de descripciones de servicio con validación integrada.
- ServiceService: Administra servicios individuales con validación contra ServiceDescription.
- Service: Entidades que contienen el atributo serviceCode a validar.
- ProxyUiApiReservedServiceCodes: Configuración de códigos de servicio reservados.

El diagrama también ilustra las relaciones clave entre ellos. La clase ReservedServiceCodesProvider carga la lista de códigos de servicio reservados desde la fuente de configuración ProxyUiApiReservedServiceCodes. Tanto ServiceDescriptionService como ServiceService utilizan este proveedor. Más específicamente, la lógica de validación dentro de estas clases llama al método isReserved de ReservedServiceCodesProvider. Esta validación del atributo serviceCode ocurre antes de registrar o actualizar una descripción de servicio (ServiceDescriptionService) y al actualizar un servicio (ServiceService). Esta interacción es fundamental para evitar que los miembros registren sus servicios utilizando códigos reservados, resolviendo así la vulnerabilidad identificada en [8] y fortaleciendo la integridad del framework X-Road.

A. Implementación Técnica

La implementación modifica el código fuente de X-Road a través de tres componentes:

1) ReservedServiceCodesProvider.
   La clase ReservedServiceCodesProvider se introdujo para cargar y exponer los códigos de servicio reservados desde la configuración del módulo Proxy-UI-API. Esta clase encapsula la lógica de validación y permite su reutilización en diferentes servicios del sistema.
2) ServiceDescriptionService.
   La lógica de validación se integró en la clase ServiceDescriptionService, que es responsable de gestionar las descripciones de los servicios. Antes de añadir una nueva descripción de servicio, el framework ahora verifica si el código de servicio proporcionado está en la lista de códigos reservados. Si se encuentra una coincidencia, se lanza una excepción de validación (ValidationFailureException), impidiendo el registro. Este enfoque asegura que la validación se aplique de manera consistente en el punto de entrada para nuevos servicios.
3) ServiceService.
   De manera similar, la validación se extendió a la clase ServiceService para cubrir los casos de actualización. En las actualizaciones de servicios existentes, el mecanismo aplica la misma validación para impedir que un servicio existente sea modificado para emplear un código reservado. Esta doble validación, tanto en la creación como en la actualización, proporciona una defensa en profundidad contra el uso indebido de identificadores.

El mecanismo de validación se ejecuta exclusivamente durante las operaciones administrativas de registro y actualización de servicios, y no forma parte del camino crítico de invocación. Por este motivo, su impacto en el rendimiento del sistema en tiempo de ejecución es acotado y no afecta a la prestación normal de los servicios.

IV. Ejemplo Ilustrativo

Para demostrar la funcionalidad y la relevancia práctica de la contribución propuesta, presentamos un escenario representativo que involucra a un administrador de servicios que intenta registrar un nuevo servicio en X-Road. El administrador, sin ser consciente de los códigos de metaservicios reservados, envía un servicio con el identificador *listMethods*, que es utilizado internamente por X-Road para consultas de metadatos [7], [18].

Tras el envío, el framework valida inmediatamente el código de servicio solicitado contra la lista definida en *proxy-ui-api.reserved-service-codes*. Al encontrar una coincidencia, la interfaz de usuario impide el registro, mostrando una advertencia localizada y emitiendo una entrada de registro de auditoría. Esto evita que una posible configuración errónea disruptiva llegue al Servidor de Seguridad subyacente.

Como ejemplo concreto, si un administrador intenta registrar un servicio cuyo `serviceCode` coincide con un código reservado, el *Proxy UI* enviaría una solicitud similar a:

```
POST /api/services
{
  "serviceCode": "getWsdl",
  "serviceVersion": "v1"
```

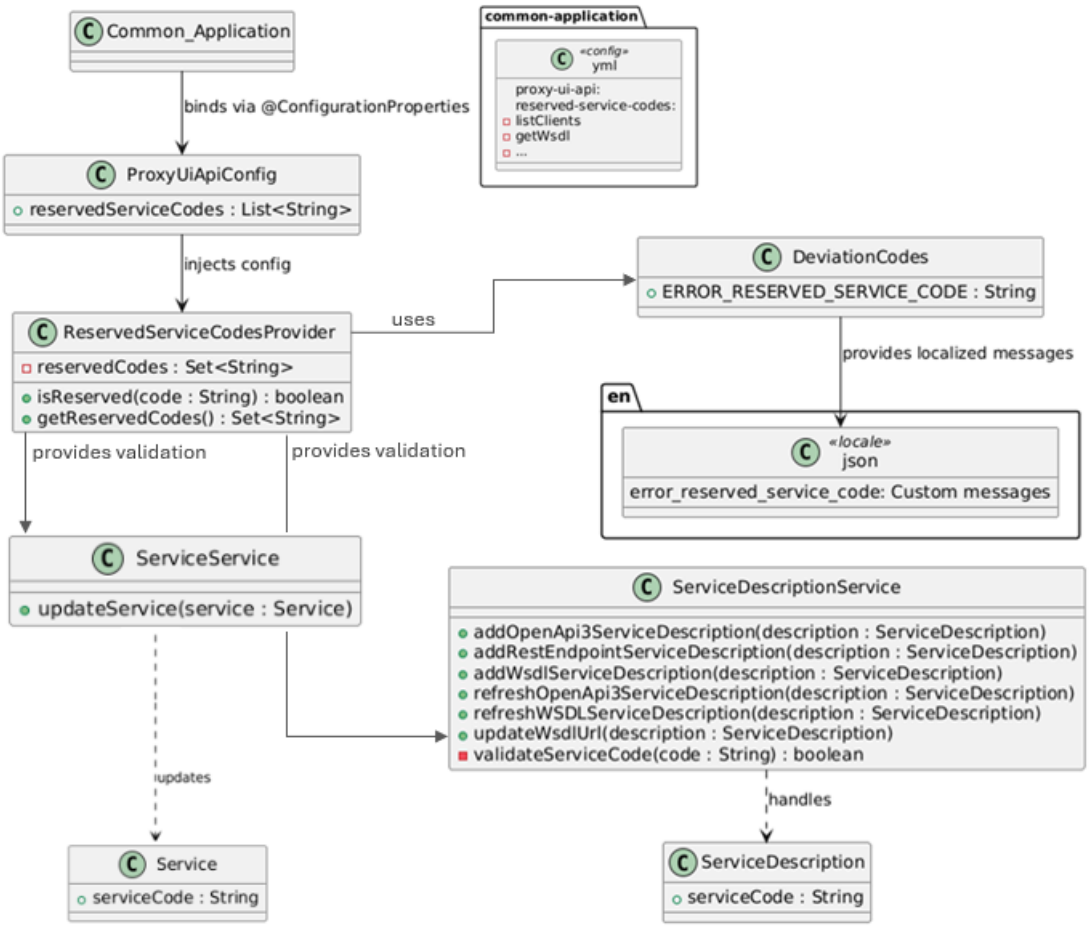


Fig. 1. Diagrama de Clases UML Simplificado de la Solución Propuesta.

```
}
```

En este caso, la interfaz devolvería una respuesta de error como:

```
HTTP 400 Bad Request
{
  "error": "ERROR_RESERVED_SERVICE_CODE",
  "message": "El código 'getWsdl' está
             reservado para metaservicios
             internos."
}
```

Si bien este tipo de configuración incorrecta no implica por sí misma una exposición directa de datos sensibles, puede inducir a comportamientos funcionales inesperados, conflictos operativos y confusión en los organismos que consumen servicios. En ecosistemas federados de interoperabilidad, estos efectos pueden erosionar la confianza en los servicios publicados y dificultar la trazabilidad de las interacciones.

En este contexto, el flujo concreto presentado complementa el escenario descrito y hace explícita la forma en que el mecanismo de validación bloquea el uso de identificadores reservados antes de que la configuración errónea llegue al Servidor de Seguridad subyacente.

La Tabla I resume tres de estos códigos de metaservicios y los riesgos asociados cuando se utilizan indebidamente:

TABLE I
EJEMPLOS DE USO INDEBIDO DE CÓDIGOS DE METASERVICIOS Y LOS RIESGOS QUE CONLLEVAN.

Cód. Reservado	Riesgo Potencial
listClients	Puede exponer relaciones internas de clientes, engañando a los miembros que consumen ese servicio.
getWsdl	Puede llevar a una generación de WSDL rota o entrar en conflicto con mecanismos de descubrimiento de metadatos.
allowedMethods	Colisiona con las convenciones de nomenclatura de los puntos finales de metadatos.

Esta medida defensiva garantiza que las funcionalidades internas permanezcan protegidas y que los usuarios finales no alteren inadvertidamente el comportamiento interno esperado del framework.

La Figura 2 ilustra la secuencia de interacciones involucradas en el proceso de validación de los códigos de servicio dentro de la plataforma X-Road. El usuario envía una solicitud de registro de servicio a través de la Interfaz de Usuario del Proxy, lo que desencadena una rutina

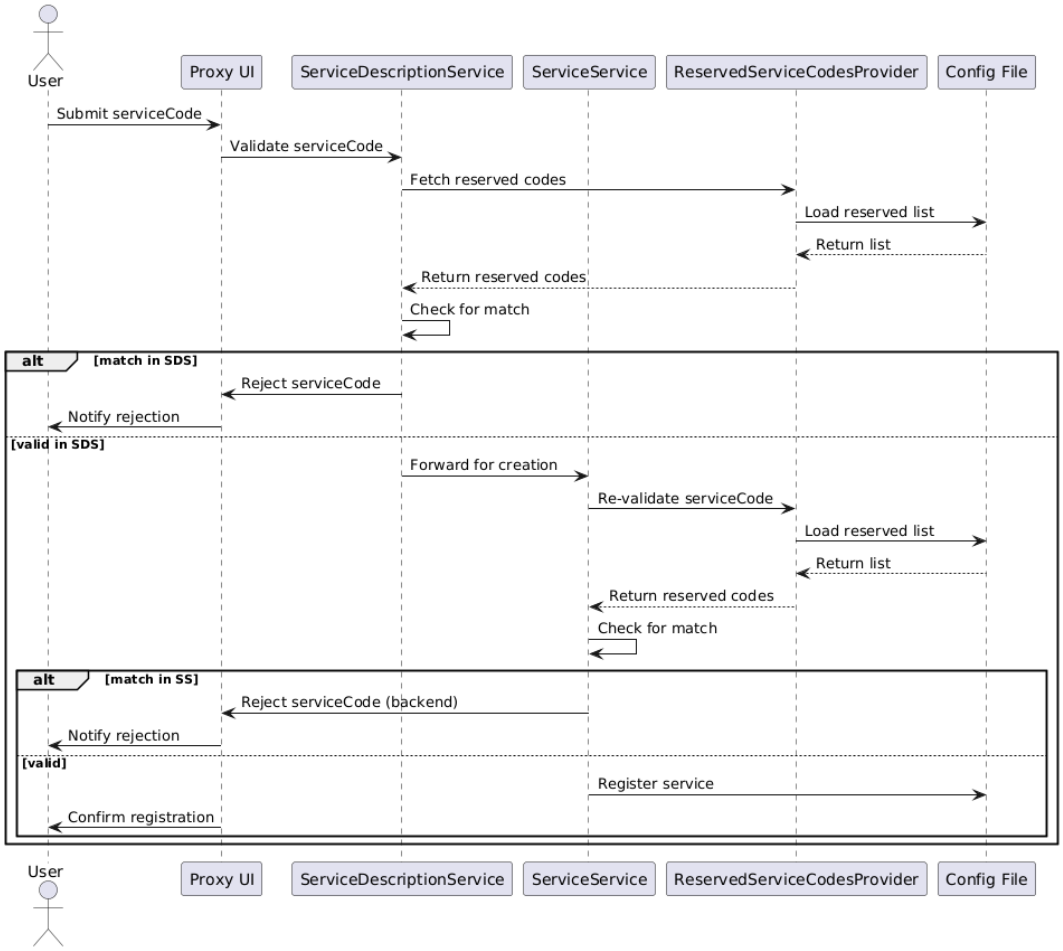


Fig. 2. Diagrama de secuencia del mecanismo de validación de código de servicio en X-Road.

de validación en ServiceDescriptionService. Este servicio consulta a ReservedServiceCodesProvider, que carga una lista de identificadores de metaservicios reservados desde el archivo de configuración. El código de servicio propuesto se verifica contra esta lista. Si coincide con un código reservado, el registro es rechazado y se proporciona retroalimentación al usuario. Si no se encuentra ninguna coincidencia, el servicio se registra con éxito. Este mecanismo asegura la integridad del espacio de nombres del servicio y previene el uso indebido de identificadores internos del framework.

V. Impacto y Potencial de Reutilización

La contribución propuesta ofrece un impacto inmediato y medible en la seguridad y robustez de las infraestructuras basadas en X-Road. Al prevenir el registro de servicios con códigos reservados, se mitiga el riesgo de conflictos de nombres, se previene la suplantación de metaservicios y se mejora la integridad general del framework. Esta mejora es particularmente relevante en ecosistemas de e-Gobierno de gran escala, donde la claridad y la previsibilidad del registro de servicios son fundamentales para garantizar la confianza y la interoperabilidad.

El potencial de reutilización de esta contribución es significativo. El mecanismo de validación, al ser modular y configurable, puede ser adaptado para otros sistemas de software que requieran la protección de identificadores reservados. Además, la contribución sirve como un caso de estudio para la aplicación de los principios del marco GAUCHO, demostrando cómo las mejoras de seguridad pueden ser integradas en sistemas de software existentes de manera no disruptiva.

VI. Limitaciones y Trabajo Futuro

La solución actual se limita a la validación en la interfaz de usuario del Proxy-UI-API, dejando fuera procesos automatizados. Además, depende de una lista estática de códigos reservados, lo que requiere actualización manual. El trabajo futuro se centrará en tres áreas principales:

1) La gestión dinámica de la lista de códigos para mejorar la usabilidad y la adaptación a amenazas emergentes [10].
2) La expansión del alcance de la validación a otros identificadores del sistema, como los de clientes o los puntos finales WSDL, para aumentar la fiabilidad de las integraciones interinstitucionales [12], [23].

3) La integración del mecanismo en la capa de gestión central de X-Road para asegurar una aplicación uniforme en todos los puntos de registro de servicios [24].

Ante situaciones excepcionales, como la imposibilidad de acceder a la configuración de códigos reservados, el mecanismo adopta un comportamiento conservador, bloqueando el registro del servicio y registrando el evento para su posterior análisis. Este diseño sigue el principio de fallo seguro y prioriza la integridad del framework. El análisis empírico del impacto de este tipo de controles en escenarios de alta concurrencia administrativa se plantea como una línea de trabajo futuro.

En conjunto, estas líneas de trabajo abren oportunidades para fortalecer la madurez operativa de X-Road dentro de ecosistemas digitales cada vez más complejos. La ampliación de controles ligeros, junto con nuevas técnicas de análisis de riesgos y capacidades de gobernanza, permitirá profundizar la integración del framework dentro de arquitecturas de ciberseguridad de amplio alcance, como GAUCHO, y avanzar progresivamente hacia infraestructuras interoperables más seguras y resilientes.

VII. Conclusiones

Este artículo ha presentado una mejora específica para el framework X-Road que aborda una vulnerabilidad previamente no mitigada en el registro de identificadores de servicios. Al implementar un mecanismo de validación para restringir el uso de códigos de metaservicios reservados, la contribución mejora tanto la robustez como la previsibilidad de las interacciones orientadas a servicios en entornos federados.

La implementación respeta la arquitectura modular de X-Road, se adhiere a los patrones de configuración e internacionalización existentes, y no introduce cambios disruptivos, lo que facilita una integración segura en implementaciones que se encuentran en producción. Su fácil configuración y su mínima sobrecarga operativa la hacen adecuada para entornos del sector público con rigurosos requisitos de control de cambios y auditoría.

Más allá de su mérito técnico inmediato, la contribución demuestra la aplicación práctica del marco holístico de ciberseguridad GAUCHO, evidenciando cómo mejoras puntuales en el software pueden mitigar riesgos latentes en ecosistemas complejos de Gobierno Digital. Asimismo, ejemplifica el potencial de incorporar controles ligeros y reutilizables en el middleware de código abierto utilizado en infraestructuras de misión crítica.

Al enviar esta característica para su inclusión en el código base principal, nuestro objetivo es compartir una solución que no solo permita mitigar una vulnerabilidad localizada, sino también promover prácticas de seguridad preventiva en entornos de interoperabilidad.

Agradecimientos

Esta publicación no hubiera sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MCIN/AEI/10.13039/501100011033 y por la Unión Europea (Programa Chist-Era); KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MCIN/AEI/10.13039/501100011033 /FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la JCCM, y FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; y la Ayuda RED2024-154240-T financiada por MICIU/AEI/10.13039/501100011033.

References

[1] R. Adeodato, S. Pournouri, Secure implementation of e-governance: A case study about Estonia, in: *Cyber Defence in the Age of AI, Smart Societies and Augmented Humanity. Advanced Sciences and Technologies for Security Applications*, Springer, (2020). DOI: https://doi.org/10.1007/978-3-030-35746-7_18

[2] G. Robles, J. Gamalielsson, B. Lundell, Setting Up Government 3.0 Solutions Based on Open Source Software: The Case of X-Road, in: *Electronic Government. EGOV 2019. Lecture Notes in Computer Science()*, vol. 11685, (2016) 111–118. DOI: https://doi.org/10.1007/9

[3] V. R. Vatsa, P. Chhaparwal, Estonia's e-governance and digital public service delivery solutions, in: *2021 Fourth International Conference on Computational Intelligence and Communication Technologies (CCICT)*, Sonepat, India, (2021), pp. 135-138. DOI: https://doi.org/10.1109/CCICT53244.2021.0003

[4] R. Saputro, I. Pappel, H. Vainsalu, S. Lips and D. Draheim, Prerequisites for the Adoption of the X-Road Interoperability and Data Exchange Framework: A Comparative Study, in: *2020 Seventh International Conference on eDemocracy & eGovernment (ICEDEG)*, pp. 216-222, Buenos Aires, Argentina, 2020. DOI: https://doi.org/10.1109/ICEDEG48599.2020.9096704

[5] H. Zhang, Z. Tang, K. Jayakar, A socio-technical analysis of China's cybersecurity policy: Towards delivering trusted e-Government services, *Telecomm. Policy*, 42(5) (2018) 409–420. DOI: https://doi.org/10.1016/j.telpol.2018.02.004

[6] NIIS, X-Road v6 Security Server Architecture (2024). https://nordic-institute.atlassian.net/wiki/spaces/XRDKB/pages/4915201/X-Road+v6+Security+Server+Architecture [Accessed: Nov. 19, 2025].

[7] NIIS, Metaservices - X-Road Knowledge Base (2024). https://nordic-institute.atlassian.net/wiki/spaces/XRDKB/pages/4915759/Metaservices [Accessed: Nov. 19, 2025].

[8] NIIS, XRDDEV-599: Prevent use of meta service codes in user-defined services, *X-Road Issue Tracker* (2025). https://nordic-institute.atlassian.net/browse/XRDDEV-599 [Accessed: Nov. 19, 2025].

[9] C. Onwubiko, K. Ouazzane, Multidimensional cybersecurity framework for strategic foresight, *Int. J. Cyber Situational Aware.*, 6(1) (2022) 46–77. DOI: https://doi.org/10.22619/IJCSA.2021.100137

[10] A. Alrubaiq, T. Alharbi, Developing a Cybersecurity Framework for e-Government Project in the Kingdom of Saudi Arabia, *J. Cybersecurity Privacy*, 1(2) (2021) 302–318. DOI: https://doi.org/10.3390/jcp1020017

[11] E. Agawu, What's Next for E-Government: Innovations in E-Government Through a Cybersecurity Lens Symposium, *Nebraska Law Rev.*, 96(2) (2017) 364–383.

[12] V. Figueroa, L. E. Sánchez, A. Santos-Olmo, D. G. Rosado, E. Fernández-Medina, Building a holistic cybersecurity framework for e-Government based on a systematic analysis of proposals, *Int. J. Inf. Secur.*, 24(121) (2025) 1–8. DOI: https://doi.org/10.1007/s10207-025-01024-0

[13] R. Wamusi et al., A Comprehensive Review on Cryptographic Techniques for Securing Internet of Medical Things: A State-of-the-Art, Applications, Security Attacks, Mitigation Measures, and Future Research Direction, *Mesopotamian Journal of Artificial Intelligence in Healthcare*, (2024) 135–169. DOI: https://doi.org/10.58496/MJAIH/2024/016

[14] S. Sarsam, H. Al-Saedi, Cybersecurity Challenges in Autonomous Vehicles: Threats, Vulnerabilities, and Mitigation Strategies, *Shifra Journal of Applied Sciences*, 2(3) (2023) 45–59. DOI: https://doi.org/10.70470/SHIFRA/2023/005

[15] M. Aljanabi, Safeguarding Connected Health: Leveraging Trustworthy AI Techniques to Harden Intrusion Detection Systems Against Data Poisoning Threats in IoMT Environments, *Babylonian Journal of Internet of Things*, (2023) 31–37. DOI: https://doi.org/10.58496/BJIoT/2023/005

[16] NIIS, X-Road: System Parameters, *X-Road Technical Documentation* (2024). https://github.com/nordic-institute/X-Road/blob/develop/doc/Manuals/ug-syspar_x-road_v6_system_parameters.md [Accessed: Nov. 19, 2025].

[17] M. Alqahtani, R. Braun, Examining the Impact of Technical Controls, Accountability and Monitoring towards Cybersecurity Compliance in e-Government organizations, *Gov. Inf. Q.*, (2021). DOI: https://doi.org/10.21203/rs.3.rs-196216/v1

[18] NIIS, X-Road: Service Metadata Protocol for REST (2023). https://docs.x-road.global/Protocols/pr-mrest_x-road_service_metadata_protocol_for_rest.html [Accessed: Nov. 19, 2025].

[19] A. Bhardwaj, P. K. Dutta, P. Chintale, Securing Container Images through Automated Vulnerability Detection in Shift-Left CI/CD Pipelines, in Babylonian Journal of Networking, (2024) 162–170. DOI: https://doi.org/10.58496/BJN/2024/016

[20] A. Stone, The Role Risk Management Plays in Reducing Cybersecurity Risks, in: *University of Maryland University College*, (2021), 28864121.

[21] C. Lino, Cybersecurity in the Federal Government: Failing to Maintain a Secure Cyber Infrastructure, in *Bulletin of the Association for Information Science and Technology*, 41(1), (2014), 24-28. DOI: https://doi.org/10.1002/bult.2014.1720410111

[22] M. I. Manda, J. Backhouse, Addressing trust, security and privacy concerns in e-government integration, interoperability and information sharing through policy: a case of South Africa, *CONF-IRM 2016 Proceedings*, Paper 67 (2016).

[23] L. E. Sánchez, A. Santos-Olmo, D. G. Rosado, C. Blanco, M. A. Serrano, H. Mouratidis, E. Fernández-Medina, MARISMA: A modern and context-aware framework for assessing and managing information cybersecurity risks, *Comput. Stand. Interfaces*, 92 (2025) 103935. DOI: https://doi.org/10.1016/j.csi.2024.103935

[24] W. Hashim, N.Hussein, Securing Cloud Computing Environments: An Analysis of Multi-Tenancy Vulnerabilities and Countermeasures, *SHIFRA*, (2024) 9–17.

Estudio comparativo del rendimiento de los estándares de firma digital post-cuánticos

Pablo Aguado Gómez
Universidad Carlos III de Madrid
Av. de Gregorio Peces-Barba Martínez, 22,
28270 Colmenarejo, Madrid
pabloag.contacto@gmail.com

Luis Hernández Álvarez
Instituto de Tecnologías Físicas y de la Información
Consejo Superior de Investigaciones Científicas
C/ Serrano 144, 28006, Madrid
luis.hernandez@csic.es

***Resumen*—En los últimos años, ha habido importantes avances en el campo de la computación cuántica. Es ya sabido que este tipo de computación representa una amenaza para la infraestructura criptográfica utilizada actualmente que garantiza la seguridad de la información y de las comunicaciones. En consecuencia, en 2016, el National Institute of Standards and Technology inició una convocatoria para elegir nuevos estándares criptográficos resistentes a ataques con computación cuántica. El objetivo de este estudio es evaluar las implementaciones de la biblioteca Botan de tres esquemas de firma considerados como estándares (ML-DSA, XMSS y SLH-DSA), por lo que se podrá medir el rendimiento de cada uno de ellos en igualdad de condiciones, utilizando los diferentes conjuntos de parámetros disponibles para cada esquema. Finalmente, se presentan todos los resultados obtenidos y se realiza un estudio comparativo del rendimiento de los tres esquemas de firma. La principal conclusión que se extrae de este estudio es que ML-DSA parece ser el esquema con mejor equilibrio entre tiempos de ejecución y tamaños de clave, seguido de cerca por SLH-DSA en algunos de sus conjuntos de parámetros, el cual ofrece tamaños de claves notablemente inferiores al resto. Las implementaciones de las pruebas y los resultados se pueden obtener en el repositorio https://github.com/100471943/TFG---Comparacion-algoritmos-de-firma-PQ**

***Index Terms*—Firma digital, algoritmo post-cuántico, rendimiento, seguridad, computación cuántica**

I. Introducción

Es un hecho que la computación cuántica plantea una amenaza significativa para la infraestructura criptográfica que utilizamos actualmente. Una de las pruebas más claras de ello son los algoritmos de Shor [1] y Grover [2]. El primero de ellos prueba que, gracias a algunas de las características intrínsecas de la computación cuántica, se podrían resolver dos problemas matemáticos que actualmente se consideran computacionalmente difíciles. Estos son la factorización de números enteros y el logaritmo discreto, ambos utilizados actualmente como base de la seguridad de mecanismos asimétricos tan importantes como RSA o ECDSA. Por su parte, el algoritmo de Grover es un algoritmo de búsqueda que acelera de manera significativa los ataques de fuerza bruta contra algoritmos de criptografía simétrica, debilitando considerablemente su seguridad.

Aún no se sabe con certeza cuánto tiempo falta hasta que se logre construir un computador cuántico capaz de ejecutar estos algoritmos y que sea lo suficientemente potente como para romper o poner en peligro los sistemas criptográficos que consideramos actualmente seguros. Pero, aunque aún no exista tal computador, es importante estar preparados antes de que esto ocurra. Es por ello que el NIST (*National Institute for Standards and Technology*) inició una convocatoria pública [3] para estandarizar algoritmos criptográficos que sean resistentes a ataques con ordenadores cuánticos. Dentro de este proceso del NIST, se desarrollaron varias rondas. En cada una de ellas se han ido analizando y descartando propuestas. En la tercera ronda fue cuando se seleccionaron cuatro algoritmos: CRYSTALS-Kyber [4] para el encapsulado de claves y CRYSTALS-Dilithium [5], FALCON [6] y SPHINCS+ [7] para firmas digitales. Además, en 2025 se ha seleccionado como estándar otro algoritmo para el encapsulado de claves: HQC [8].

Para este trabajo se han escogido las implementaciones de la biblioteca Botan de tres esquemas de firma digital post-cuánticos: ML-DSA (*Module-Lattice-Based Digital Signature Algorithm*), XMSS (*eXtended Merkle Signature Scheme*) y SLH-DSA (*Stateless Hash-Based Digital Signature Algorithm*). El primero de ellos, ML-DSA [9], es el estándar derivado de CRYSTALS-Dilithium, que fue publicado como estándar en 2024. Se trata de un esquema de firma basado en retículos. En el caso de XMSS, es un esquema de firma que fue estandarizado inicialmente por la IETF (*Internet Engineering Task Force*) en el RFC 8391 [10] en el año 2018 y, posteriormente fue evaluado y aprobado por el NIST [11]. Por último, SLH-DSA [12] es una versión estandarizada de SPHINCS+, también publicado por el NIST, en 2024. Al igual que XMSS, SLH-DSA es un esquema de firma basado en *hash*. En concreto, SLH-DSA hace uso de una versión extendida de XMSS para su propio funcionamiento. Por razones de espacio, supondremos que el lector está familiarizado con los tres esquemas de firmas considerados, por lo que no nos detendremos en detallar cada uno de los algoritmos de cada estándar.

Debe hacerse notar que en este estudio no se ha considerado FALCON, a pesar de ser un esquema de firma digital estándar, porque no ha sido publicado aún como un FIPS por el NIST. Por esta razón, es probable que su versión estandarizada difiera de la propuesta original y esta comparativa no se realizaría en igualdad de condiciones.

El fin de este trabajo es evaluar las implementaciones de la librería Botan de estos esquemas de firma bajo las mismas condiciones y realizar mediciones para evaluar su rendimiento. Una vez obtenidos los resultados, se podrá hacer una comparación del rendimiento de cada esquema de firma. Para ello, el primer paso es obtener dichas implementaciones, e incluir mediciones de los tres principales algoritmos en un esquema de firma: generación de claves, firma de un

mensaje y verificación de firmas. Se han medido los tiempos de ejecución de los tres procesos, considerando los diferentes tamaños de claves y firmas de cada esquema. Con estos parámetros se puede evaluar el rendimiento de cada algoritmo en diferentes situaciones.

El resto del documento se ha organizado la siguiente manera: la Seccion II describe la implementación de los tres esquemas de firma. A continuación, en la Sección III se muestran los resultados de cada esquema por separado para hacer un primer análisis del rendimiento de los principales procesos de cada algoritmo. Posteriormente, en la Sección IV se presentan las comparaciones directas entre el rendimiento de los tres algoritmos de cada esquema, para que, por último, se pueda hacer un análisis de estas comparaciones. En la Sección V se discuten las conclusiones acerca de qué esquema de firma puede ser más apto en diferentes situaciones.

II. Implementación

Se desarrollaron tres programas para la implementación de cada esquema de firma, en los cuales se generan las claves, se firma un mismo mensaje utilizando su respectivo algoritmo de firma, y posteriormente se verifica la firma con el algoritmo correspondiente. Además, para poder comparar correctamente el rendimiento, se han implementado mediciones durante todo este proceso. En concreto, el objetivo es medir el rendimiento de cada algoritmo de los tres esquemas de firma: la generación de claves, la firma de un mensaje y la verificación de una firma. Para ello, se miden los tiempos de ejecución y ciclos CPU de cada una de estas tres fases, además de medir los tamaños de las claves públicas y privadas, y de las propias firmas que genera cada esquema. Con estas mediciones se puede hacer un análisis del rendimiento de cada esquema de una manera fiable. Por otra parte, es importante que estas pruebas se ejecuten en igualdad de condiciones; es decir, que cada uno de los tres *scripts* sea lo más similar posible.

El equipo utilizado para la ejecución de todas las pruebas ha sido un ordenador de sobremesa. El procesador es un i5-11400f, con 32 GB de RAM DDR4, un NVME M.2 PCIE 4.0, y el sistema operativo utilizado ha sido Ubuntu 24.04, con una instalación local común en el propio NVME.

Para realizar la implementación completa de todas las pruebas, se ha elegido C++ como lenguaje de programación, principalmente porque permite medir con exactitud tiempos de ejecución, ciclos CPU y memoria ocupada por cada variable, es decir, medir tamaños de claves y firmas. También ofrece un buen balance entre eficiencia y accesibilidad. Por otro lado, ofrece varias librerías criptográficas para poder implementar cada uno de los tres esquemas de firma elegidos.

De hecho, se ha elegido la librería criptográfica Botan [13]. Botan es una librería criptográfica en C++ de código abierto, licenciada bajo *Simplified BSD License*. Esta librería ofrece una API para implementar cada uno de los tres esquemas de firma que se han elegido, además de poder elegir el conjunto de parámetros que se quiera utilizar.

Para automatizar todas las pruebas que se deben hacer a cada esquema, se ha creado un *script* de Python. Este *script* ejecuta todas las pruebas posibles de cada uno de los tres programas de C++ que implementan cada esquema de firma. Es decir, para cada esquema de firma, hemos llevado a cabo una ejecución con cada uno de los conjuntos de parámetros que tiene disponibles en la librería Botan, además de hacer la ejecución utilizando o no pre-hash, cuando el esquema de firma lo permita. Tras cada ejecución, se guardan todos los resultados obtenidos de las diferentes mediciones en un fichero *csv*, esto es, se crea un fichero *csv* por cada esquema de firma.

Al haber obtenido mediciones con todos los conjuntos de parámetros diferentes, se puede hacer una comparación más precisa entre los esquemas de firma, ajustando su nivel de seguridad en cada comparación.

Todos los *scripts* que se mencionan, junto con los tres ficheros *csv*, se pueden encontrar en el repositorio público: https://github.com/100471943/TFG---Comparacion-algoritmos-de-firma-PQ. El *script* de Python permite elegir entre ejecutar las pruebas de un esquema de firma específico o ejecutarlas todas a la vez. La estructura de los tres *scripts* de C++ permite una fácil modificación de su funcionamiento, por si se quieren ejecutar pruebas diferentes; por ejemplo, firmando mensajes distintos, o que varíen en su longitud.

En el propio repositorio de *GitHub* se incluyen instrucciones para la compilación de los *scripts* de C++ y para la ejecución de todas las pruebas.

III. Resultados obtenidos de cada esquema de firma

En esta sección se presentan los resultados más relevantes obtenidos de cada esquema de firma de manera individual. Para ello, y con el fin de lograr una mejor visualización de los resultados, se hace uso de gráficas (no se incluyen tablas por razones de espacio). Así, con el objetivo de sintetizar toda la información de la mejor manera posible, se muestran dos gráficas para cada esquema de firma: una gráfica con los tiempos de ejecución de los tres algoritmos principales y otra gráfica presentando los tamaños de claves y firmas. En cada una de estas gráficas se comparan también los diferentes conjuntos de parámetros que están disponibles en cada esquema.

Cabe destacar que los resultados que se presentan a continuación pertenecen a las pruebas mencionadas anteriormente, en las cuales siempre se firma el mismo mensaje, con la misma longitud de 4 *bytes*. Además, cada prueba se ha ejecutado cinco veces, y el resultado final que se utiliza es la media de esas cinco ejecuciones; de esta manera, se obtiene un resultado más representativo.

III-A. Resultados para ML-DSA

En el caso del esquema de firma ML-DSA, únicamente tiene disponibles tres conjuntos de parámetros: ML-DSA-4x4, ML-DSA-6x5 y ML-DSA-8x7. Cada uno de ellos ofrece una categoría de seguridad diferente de las consideradas por el NIST, en concreto, las categorías 2, 3 y 5, respectivamente, como se puede ver en su especificación oficial FIPS 204 [9]. La diferencia principal entre cada conjunto es el tamaño de la matriz A, que forma parte de la clave pública del esquema y aparece en el en el propio nombre del conjunto: 4x4, 6x5 y 8x7. También varían otros parámetros internos de los diferentes algoritmos que componen el esquema.

La Figura 1 muestra los tiempos de ejecución medidos en las pruebas realizadas sobre el esquema de firma ML-DSA, implementado en C++ con la librería Botan.

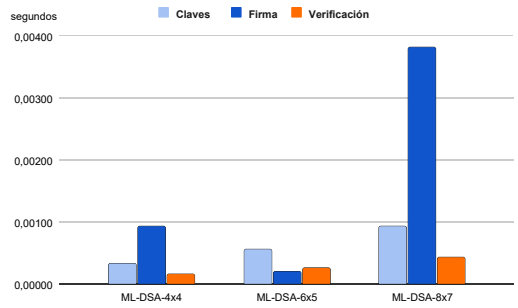


Figura 1. Tiempos de ejecución de los tres algoritmos de ML-DSA.

Como se puede ver en la Figura 1, en general, todos los tiempos de ejecución de ML-DSA son bajos. Por otra parte, se puede ver de una manera clara como varían los tiempos de ejecución de los tres algoritmos con los diferentes conjuntos de parámetros. En primer lugar, se puede observar que tanto la generación de claves como la verificación de una firma siguen un patrón lógico, es decir, a medida que aumenta el nivel de seguridad, aumentan los tiempos de ejecución, lo que se debe al incremento de los elementos considerados en los diferentes procesos. Como es de esperar, los tiempos de computación crecen a medida que los parámetros se hacen más robustos, y por lo tanto los tiempos de ejecución de cada algoritmo aumentan ligeramente.

En cuanto al proceso de generación de una firma, se puede observar que los tres conjuntos de parámetros no siguen la misma tendencia. En el conjunto de parámetros de categoría de seguridad media, ML-DSA-6x5, se puede ver una disminución en el tiempo de firma de un 78 % respecto del de categoría baja. Esto se debe a que el parámetro w dentro de este conjunto se reduce ligeramente respecto al conjunto de parámetros ML-DSA-4x4: pasa de 80 a 55. El parámetro w indica el tamaño de un polinomio auxiliar h que se utiliza en el algoritmo de elaboración de firmas en ML-DSA. Este algoritmo utiliza un proceso iterativo en el que se genera un nuevo polinomio auxiliar h hasta que se obtenga una firma válida. Por lo tanto, si dicho polinomio h se reduce en tamaño con respecto al conjunto de parámetros anterior, el tiempo de ejecución de este proceso iterativo también se reducirá y, en consecuencia, el tiempo de ejecución total del algoritmo de elaboración de firma también se reduce. En el conjunto de parámetros ML-DSA-8x7, el parámetro w vuelve a aumentar su tamaño y, por lo tanto, el tiempo para generar una firma vuelve a aumentar.

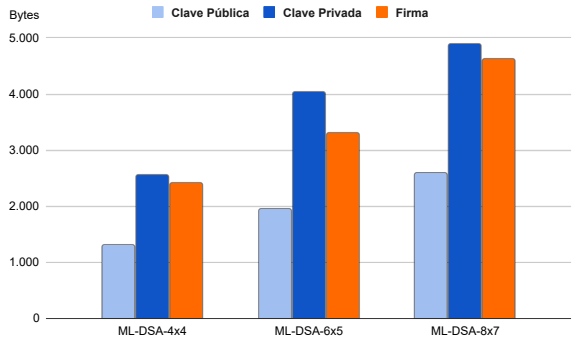


Figura 2. Uso de memoria en ML-DSA.

En cuanto a los tamaños de las claves y firmas, se puede ver en la Figura 2 que, en general, no son demasiado grandes, estando todos por debajo de los 5000 bytes. También se aprecia que los tamaños de las claves y de las firmas están relativamente equilibrados. Por último, de nuevo se sigue la misma tendencia que se observa en los tiempos de ejecución, es decir, a medida que aumenta la categoría de seguridad, en cuyo caso aumenta el tamaño de la matriz A, aumentan los tamaños de las claves y de las firmas.

III-B. Resultados para XMSS

El esquema XMSS tiene más conjuntos de parámetros disponibles. En concreto, y al margen de las familias de *hash*, los conjuntos de parámetros de XMSS se dividen en dos grupos, según el valor del parámetro n, que representa, entre otras cosas, el tamaño de cada nodo del árbol de Merkle. El primer grupo tiene un valor de n de 256 bits y, el segundo, de 512 bits. Dentro de cada uno de estos dos grupos se encuentran tres conjuntos de parámetros distintos, diferenciados por el parámetro h, es decir, la altura del árbol, la cual varía entre 10, 16 y 20 niveles. Por otro lado, el esquema XMSS ofrece la posibilidad de utilizar tanto SHA-2 como SHAKE para realizar los diferentes *hashes*. Los resultados que se ven a continuación se han obtenido utilizando SHA-2, al igual que los del resto del estudio, con el objetivo de que la comparación entre esquemas sea lo más similar posible.

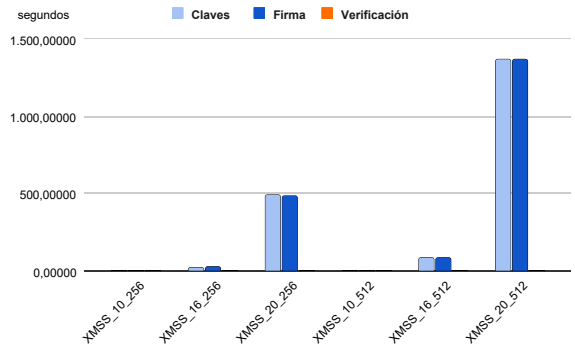


Figura 3. Tiempos de ejecución de los tres algoritmos de XMSS.

En cuanto a los tiempos de ejecución de XMSS, en la Figura 3 se puede ver que son considerablemente superiores a los de ML-DSA y a los que se van a ver posteriormente en SLH-DSA. Esto es principalmente debido a que Botan no implementa en XMSS algunas de las optimizaciones que se introducen en la especificación de SLH-DSA y, por lo tanto, ML-DSA e incluso SLH-DSA (que incluye en su ejecución una variante multi-árbol de XMSS) van a tener mejores tiempos de ejecución en este caso. Esta consideración es de gran relevancia y merece la pena tenerla en cuenta pues, en trabajos futuros, sería deseable incluir tales optimizaciones.

Por otra parte, en la Figura 3 se pueden diferenciar los dos grupos principales separados por el parámetro n, es decir, los conjuntos con una n de 256 bits o 512 bits. Dentro de cada conjunto se repite la misma tendencia: a medida que aumenta el parámetro h, es decir, la altura del árbol, aumenta el tiempo de ejecución. Este hecho no es sorprendente porque altura del árbol es el parámetro más determinante en cuanto a la

complejidad computacional del algoritmo. En concreto, para la generación de claves es necesario calcular todos los nodos del árbol de Merkle hasta llegar al nodo raíz y, por lo tanto, si la altura del árbol crece, el número total de nodos va a crecer exponencialmente. En consecuencia, al haber más nodos que computar hasta llegar a la raíz, el tiempo para generar las claves va a aumentar. Algo similar sucede a la hora de firmar un mensaje. Para firmar un mensaje con XMSS es necesario generar un camino de autenticación, es decir, una cadena de nodos del árbol, uno por nivel, que permita al receptor llegar desde la base del árbol hasta el nodo raíz. Este camino de autenticación se crea al elaborar una firma. De nuevo, si la altura del árbol aumenta, el camino de autenticación aumenta y, en consecuencia, el tiempo para generar una firma aumenta.

Dicha tendencia no se cumple para la verificación de una firma. De hecho, este proceso es rápido en todos los conjuntos de parámetros, lo cual se debe a que en XMSS, este camino de autenticación se almacena como parte de la firma. Por lo tanto, cuando el receptor quiere verificar una firma, realmente no tiene que generar cada uno de los nodos (lo cual consume gran parte del tiempo de ejecución), sino que ya los tiene generados en la propia firma, y por lo tanto se acelera el proceso.

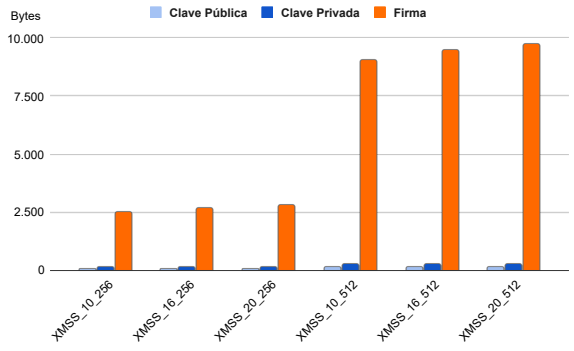


Figura 4. Tamaño de claves en XMSS.

Sobre los tamaños de claves y firmas en XMSS, se puede ver, en la Figura 4, que se sigue la misma tendencia que en los tiempos de ejecución. Además, en esta figura se aprecia de manera clara lo que se acaba de comentar sobre el camino de autenticación. Se puede ver que para todos los conjuntos de parámetros, el tamaño de las firmas es notablemente superior al de las claves, debido a que se está almacenando una serie de nodos del árbol en la propia firma.

III-C. Resultados para SLH-DSA

Por último, en cuanto a los conjuntos de parámetros de SLH-DSA, hay disponibles dos tipos de conjuntos de parámetros principales diferentes: los de tipo f y los de tipo s. Los conjuntos de parámetros de tipo f (proveniente de *fast*) optimizan todos sus parámetros para que los tres algoritmos sean más rápidos. Por otro lado, el otro tipo de conjuntos de parámetros son los de tipo s (proveniente de la palabra *small*) los cuales optimizan sus parámetros para generar firmas más pequeñas en comparación con los conjuntos de tipo f.

En este caso, se presentan los conjuntos que utilizan *hashes* SHA2. También hay disponibles conjuntos con *hashes* SHAKE. Éstos últimos son algo más lentos que los SHA2, pero ambos siguen las mismas tendencias.

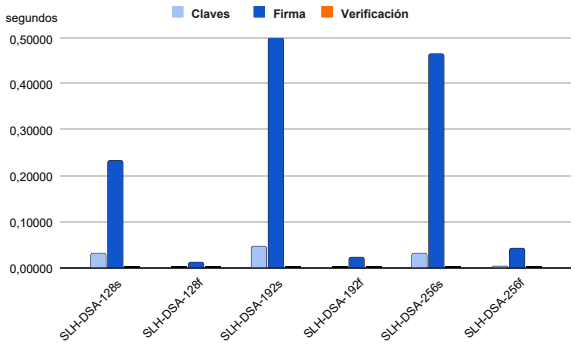


Figura 5. Tiempos de ejecución de los tres algoritmos de SLH-DSA.

En la Figura 5 se puede observar de manera clara el propósito de estos dos tipos de conjuntos de parámetros. Se puede ver que los conjuntos de tipos f son considerablemente más rápidos que los de tipo s, para todos los niveles de seguridad. De este gráfico también se puede destacar que los tiempos de verificación son considerablemente inferiores a los de los otros dos algoritmos, por el mismo motivo que lo eran en XMSS.

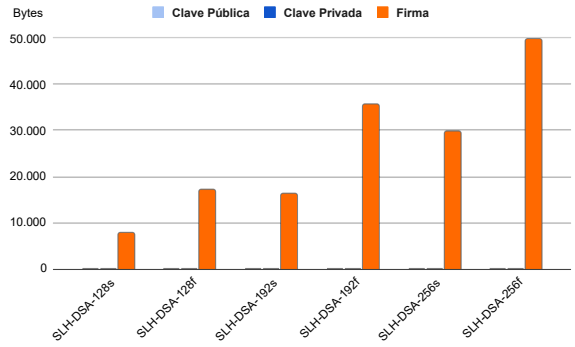


Figura 6. Tamaño de claves en SLH-DSA

En cuanto a los tamaños de claves y firmas en SLH-DSA, se puede ver, en la Figura 6, que los conjuntos de tipo s tienen firmas más pequeñas que los correspondientes de tipo f. En este caso, cabe destacar que la diferencia, dentro de un mismo nivel de seguridad, entre un conjunto de tipo s y uno de tipo f en cuanto al tamaño de las claves generadas, beneficia a los de tipo s y no es tan grande como lo era en el caso de los tiempos de ejecución. Esto indica que, en general, los conjuntos de tipo f ofrecen un mejor comportamiento que los de tipo s.

IV. Comparación de rendimiento entre esquemas de firma

En esta sección se presenta el análisis comparativo de los resultados de las pruebas de rendimiento hechas a cada esquema de firma, para, por último, obtener conclusiones sobre cuál de los esquemas de firma puede resultar más adecuado en cada situación.

Antes de presentar las comparaciones, es importante aclarar que, para que estas sean fiables, los conjuntos de parámetros

que se elijan para cada esquema deben ofrecer un nivel de seguridad lo más similar posible. Para ello, se han clasificado los conjuntos de parámetros en tres grupos de niveles de seguridad: bajo, medio y alto. En este trabajo únicamente se presentan las comparaciones entre conjuntos de nivel bajo, aunque los resultados se replican en los tres niveles de seguridad. A su vez, en todos los casos se utiliza la familia de *hash* SHA2 y no se utiliza *pre-hashing*.

Para realizar el estudio mencionado, se compararán en primer lugar, los tiempos de ejecución de los tres algoritmos de cada esquema y, a continuación, se comparan los tamaños de las claves y firmas. Con este análisis se pueden obtener conclusiones acerca del rendimiento de cada esquema en diferentes contextos típicos que se pueden dar a la hora de elegir un determinado esquema de firma a implementar.

Por último, cabe destacar que, para la comparación de tiempos de ejecución, se ha eliminado temporalmente el esquema XMSS ya que, como se ha explicado, los tiempos de ejecución no son competitivos respecto a los otros dos esquemas y, por lo tanto, al introducirlos en la comparación distorsionan demasiado la gráfica, dificultando la comparación de los otros dos esquemas.

IV-A. Comparación de tiempos de ejecución

En la Figura 7 se presenta la comparación directa entre los tiempos de ejecución de cada esquema de firma con conjuntos de parámetros de nivel de seguridad bajo.

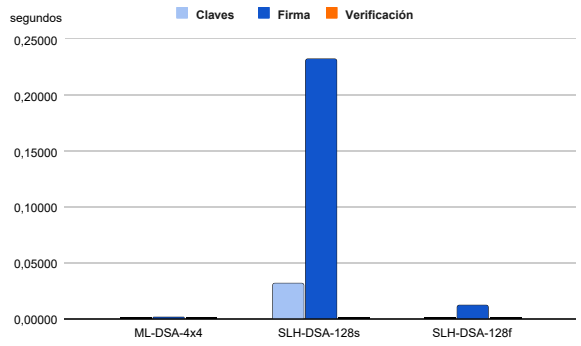


Figura 7. Comparación de los tiempos de ejecución.

Se puede observar que el esquema de firma con mejores tiempos de ejecución es ML-DSA en los tres algoritmos. En el caso de SLH-DSA de tipo f, la generación de claves y la verificación son prácticamente idénticas, y el proceso de generación de firmas es ligeramente más lento, aunque la diferencia no es demasiado importante, ya que la firma de SLH-DSA en este caso es muy rápida, de aproximadamente 0,01 segundos.

Por otro lado, sí que se observa una diferencia notable entre el esquema SLH-DSA tipo s con respecto a los otros dos. Aún sin tener una generación de firmas lenta, sí es considerablemente más lenta que la de los dos casos anteriores, de aproximadamente 0,24 segundos. Su generación de claves también es más lenta que en los casos de ML-DSA y SLH-DSA de tipo f.

Con estas diferencias, aún no se pueden sacar conclusiones sólidas acerca de qué esquema es objetivamente mejor en cualquier situación. Pero sí se puede decir que ML-DSA es el más rápido en los tres procesos, seguido muy de cerca por SLH-DSA de tipo f.

IV-B. Comparación de tamaño de claves

Por último, se comparan los tamaños de las claves públicas, privadas y de las firmas de cada esquema con conjuntos de parámetros de nivel de seguridad bajo. En este caso, se considera el esquema XMSS en la comparación.

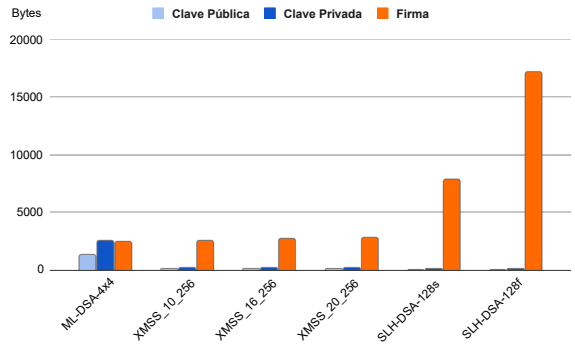


Figura 8. Comparación del tamaño de claves.

En la Figura 8 se puede ver que el tamaño de las claves de XMSS es mucho más competitivo que sus tiempos de ejecución. De hecho, en cuanto al tamaño total, teniendo en cuenta claves y firma, es el que presenta un valor más bajo. Sus claves tienen un tamaño considerablemente inferior a las de ML-DSA y son muy similares a las de SLH-DSA. Además, sus firmas son considerablemente inferiores a las de SLH-DSA, incluso en sus conjuntos de tipo s, optimizados para generar firmas más pequeñas.

ML-DSA sigue siendo competitivo en cuanto al tamaño de claves. Aunque sus claves son superiores a las del resto de esquemas, todas son inferiores a 5000 bytes, y sus firmas son las más pequeñas, junto con las de XMSS.

Respecto a SLH-DSA, aunque el tamaño de sus claves es de los más pequeños de los tres esquemas, sus firmas son las de mayor tamaño, incluso en los conjuntos de tipo s, optimizados para generar firmas más pequeñas.

IV-C. Discusión acerca de las comparaciones

Por último, una vez presentadas todas las comparaciones del rendimiento de los tres esquemas de firma seleccionados, se pueden deducir algunos hechos acerca de qué esquema de firma puede ser más adecuado en diferentes situaciones. Por ello, proponemos 4 escenarios con diferentes criterios de optimización:

- Escenario 1: sistemas con limitación en el tamaño de claves y firmas. Si los tiempos de ejecución no son críticos y la limitación principal es el tamaño total de las claves y firmas, XMSS puede ser una opción viable. Este esquema presenta el menor tamaño global entre los tres esquemas.
- Escenario 2: balance entre rendimiento y limitación de tamaño. Cuando se busca un equilibrio y los tamaños de clave no son limitantes, ML-DSA es la opción más adecuada. Sus tiempos de ejecución son los mejores en

general y, aunque sus claves son grandes, las firmas son más compactas que las de SLH-DSA.

- Escenario 3: minimización del tamaño de claves. Si el objetivo es reducir al máximo el tamaño de las claves públicas y privadas, SLH-DSA es preferible. Sus claves son muy pequeñas (inferiores a 128 bytes) y, aunque las firmas son más grandes que las de ML-DSA, los conjuntos tipo f ofrecen tiempos de ejecución competitivos.
- Escenario 4: aplicaciones con alta frecuencia de verificación. En entornos como validación de certificados, la operación de verificación se ejecuta muchas más veces que la firma. En este contexto, XMSS queda descartado por sus tiempos de ejecución elevados, especialmente en niveles de seguridad medio y alto. Tanto ML-DSA como SLH-DSA de tipo f ofrecen verificaciones rápidas y muy similares, por lo que la elección dependerá de otros factores como el tamaño de claves y firmas. En este caso, la opción preferible será ML-DSA si el tamaño de las claves no son limitantes, mientras que SLH-DSA será la elegida si se busca que el tamaño de las claves públicas y privadas sea el menor posible.

Por lo tanto, las comparaciones realizadas muestran que XMSS solo resulta adecuado en entornos con restricciones extremas en el tamaño de las claves y firmas y baja sensibilidad al tiempo. Entre ML-DSA y SLH-DSA, la decisión depende del criterio dominante: rendimiento global favorece ML-DSA, mientras que tamaño mínimo de claves beneficia a SLH-DSA.

V. Conclusiones

El objetivo principal de este trabajo ha sido el estudio y evaluación de las implementaciones de la librería Botan de tres esquemas de firma digital post-cuánticos, ML-DSA, XMSS y SLH-DSA, junto con la medición de diferentes parámetros para poder estimar su rendimiento. Una vez obtenidos los resultados, hemos podido hacer un análisis comparativo del rendimiento de los tres esquemas. Para ello, se generó así una plataforma de prueba para estos esquemas de firma. Dicha plataforma se encuentra disponible en el repositorio público mencionado anteriormente, y está accesible a cualquiera que quiera replicar, modificar o expandir cualquiera de estas pruebas.

Una vez obtenidas todas estas mediciones sobre el rendimiento de cada esquema, se han presentado tanto los resultados individuales, como las comparaciones directas del rendimiento de cada algoritmo, centradas principalmente en tiempos de ejecución y tamaño de claves y, finalmente, se han extraído unas conclusiones finales acerca del desempeño de cada esquema de firma en diferentes contextos en los que se necesite implementar alguno de estos algoritmos.

Como trabajo futuro, se puede utilizar esta plataforma para ejecutar estas mismas pruebas en sistemas diferentes, posiblemente con limitaciones mayores que las que tiene un ordenador convencional. Otra opción puede ser aumentar las pruebas implementadas, midiendo los mismos parámetros que evalúan el rendimiento, pero esta vez variando el tamaño de los mensajes que se firman, para comprobar cómo afecta su tamaño al rendimiento. También se podrían añadir a esta plataforma otros esquemas de firma digital post-cuánticos para extender el análisis comparativo. Como se ha mencionado anteriormente, sería de gran interés ampliar el estudio presentado teniendo en cuenta las optimizaciones de la especificaicón de SLH-DSA en XMSS. Por último, otra vía de trabajo futuro es hacer esta misma implementación utilizando otras herramientas, en concreto, otras librerías criptográficas e incluso otros lenguajes de programación.

Agradecimientos

L.H.A. agradece al proyecto del CSIC Seguridad Algebraica y Aplicaciones de la Inteligencia Artificial en la Criptología Actual y Postcuántica (SAIACAP), Ref.: 202450E017, su apoyo.

Referencias

[1] P.W. Shor: “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”, *SIAM Journal on Computing*, vol. 26, n. 5, pp. 1484-1509, https://epubs.siam.org/doi/10.1137/S0097539795293172, 1997.

[2] K.L. Grover, “Quantum mechanics helps in searching for a needle in a haystack”, *Phys. Rev. Lett.* 79 (1997), 325–328, https://doi.org/10.1103/PhysRevLett.79.325

[3] National Institute of Standards and Technology: “NIST Asks Public to Help Future-Proof Electronic Information”, https://www.nist.gov/news-events/news/2016/12/nist-asks-public-help-future-proof-electronic-information, 2016.

[4] J.W. Bos et al.: “CRYSTALS-Kyber Algorithm Specifications and Supporting Documentation, Version 3.02, NIST PQC Round 3 submission”, https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf, 2021.

[5] V. Lyubashevsky, L. Ducas, E. Kiltz, T. Lepoint, P. Schwabe, G. Seiler, and D. Stehle: “CRYSTALS-Dilithium”, https://pq-crystals.org/dilithium/index.shtml, 2020.

[6] T. Prest, P.-A. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Ricosset, G. Seiler, W. Whyte, and Z. Zhang: “FALCON”, https://falcon-sign.info/, 2020.

[7] A. Hulsing, D. J. Bernstein, C. Dobraunig, M. Eichlseder, S. Fluhrer, S.-L. Gazdag, P. Kampanakis, S. Kolbl, T. Lange, M. M. Lauridsen, F. Mendel, R. Niederhagen, C. Rechberger, J. Rijneveld, P. Schwabe, and J.-P. Aumasson: “SPHINCS+”, https://sphincs.org/, 2020

[8] C. Aguilar Melchor, N. Aragon, S. Bettaieb, L. Bidoux, O. Blazy, J.-C. Deneuville, P. Gaborit, E. Persichetti, G. Zémor and J. Bos: “HQC (Hamming Quasi-Cyclic)”, http://pqc-hqc.org/index.html, 2020

[9] National Institute of Standards and Technology: “FIPS 204: Module-Lattice-Based Digital Signature Standard (ML-DSA)”, en *U.S. Department of Commerce*, FIPS 204, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.204.pdf, 2024.

[10] A. Hülsing, D. Butin, S. Gazdag y J. Rijneveld: “XMSS: eXtended Merkle Signature Scheme”, en *Internet Research Task Force*, RFC 8391, https://datatracker.ietf.org/doc/html/rfc8391, 2018.

[11] National Institute of Standards and Technology: “Recommendation for Stateful Hash-Based Signature Schemes”, en *U.S. Department of Commerce*, SP 800-208, https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-208.pdf, 2020.

[12] National Institute of Standards and Technology: “FIPS 205: Stateless Hash-Based Digital Signature Standard (SLH-DSA)”, en *U.S. Department of Commerce*, FIPS 205, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.205.pdf, 2024.

[13] Randombit: ”Botan: C++ Crypto and TLS Library”, https://botan.randombit.net/, 2025.

FAIR$^+$: Protocolo Federado, Anónimo e Interoperable para Resoluciones colectivas

Macià Mut-Puigserver
Dpt. de Matemáticas e Informática
Universitat de les Illes Balears
Palma, España
macia.mut@uib.cat

M. Magdalena Payeras-Capellà
Dpt. de Matemáticas e Informática
Universitat de les Illes Balears
Palma, España
mpayeras@uib.cat

Rosa Pericàs-Gornals
Dpt. de Matemáticas e Informática
Universitat de les Illes Balears
Palma, España
rosa.pericas@uib.cat

Jaume Ramis-Bibiloni
Dpt. de Matemáticas e Informática
Universitat de les Illes Balears
Palma, España
jaume.ramis@uib.cat

Jordi Castellà-Roca
Dpt. de Ingeniería Informática y Matemáticas
Universitat Rovira i Virgili
Tarragona, España
jordi.castella@urv.cat

***Resumen*—Este artículo presenta FAIR$^+$ (protocolo Federado, Anónimo e Interoperable para Resoluciones colectivas), la extensión anónima del protocolo FAIR [1], un protocolo cross-chain diseñado para mejorar la escalabilidad, la transparencia y la auditabilidad en los procesos descentralizados de toma de decisiones. A diferencia de los sistemas tradicionales de votación basados en blockchain que dependen de una única blockchain global, FAIR adopta un modelo federado en el que las decisiones locales se realizan en blockchains independientes y posteriormente se agregan en un resultado global mediante un mecanismo seguro entre cadenas. El protocolo utiliza Soulbound Tokens (SBTs) y SBTs Rechazables (RejSBTs) para garantizar verificabilidad, inmutabilidad y aceptación explícita de los resultados, fortaleciendo así la integridad y la seguridad de las operaciones entre cadenas. En este artículo se presentan los fundamentos del reciente protocolo propuesto de votación cross-chain público, incorporando la propuesta de funcionalidad del voto anónimo, a través de mecanismos de ZKP y agregación homomórfica.**

***Index Terms*—blockchain, votación electrónica, token, cross-chain, gobernanza descentralizada, contratos inteligentes.**

I. Introducción

La integración de la tecnología blockchain en los procesos de toma de decisiones basados en votación ofrece beneficios sustanciales para mejorar la confianza y la transparencia. Gracias a su registro inmutable y auditable, la blockchain garantiza la rendición de cuentas al impedir la alteración o manipulación de los votos registrados. De este modo, refuerza la integridad de los resultados electorales y crea un entorno transparente y resistente a la manipulación. Así, la blockchain puede fomentar procedimientos de decisión democrática más seguros, fiables e inclusivos.

El proceso federado de toma de decisiones, según la aplicación, puede recibir diferentes denominaciones como votación distribuida o elecciones basadas en distritos [2], [3]. Puede definirse como un modelo de gobernanza en el que una decisión global surge de la agregación y coordinación de decisiones previas tomadas de forma independiente a nivel local o federado. En este marco, cada federación o entidad local ejerce autonomía para deliberar y alcanzar su propia decisión de acuerdo con sus necesidades y limitaciones contextuales, mientras que los resultados se integran posteriormente en un proceso de nivel superior que determina la decisión colectiva o global.

Consideramos que un proceso federado de toma de decisiones puede mitigar los problemas de escalabilidad inherentes a los procesos globales que emplean una única blockchain, además de proporcionar la auditabilidad entre cadenas. En el protocolo presentado recientemente, cada federación realiza su propia elección pública en una blockchain local, garantizando transparencia, seguridad y responsabilidad a nivel local, sin sobrecargar la infraestructura global única. Una vez validados los resultados locales, estos pueden agregarse y registrarse en una blockchain global mediante un módulo de interoperabilidad entre cadenas.

En este artículo presentamos la importancia de los sistemas de votación anónimos. A través de la incorporación de mecanismos como Zero Knowledge Proofs (ZKPs) y Compromisos de Pedersen, aplicables al protocolo propuesto, presentamos un primer estudio orientado a dotarlo de la propiedad de anonimato, característica fundamental en numerosos procesos de toma de decisiones.

Este enfoque innovador permite la votación anónima realizada localmente en diferentes cadenas y el revelado de los resultados locales, representados con la emisión de SBTs. Seguidamente, se propone la integración de estos, dentro de una única blockchain de destino. Al consolidar estos resultados tokenizados, el sistema garantiza una agregación de datos eficiente y auditable, mejorando la fiabilidad y transparencia global del proceso. Esta característica de escalabilidad no solo permite admitir un mayor número de participantes, sino que también facilita la gestión de escenarios de votación complejos, convirtiéndose en una solución robusta para las necesidades modernas de votación electrónica.

El sistema descentralizado y anónimo de toma de decisiones colectivas propuesto podría resultar especialmente útil en distintos escenarios. Por ejemplo, en una gran corporación multinacional con filiales en todo el mundo, cada sucursal podría utilizar su propia blockchain para llevar a cabo votaciones internas sobre políticas o decisiones empresariales.

Los resultados de cada sucursal podrían agregarse de forma segura en una única blockchain de destino, proporcionando una visión global y transparente de los resultados. De manera similar, en elecciones populares divididas por distritos, cada distrito podría utilizar su propia blockchain para gestionar las votaciones locales, y los resultados combinarse posteriormente a nivel superior para determinar el resultado general de la elección. Además, en el contexto de la gobernanza descentralizada, las organizaciones impulsadas por la comunidad (por ejemplo, las organizaciones autónomas descentralizadas, DAOs) pueden apoyarse en capítulos locales o subcomités para votar sobre propuestas específicas, tras lo cual un módulo entre cadenas consolida las decisiones locales en un resultado global unificado, garantizando que todos los votos se registren y agreguen de forma transparente. Esta escalabilidad y flexibilidad convierten el sistema de votación descentralizado y anónimo en una solución robusta para las necesidades contemporáneas de votación.

II. Estado del arte

La tecnología blockchain ha sido ampliamente explorada como base para sistemas de votación electrónica seguros y transparentes. Estudios de referencia, como [4], ofrecen una visión general exhaustiva de cómo la blockchain puede abordar los requisitos fundamentales de la votación electrónica, incluyendo transparencia, inmutabilidad, verificabilidad y resistencia a la manipulación. Estos trabajos destacan que las soluciones basadas en blockchain pueden aumentar la confianza del votante al garantizar que, una vez emitido un voto, este no puede ser alterado ni eliminado, permitiendo al mismo tiempo que los votantes y auditores verifiquen la corrección del resultado global. No obstante, también señalan desafíos significativos como la escalabilidad, la preservación de la privacidad, la usabilidad para votantes no técnicos y la complejidad de gestionar elecciones a gran escala con infraestructuras heterogéneas.

Paralelamente, el trabajo [5] revisa distintos esquemas de votación basados en blockchain y los clasifica según su arquitectura, mecanismos de consenso y grado de descentralización. Los autores argumentan que muchas de las propuestas existentes presentan un alcance limitado, ya sea porque se centran en elecciones de pequeña escala o porque dependen de una única infraestructura blockchain, lo que puede generar cuellos de botella en el rendimiento y riesgos de centralización. Su propuesta apunta hacia diseños más flexibles e interoperables, que integren primitivas criptográficas con infraestructuras blockchain para mejorar tanto la seguridad como la escalabilidad.

El estudio presentado en [6] ofrece un análisis comparativo entre los sistemas de votación tradicionales y los basados en blockchain. Los sistemas tradicionales son centralizados, dependen de la confianza institucional, son vulnerables a fallos de un único punto y ofrecen una transparencia limitada, aunque resultan familiares para los votantes. Los sistemas basados en blockchain, en cambio, proporcionan descentralización, transparencia e inmutabilidad mediante un registro resistente a la manipulación, lo que incrementa la seguridad y la auditabilidad. Sin embargo, afrontan desafíos relacionados con la usabilidad y la adopción, los límites de escalabilidad en elecciones de gran tamaño y los mayores costes de transacción a medida que crece la red.

En cuanto a la votación anónima sobre blockchain, en [7] los autores presentan un protocolo que preserva la privacidad tanto de la identidad del votante como del voto. Para ello, emplean un un algoritmo de recuento basado en criptografía homomórfica, que permite agregar los votos sin revelar su contenido, así como pruebas de ZKPs para ocultar la identidad del votante. Por su parte, en [8] se propone un protocolo de votación anónima basado en circuitos ZKP, que incluye tanto un sistema de verificación de elegibilidad de los votantes y un mecanismo para la votación, garantizando la privacidad del votante sin comprometer la verificabilidad del proceso.

Aunque la blockchain garantiza integridad y auditabilidad, siguen existiendo problemas de interoperabilidad, privacidad del votante y soporte para procesos complejos y de gran escala. Esto motiva el desarrollo de nuevos protocolos, como el que se propone en este trabajo, que adoptan mecanismos para conseguir procesos de toma de decisiones anónimas, el uso de modelos federados o entre cadenas para mejorar la escalabilidad, la verificación y la robustez de los sistemas de votación electrónica.

III. Conceptos previos

III-A. Programación en Blockchain

En el contexto de la tecnología blockchain, los *contratos inteligentes* son programas autoejecutables desplegados en un libro de contabilidad distribuido que hacen cumplir automáticamente reglas y acuerdos predefinidos sin necesidad de intermediarios. Una vez desplegado, el contrato inteligente se ejecuta de forma determinista y transparente, ya que su lógica se replica en todos los nodos validadores de la red, garantizando confianza, auditabilidad y resistencia a la manipulación.

Además, los *tokens* representan activos digitales o unidades de valor gestionadas en la blockchain mediante protocolos estandarizados (por ejemplo, ERC-20 o ERC-721 en Ethereum) que facilitan la interoperabilidad. Los tokens pueden incorporar distintas funcionalidades, incluyendo el intercambio similar a monedas, derechos de gobernanza o identificadores únicos para activos digitales y físicos. En conjunto, los contratos inteligentes y los tokens constituyen bloques fundamentales para las aplicaciones descentralizadas, permitiendo nuevos marcos de programación distribuida.

III-B. Tokens SBT y RejSBT

Los *Soulbound Tokens* (SBT) se han propuesto recientemente como una nueva clase de activos digitales no transferibles diseñados para representar compromisos, credenciales o afiliaciones dentro de ecosistemas descentralizados. A diferencia de los tokens fungibles o no fungibles convencionales, que pueden intercambiarse libremente entre usuarios, los SBT están permanentemente vinculados a una dirección específica en la blockchain [9].

Habitualmente, los SBT se utilizan para codificar aspectos de identidad personal, reputación o relaciones sociales de manera verificable y resistente a la manipulación. Estos tokens resultan especialmente adecuados para aplicaciones como la gestión de identidad descentralizada, pruebas de asistencia,

certificaciones académicas o profesionales y sistemas de gobernanza donde la transferibilidad podría comprometer la integridad. En el sistema propuesto en este trabajo, los SBT se utilizan para almacenar el resultado del proceso de toma de decisiones y facilitar la transferencia de los resultados entre cadenas.

Los *Rejectable Soulbound Tokens* (RejSBT) representan una forma avanzada de SBT caracterizada por su no transferibilidad inherente, junto con un mecanismo criptográficamente verificable que permite a los destinatarios aceptar o rechazar explícitamente el token dentro de una ventana temporal específica [10]. Este diseño mantiene la propiedad principal de preservación de identidad (tokens no transferibles), pero introduce una funcionalidad controlable para el destinatario, con capacidad de revocación durante el proceso de emisión. Esta característica resulta especialmente relevante en la emisión de credenciales digitales, derechos de acceso o acreditaciones donde se requiere el consentimiento explícito del destinatario antes de que la autoridad emisora pueda generar el token. En este trabajo, los tokens RejSBT se emplean para aumentar la precisión de las operaciones entre cadenas.

IV. Visión general del sistema

El marco de toma de decisiones propuesto se estructura en dos fases diferenciadas:

1. **Local:** los decisores o votantes eligen una opción entre las alternativas propuestas en un entorno local.
2. **Global:** todas las decisiones locales se integran mediante el protocolo de interoperabilidad de cadenas (cross-chain).

Los actores implicados en estas dos fases y los roles que desempeñan son los siguientes:

VA *Autoridad de Votación*: Entidad responsable de desplegar el contrato electoral y gestionar los parámetros del proceso de votación.
PR *Proponentes*: Entidades o individuos que crean propuestas para ser votadas.
VT *Votantes*: Participantes o decisores que eligen una propuesta y confirman su elección.
GA *Autoridad Global*: Entidad responsable de desplegar el contrato entre cadenas.
RL *Relayer*: Entidad que transmite la información del token a la blockchain de destino.

El proceso de la **Fase Local** se muestra en la Fig. 1. En esta fase, la Autoridad de Votación (VA) comienza desplegando el contrato inteligente en la blockchain para gestionar el proceso de toma de decisiones (Paso 1). Este contrato actúa como controlador de las acciones de votación y garantiza que las operaciones sensibles (como la declaración de resultados) se ejecuten conforme a la aprobación colectiva de los votantes autorizados. Denominaremos a este contrato inteligente *Contrato de Elección Colectiva*.

A continuación, los Proponentes (PR) —por ejemplo, partidos políticos— invocan la función del *Contrato de Elección Colectiva* para registrar las distintas opciones de votación (Paso 2). Seguidamente, durante la etapa de decisión, los Votantes (VT) eligen una opción mediante el mismo contrato (Paso 3). Tras concluir el período de votación, la VA ejecuta una función del contrato para contabilizar los resultados.

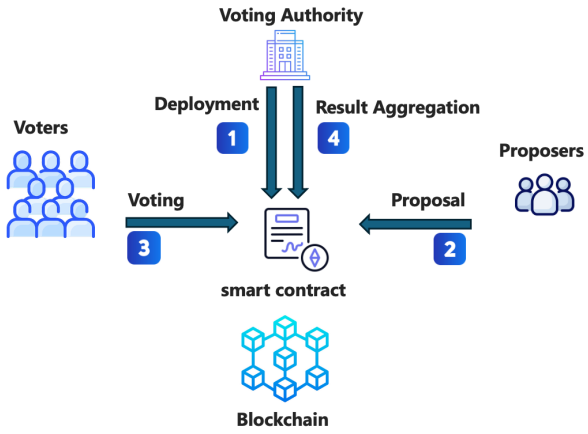


Figura 1. Fase local del proceso de toma de decisiones.

Se emite un nuevo SBT con los metadatos de la opción ganadora (Paso 4), representando el resultado electoral en la blockchain local. Este esquema de decisión garantiza un proceso transparente, seguro y eficiente.

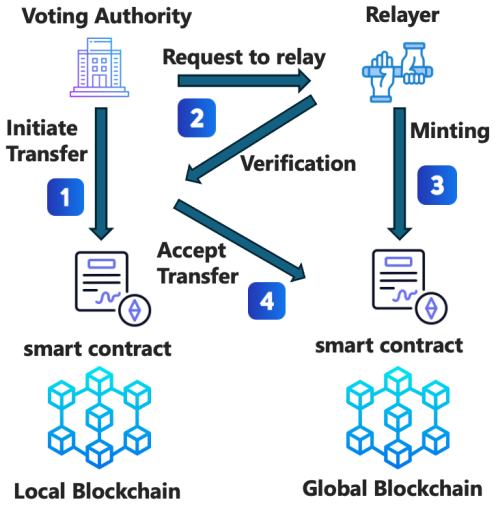


Figura 2. Fase global del proceso de toma de decisiones.

En la **Fase Global**, los tokens que representan los distintos resultados en las blockchains locales pueden reunirse en una única blockchain mediante el módulo cross-chain (Fig. 2). De esta forma, las elecciones locales pueden consolidarse de manera segura en una sola plataforma y convertirse en una elección global. Así, el sistema de votación puede superar el problema de escalabilidad presente en algunas aplicaciones blockchain.

En esta fase, cada VA bloquea el token de resultado en la blockchain local (Paso 1), garantizando que dicho resultado no pueda reutilizarse ni transferirse de nuevo a la cadena global. Luego, la VA solicita la operación de transmisión al Relayer (RL) (Paso 2). Tras verificar que el token está bloqueado, el RL genera un nuevo RejSBT con los resultados locales en la blockchain global (Paso 3), definiendo la dirección de la VA como propietaria del RejSBT. Finalmente, la VA acepta el nuevo token en la cadena global si los metadatos del token son correctos (Paso 4).

V. Protocolo para la toma de decisiones

En esta sección se presentan las diferentes etapas del protocolo de toma de decisiones sin anonimato propuesto. El protocolo descrito en esta sección se basa en el propuesto previamente por los mismos autores en [1]. Una descripción más detallada, junto con los aspectos específicos de su implementación, puede encontrarse en dicha referencia. Los mecanismos de anonimato destinados a preservar la privacidad de la identidad de los usuarios y de su voto se abordará en la siguiente sección.

V-A. Local Stage: Toma de Decisiones en Local

La etapa inicial, denominada *Local Stage*, en la que se lleva a cabo el procedimiento de toma de decisiones a nivel local, se describe formalmente en el Algoritmo 1. En aras de la claridad y la concisión, en lo sucesivo se hará referencia al *Collective-Choice Contract* como *C_Choice*.

Algorithm 1: Protocolo para la toma de decisiones: Local Stage

`Step 1: Deployment`

On-Chain **Ejecutar:** ***VA*** ▷ *Deploy C_Choice*

On-Chain **Entrada:** *proposers, voters, proposalDuration, confirmationDuration*
On-Chain **Salida:** *new C_Choice deployed*

`Step 2: Proposal`

On-Chain **Ejecutar:** ***PR*** ▷ *C_Choice.submitProposal()*

Requiere: ***PR*** *Authentication*
Requiere: *Before proposalDuration Deadline*

On-Chain **Entrada:** *statement*
On-Chain **Salida:** *new voting proposal*

`Step 3: Voting`

On-Chain **Ejecutar:** ***VT*** ▷ *C_Choice.confirmProposal()*

Requiere: ***VT*** *Authentication*
Requiere: *After proposalDuration Deadline*
Requiere: *Before confirmationDuration Deadline*

On-Chain **Entrada:** *proposalIndex*
On-Chain **Salida:** *new voting option, VT has voted*

`Step 4: Result Aggregation`

On-Chain **Ejecutar:** ***VA*** ▷ *C_Choice. winningProposal ()*

Requiere: ***VA*** *Authentication*
Requiere: *After confirmationDuration Deadline*

On-Chain **Entrada:**
On-Chain **Salida:** *new voting result token*

De acuerdo con el Algoritmo 1, VA implementa un nuevo contrato inteligente *C_Choice* en la cadena de bloques con los siguientes parámetros en el constructor:

`proponente`
: Matriz de PRs. Las direcciones de la cadena de bloques, que identifican a cada proponente, se introducen en esta matriz.

`votante`
: Matriz de VTs. Las direcciones de la cadena de bloques, que identifican a cada participante, se introducen en esta matriz.

`proposalDuration`
: Fecha límite fijada para almacenar todas las propuestas que se someterán a votación. Antes de esta fecha límite, cualquier PR puede presentar una propuesta.

`confirmationDuration`
: Fecha límite establecida para recibir votos. Los VT emiten su voto después de la fecha límite `proposalDuration` y antes de la fecha límite `confirmationDuration`.

El constructor valida las direcciones de todos los PRs y VTs estableciendo el valor de retorno de la asignación correspondiente de cada dirección en verdadero. Además, la dirección de blockchain que realiza el despliegue del smart contract se fija como VA. A continuación, el constructor establece la fecha límite para las propuestas y la recepción de votos. Por último, se implementa un nuevo contrato inteligente para crear el token de resultado con los metadatos de la propuesta ganadora. Tenga en cuenta que la última acción podría realizarse antes de la implementación de este contrato inteligente y, en ese caso, la dirección del contrato inteligente que implementa el token de resultado debería proporcionarse como un nuevo parámetro del constructor.

V-B. Global Stage: Recopilación de Resultados

Para garantizar la integridad y la continuidad del protocolo de votación en blockchains heterogéneas, proponemos un mecanismo entre blockchains para transferir de forma segura los tokens de resultados entre las redes. Esto permite agregar las decisiones locales en un resultado colectivo global.

El mecanismo de transferencia entre cadenas propuesto implica los siguientes pasos:

- Para iniciar el proceso *cross-chain*, el VA llama a la función `callInititateTransfer()` para bloquear el token que representa el resultado de la votación local. Operación descrita en el Algoritmo 2.

Algorithm 2: Global Stage: Iniciación del mecanismo de transferencia

On-Chain **Ejecutar:** ***VA*** ▷ *C_Choice.callInititateTransfer()*
On-Chain **Ejecutar:** *callInititateTransfer* ▷ *CrossChainTransfer.initiateTransfer()*

Requiere: ***VA*** *Authentication*
Requiere: ***C_Choice*** *Authentication*

On-Chain **Entrada:** *crossChainTransferContract, [tokenId]*
On-Chain **Salida:** *event*

if *sbt.tokenOwner(tokenId) == votingAuthority & token not transferred* **then**
  sbt.transferred(tokenId) = true; \\token locked
  Emit event transfer initiation
end

- RL verifica la transacción de bloqueo en la blockchain de origen (ver Algoritmo 3).
- RL comprueba que los detalles de la transacción sean precisos, que la dirección del propietario esté correcta-

Algorithm 3: Global Stage: Verificación preliminar al Token Relay

Off-Chain* Ejecutar:** ***VA $\succ$ *RL: Request to relay token [tokenId]*
On-Chain* Ejecutar:** ***RL $\triangleright$ *Verify lock transaction*

Requiere: *[tokenId], Token locked*

***Off-Chain* Entrada:** *[tokenId]*
***On-Chain* Salida:** *tokenId lock status*

mente registrada y realiza el minteo del token RejSBT en la blockchain de destino. Finalmente, de acuerdo con el Algoritmo 4, VA acepta el token RejSBT. Esta aceptación indica que los datos son correctos y que el token de los resultados en la cadena de bloques de origen permanece bloqueado.

Algorithm 4: Global Stage: Token Relay

On-Chain* Ejecutar:** ***RL $\triangleright$ *TokenMinting.mintToken()*
On-Chain* Ejecutar:** ***VA $\triangleright$ *TokenMinting.acceptTranfer()*

Requiere: ***RL*** *Authentication*
Requiere: ***VA*** *Authentication*

***On-Chain* Entrada:** *sourceAddress, sourceTokenId, sourceBlockchain, token metadata*
***On-Chain* Salida:** *new result token*

```
TokenMinting.mintToken()
if Token[tokenId] from destination blockchain not already
 processed then
    address to = addressMapping[sourceAddress];
    // Mapping between blockchain source-destination
      addresses
    mint(to, newTokenId)
end
```

```
TokenMinting.acceptTransfer()
if transferableOwners[tokenId] == msgSender &
 states[tokenId] == minted then
    owners[tokenId] = to;
    states[tokenId] = State.Accepted;
    transferableOwners[tokenId] = address(0);
    emit AcceptTransfer(from,to,tokenId);
end
```

VI. Anonimato del voto

En protocolos de toma de decisiones como el presentado, una de la propiedades fundamentales es conseguir el anonimato del voto. Para aplicar esta propiedad en el protocolo propuesto de votación federada, en esta sección se presenta el uso de mecanismos de Zero-Knowledge Proof (ZKP) para verificar tanto que el usuario tiene derecho a voto como la validez del voto, combinado con la ocultación del voto mediante el uso de Compromisos de Pedersen.

VI-A. Verificación anónima del votante

Durante la fase de toma de decisiones en local, previo al paso de votación (Algoritmo 1, Paso 3), los votantes, deben registrarse para formar parte del conjunto de votantes (VT). Este registro se debe llevar a cabo off-chain, mediante un sistema privado de verificación de la identidad del votante y de su derecho a voto. En caso de registro positivo, el sistema genera un identificador anónimo único ("*nullifier*", n), el cual será utilizado posteriormente para verificar el derecho a voto y garantizar que cada VT únicamente introduce un voto en el sistema. Dicho *nullifier*, en ningún caso se almacena en el sistema, debe ser almacenado de forma privada por el votante.

En esta fase el sistema genera un árbol de merkle para la verificación de los votantes registrados. Las hojas del árbol de merkle, con el fin de garantizar una adecuada seguridad, están formadas por el compromiso de identidad (c), definido como el hash del secreto de identidad (s). A su vez, este secreto se obtiene aplicando un hash al identificador de la identidad del votante (ID_v) combinado con el nullifier generado para dicho votante. Entonces:

$$s = H(\mathrm{ID}_v \parallel n) \tag{1}$$

$$c = H(s) \tag{2}$$

A partir de esta configuración, se define un circuito ZKP [11] encargado de verificar que el usuario está registrado en la votación. Dicho circuito recibe como clave privada el secreto de identidad (s) del usuario y mediante el árbol de merkle verifica el derecho de votación del usuario. El resultado de la ejecución del circuito ZKP se proporciona como parámetro de entrada en el Paso 3 de la fase de toma de decisiones en local. Es importante destacar que la ejecución del circuito necesario para obtener la prueba de autenticidad se realiza de manera local por cada votante; en consecuencia, el secreto de su identidad s no se revela en ningún momento.

VI-B. Voto anonimo

Para garantizar el anonimato en el voto, es decir, ocultar el contenido de los votos, se propone el uso de Compromisos de Pedersen (*Pedersen Commitment*) [12]. Este mecanismo posee la propiedad de suma homomórfica, lo que permite obtener resultados agregados sin revelar los datos individuales. Específicamente, esta propiedad posibilita calcular la suma de valores directamente sobre sus versiones cifradas, de modo que, tras el descifrado, el resultado coincide con la suma de los valores originales. Esta funcionalidad, hace los compromisos de Pedersen, especialmente utiles para el recuento de votos, ya que permite agregar los votos sin exponer cada votación individual.

Un compromiso de Pedersen se define como:

$$C_i = v_i G + r_i H \tag{3}$$

donde:

- v_i: valor (voto), representado como un número binario $\{0, 1\}$.
- r_i: valor aleatorio secreto generado individualmente por el votante.
- $G, H \in \mathbb{G}$: puntos generadores públicos e independientes.

Entonces, cada votante puede introducir su voto anónimo en el smart contract mediante un compromiso de Pedersen. Y los diferentes votos recibidos se pueden sumar gracias a la propiedad aditiva homomórfica, definida como:

$$C_{\text{sum}} = \sum_i C_i = \left(\sum_i v_i\right) G + \left(\sum_i r_i\right) H \tag{4}$$

Además de los mecanismos propuestos, con el objetivo de evitar la trazabilidad y la posible identificación o pseudoidentificación de la identidad del votante, se recomienda emplear direcciones blockchain generadas específicamente para el proceso de votación, así como redes blockchain que garanticen anonimato o, alternativamente, una red descentralizada de relayers encargada de ejecutar las transacciones en el contrato inteligente.

VI-C. Revelación de resultados

Una vez finalizado el periodo de votación, el smart contract contiene el compromiso agregado de la votación. Entonces, para obtener el resultado final de la votación se debe revelar el sumatorio de votos.

Para una mayor descentralización, proponemos seguir un mecanismo de Threshold Tally [13] (recuento por umbral), donde en la fase de votación, los participantes dividen su r_i y lo comparten con una red distribuida de nodos privada o bien con distintas autoridades. De tal forma que se distribuye la confianza y se evita que la autoridad (VA) pueda manipular o conocer los votos individuales.

Entonces, el uso de Threshold Tally obliga que un mínimo de t de n autoridades o nodos intervengan en la revelación de los resultados. Distribuyendo la revelación del resultado final e impidiendo que individualmente la VA pueda obtener los votos individuales.

VII. Conclusiones

Este artículo presenta el protocolo FAIR$^+$, la extensión anónima del reciente protocolo FAIR presentado en [1] consiguiendo un protocolo anónimo que preserva la privacidad del voto sobre el modelo de toma de decisiones federadas, con el uso de mecanismos cross-chain para ampliar la escalabilidad, la transparencia y la robustez de los sistemas de votación electrónica basados en blockchain.

El protocolo FAIR$^+$ se enfoca en proporcionar las funcionalidades de anonimato y preservación de la privacidad del voto, El diseño de FAIR$^+$ propone el uso de técnicas criptográficas avanzadas, como pruebas de conocimiento cero (ZKP) y Compromisos de Pedersen, para garantizar la verificación privada del derecho al voto, el anonimato del votante y la confidencialidad del voto. Estas técnicas permiten a los votantes demostrar que tienen derecho a votar sin revelar su identidad ni el contenido de su voto, asegurando así la privacidad y la integridad del proceso electoral. Además, el uso de compromisos de Pedersen juntamente con el uso de técnicas de Threshold Tally, permite conseguir que únicamente se revele el resultado final de la votación, sin exponer los votos individuales emitidos por los votantes.

Por su parte, el protocolo FAIR$^+$ mantiene las características fundamentales del protocolo FAIR original, incluyendo la escalabilidad y la interoperabilidad entre cadenas. Al igual que en FAIR, las decisiones locales se llevan a cabo en blockchains independientes y posteriormente se consolidan en un resultado global mediante un módulo cross-chain. Este enfoque federado permite mitigar los cuellos de botella de escalabilidad asociados a los sistemas basados en una sola cadena, al tiempo que garantiza la transparencia y la auditabilidad del proceso electoral.

Como trabajo futuro, se plantea la implementación de una prueba de concepto y su correspondiente evaluación del protocolo FAIR$^+$, así como la verificación de las propiedades de seguridad, así como la exploración de posibles optimizaciones que mejoren su eficiencia y usabilidad. Asimismo, se considera la extensión del protocolo mediante el uso de una red descentralizada de relayers, encargados de realizar la transmissión crosschain de los resultados. Y estudiar la posibilidad de reutilizar estos relayers como una red de nodos participantes en la fase de revelación de los resultados mediante Threshold Tally.

Agradecimientos

Proyecto de investigación BLOBSEC PID2021-122394OB-I00 financiado por MICIU/AEI /10.13039/501100011033 y por FEDER, EU.

Referencias

[1] M. Mut-Puigserver, M.M. Payeras-Capellà, R. Pericàs-Gornals, J. Ramis-Bibiloni, “FAIR - Federated Agreement for Independent Resolution: a cross-chain decision-making protocol,” Proceeding of the 35th International Conference on Computer Theory & Applications (ICCTA 2025), November 2025.

[2] A. Filos-Ratsikas, E. Micha, and A. A. Voudouris, “The distortion of distributed voting,” Artificial Intelligence, vol. 286, ISSN 0004-3702, Jan. 2020. https://doi.org/10.1016/j.artint.2020.103343.

[3] A. Mitra, “Agent-based Simulation of District-based Elections with Heterogeneous Populations,” Proceeding of the of the 2023 International Conference on Autonomous Agents and Multiagent Systems (AAMAS), ISBN 9781450394321, June 2023.

[4] M. A. Salahuddin, A. Al-Fuqaha, M. Guizani, and M.-S. Hossain, “Blockchain for securing electronic voting systems: a survey of architectures, trends, solutions, and challenges,” Cluster Computing, vol. 28, Article No. 132, 2025.

[5] M. Sharp, L. Njilla, C.-T. Huang, and T. Geng,“Blockchain-Based E-Voting Mechanisms: A Survey and a Proposal,” Network, vol. 4, no. 4, Article 21, 2024, published 26 September 2024. DOI: 10.3390/network4040021

[6] R.V. Nikhare, B.R. Chandavarkar, “A comparative analysis of traditional versus blockchain-based voting systems,” In: 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT). IEEE (2023). DOI: 10.1109/icccnt56998.2023.10307550

[7] W. Chai, M. Liu, Z. Zhang, and L. Lv, “Blockchain-based privacy preserving electronic voting protocol”, International Journal of Network Security, Vol. 24, No. 2, Mar. 2022. DOI: 10.6633/IJNS.202203 24(2).05

[8] I. N. Ahmad, P. Sukarno and A. A. Wardana, “Implementation of Zero Knowledge Proof and Smart Contract for E-Voting System on Polygon Blockchain,” 2025 International Conference on Advancement in Data Science, E-learning and Information System (ICADEIS), Bandung, Indonesia, 2025, pp. 1-6, doi: 10.1109/ICADEIS65852.2025.10932999.

[9] V. Buterin, P. Ohlhaver, and E. G. Weyl, Decentralized Society: Finding Web3’s Soul. SSRN Electronic Journal, 2022. DOI: 10.2139/ssrn.4105763.

[10] R. Pericàs-Gornals, M. Mut-Puigserver, M.M. Payeras-Capellá, M.À. Cabot-Nadal, and J. Ramis-Bibiloni, “Digital credentials management system using rejectable Soulbound Tokens,” Annals of Telecommunications, 2024. DOI: 10.1007/s12243-024-01032-6.

[11] M. Bellés-Muñoz, M. Isabel, J. L. Muñoz-Tapia, A. Rubio and J. Baylina, “Circom: A Circuit Description Language for Building Zero-Knowledge Applications,” in IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 6, pp. 4733-4751, Nov.-Dec. 2023, doi: 10.1109/TDSC.2022.3232813

[12] Pedersen, T.P. (1992). Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing. In: Feigenbaum, J. (eds) Advances in Cryptology — CRYPTO ’91. CRYPTO 1991. Lecture Notes in Computer Science, vol 576. Springer, Berlin, Heidelberg. DOI: 10.1007/3-540-46766-1_9

[13] R. Cramer, R. Gennaro, y B. Schoenmakers, “A secure and optimally efficient multi-authority election scheme,” European Transactions on Telecommunications, vol. 8, no. 5, pp. 481–490, 1997.

Diseño y Desarrollo de un Sistema de Respuesta a Ciberamenazas Basado en Large Language Models y Retrieval Augmented Generation

Rodrigo Tavares de Pina Simões
r.tavaresdepina@alumnos.upm.es

Luis Pérez Miguel
luis.pmiguel@upm.es

Xavier Larriva-Novo
xavier.larriva.novo@upm.es

David Álvarez Muñiz
david.alvarez.muniz@upm.es

Carmen Sánchez-Zas
carmen.szas@upm.es

Victor A. Villagrá González
victor.villagra@upm.es

Departamento de Ingeniería de Sistemas Telemáticos (DIT),
Escuela Técnica Superior de Ingenieros de Telecomunicación (ETSIT),
Universidad Politécnica de Madrid (UPM). Avda. Complutense 30, 28040 Madrid

***Resumen*—La ciberseguridad se ha consolidado como un pilar fundamental para las organizaciones ante el aumento constante y la creciente sofisticación de los ciberataques. Uno de los principales desafíos es garantizar una detección y una respuesta ágiles frente a estas amenazas. En este ámbito, los sistemas de Gestión de Información y Eventos de Seguridad (SIEM) son herramientas esenciales; no obstante, el análisis de sus alertas requiere un alto consumo de recursos. La adopción de Inteligencia Artificial Generativa combinada con un enfoque de Generación Aumentada por Recuperación (RAG) permite mejorar la evaluación de los registros de ataque mediante la incorporación de información externa, incrementando la precisión y reduciendo los tiempos de respuesta. Este trabajo propone un sistema basado en IA generativa y RAG capaz de procesar registros de un SIEM, clasificar amenazas según su nivel de riesgo y recomendar Cursos de Acción orientados a su mitigación. La validación se realiza en dos escenarios: un ataque simulado con CALDERA y el análisis de registros correspondientes a una práctica tipo Capture the Flag. La evaluación considera tanto la capacidad de detección del sistema como la calidad y utilidad de los informes y recomendaciones generados.**

***Index Terms*—Ciberseguridad, Modelos de lenguaje (LLM), Recuperación aumentada por generación (RAG), Respuesta a incidentes, Análisis de amenazas.**

I. Introducción

La creciente complejidad de los ciberataques, unida al volumen cada vez mayor de datos que generan los sistemas corporativos, plantea un desafío crítico para la ciberseguridad. Las herramientas de detección de intrusiones y manejo de eventos como los *Security Information and Event Management* (SIEM), resultan indispensables para correlar eventos y detectar ataques. Sin embargo, la elevada cantidad de alertas que producen junto con la dificultad para discernir aquellas realmente críticas, derivan en la llamada fatiga del analista de seguridad.

Una de las soluciones planteadas en este ámbito es el desarrollo y aplicación de la Inteligencia Artificial, y en particular de los Modelos de Lenguaje a Gran Escala [1] (LLMs). Estos modelos destacan por su capacidad de procesar grandes volúmenes de datos no estructurados y generar respuestas coherentes. No obstante, también presentan limitaciones significativas, como la posibilidad de producir alucinaciones: respuestas incorrectas pese a tener una lógica interna coherente [2], lo que dificulta su aplicación directa en contextos críticos.

Retrieval-Augmented Generation [3] (RAG) permite combinar el potencial generativo de los LLMs con la recuperación de información contextual de fuentes externas. Este paradigma permite reducir el riesgo de respuestas inexactas, además de habilitar la integración de bases de conocimiento heterogéneas como: marcos de referencia tipo MITRE ATT&CK [4] o guías de ciberseguridad como los catálogos NIST [5].

Con este objetivo, este trabajo formula tres hipótesis que permiten evaluar el impacto del enfoque LLM–RAG en la clasificación y respuesta a ciberamenazas. Estas hipótesis se centran en analizar si la recuperación de conocimiento mejora la identificación técnica, la utilidad de las recomendaciones y la eficiencia operativa del sistema:

- H_1 **Impacto en la clasificación de amenazas**. Un modelo LLM-RAG mejora la calidad y fundamentación de la clasificación de amenazas.
- H_2 **Impacto en la generación de Cursos de Acción**: La integración de RAG permite sugerir contramedidas más acertadas.
- H_3 **Impacto en el tiempo de ejecución**: El uso de RAG reduce el tiempo de respuesta.

II. Trabajos relacionados

La aplicación de los LLMs ha transformado la ciberseguridad gracias a su capacidad para comprender el lenguaje natural y generar texto coherente a partir de grandes volúmenes de datos no estructurados [6], lo que permite automatizar tareas de análisis tradicionalmente dependientes de una fuerte intervención humana [7, 8]. Estos modelos pueden adaptarse a tareas específicas mediante técnicas de *finetuning*, mejorando su desempeño en detección de anomalías y vulnerabilidades en código [9], apoyo al análisis estático [10] o generación de informes de inteligencia sobre amenazas [11]. Sin embargo, siguen existiendo limitaciones relevantes, como las alucinaciones y la falta de control sobre las fuentes de conocimiento [1, 12].

Para mitigar estas deficiencias, el enfoque Retrieval-Augmented Generation (RAG) combina modelos generativos con un componente de recuperación que aporta contexto externo, frente a alternativas basadas únicamente en ampliar el *prompt* con documentación estática [13]. En el ámbito concreto del análisis de amenazas, se han propuesto soluciones basadas en LLM sin RAG para clasificar alertas según su nivel de interés [14], asistir en tareas de respuesta mediante agentes conversacionales [15] o integrarse en sistemas de detección host-based [16]. Más recientemente, trabajos como MorSE [17] y RAGlog [18] incorporan recuperación semántica sobre documentación de referencia y conjuntos de logs, respectivamente, evidenciando el potencial de RAG para mejorar la contextualización y explicabilidad de las recomendaciones en ciberseguridad.

III. Propuesta

En este contexto, el presente trabajo propone un sistema de respuesta a ciberamenazas basado en LLM y RAG que integra un entorno SIEM real (Wazuh [19]) con un motor de recuperación semántica y catálogos especializados como MITRE ATT&CK [4] y NIST [5]. Esta integración permite clasificar amenazas, contextualizarlas con conocimiento verificable y generar Cursos de Acción (COAs) fundamentados. Además, este trabajo combina un modelo de código abierto, como Llama 3.1 [20], con información estructurada, reduciendo la dependencia del analista, mejorando la fiabilidad de las recomendaciones y acelerando la respuesta ante incidentes.

Su validación se desarrolla en un escenario realista que incluye un ataque simulado mediante CALDERA [21] y un ejercicio Capture the Flag, en el cual se demuestra la aplicabilidad y reproducibilidad.

III-A. Arquitectura del sistema

En la Figura 1, se presenta la propuesta del flujo operacional del sistema que combina tres componentes principales: un sistema SIEM, que recopila y analiza eventos de seguridad; un motor de recuperación de información, que dado un conjunto de eventos inicial recupera un contexto específico de una base de conocimiento vectorial; y un modelo de lenguaje, encargado de generar respuestas a la pregunta y contexto recogidos.

Este diseño permite automatizar la interpretación de alertas y proponer respuestas adaptadas. El sistema busca complementarlas mediante una capa adicional de inteligencia generativa basada en el enfoque RAG.

El flujo de trabajo del sistema descrito en la Figura 1 muestra cómo se gestionan los eventos de seguridad desde su detección inicial hasta la generación de una respuesta accionable. El proceso comienza en paralelo con la recepción de la consulta del usuario (1) y la recogida de los últimos logs (2), seguida de la generación de alertas en el SIEM (3) y el envío de alertas al módulo RAG (4). A partir de ahí se realiza la construcción de contexto adicional (5) y la generación de respuesta con el LLM (6), tras lo cual el sistema finalmente entrega la clasificación y recomendaciones al usuario (7).

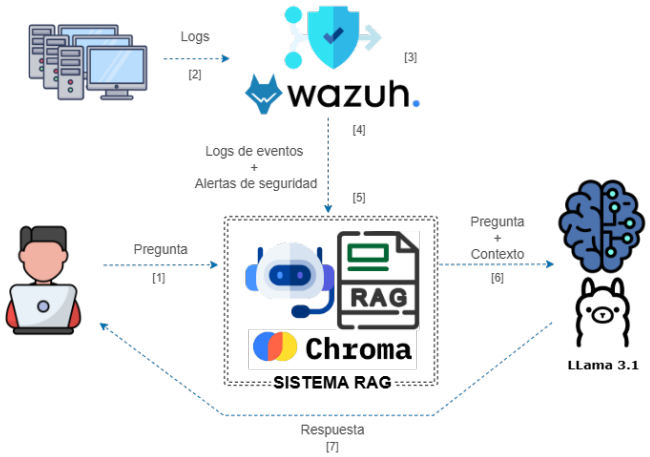


Figura 1. Flujo de operación del sistema.

IV. Implementación de la tecnología LLM-RAG

IV-A. Selección y procesamiento de fuentes

Para que el sistema proporcione análisis precisos y contextualizados, se ha construido una base de conocimiento a partir de documentos reconocidos internacionalmente en ciberseguridad. Se priorizaron estándares, marcos y catálogos elaborados por entidades de referencia, ya que ofrecen una visión estructurada de las Tácticas, Técnicas y Procedimientos (TTPs) y directrices defensivas aplicables a entornos reales.

Cada recurso fue procesado individualmente con el objetivo de convertir las fuentes originales en bloques de texto limpios, segmentados y posteriormente vectorizados, que pudieran integrarse en una base de datos vectorial (ChromaDB [22]) y recuperarse posteriormente según su similitud semántica.

IV-B. Metodología de Procesamiento y Generación de Conocimiento

Para todas las fuentes incluidas en la base de datos se ha aplicado un flujo de procesamiento común, compuesto por las siguientes etapas: *recolección de documentos*, *segmentación en textos*, *procesamiento semántico* y *generación de embeddings*. Aunque las fases iniciales varían según la naturaleza de cada recurso, en todos los casos se ha utilizado el tokenizador y el generador de embeddings del modelo LLaMA 3.1:8B, garantizando así una integración homogénea con el sistema, dado que este mismo modelo se emplea posteriormente en la fase de generación de respuestas.

IV-B1. Recolección de Documentos: Se recopilaron recursos oficiales procedentes de NIST, MITRE ATT&CK, MITRE D3FEND, CAPEC y CWE en sus formatos originales (PDF, JSON o CSV).

IV-B2. Segmentación en Textos: Los documentos fueron transformados y segmentados en unidades textuales adecuadas para su vectorización posterior. Dependiendo de la fuente, se aplicaron las siguientes estrategias:

- **PDF (NIST):** conversiones a texto plano y fragmentación en bloques de 1000 caracteres con un solapamiento de 200 caracteres.
- **JSON (ATT&CK, CAPEC, CWE):** generación de un documento por entrada.

- **CSV convertido a JSON (D3FEND):** división registro a registro, creando un bloque textual por técnica defensiva.

IV-B3. Procesamiento Semántico: Cada fragmento o documento fue normalizado y estructurado para mantener consistencia semántica en toda la base de datos. Se seleccionaron únicamente los campos y documentos más relevantes para cada recurso, garantizando un contenido conciso y representativo:

- **NIST** [5]: Se procesaron los documentos oficiales SP800-53 [23], SP800-61 [24] y SP800-92 [25]
- **MITRE ATT&CK** [4]: Se extrajeron todas las entradas de tipo *attack-pattern*, generando un documento independiente por táctica o técnica con su nombre y descripción.
- **MITRE D3FEND** [26]: Cada registro incluye tácticas asociadas, nombre de la defensa y su definición.
- **CAPEC** [27]: Cada documento incluye: identificador, nombre, severidad típica, estado, descripción y mitigaciones.
- **CWE** [28]: Cada documento incluye: identificador, nombre, estado, probabilidad de explotación, descripción y patrones relacionados.

IV-B4. Generación de Embeddings: Finalmente, todos los textos normalizados fueron procesados mediante el modelo LLaMA 3.1:8B para la obtención de embeddings de alta dimensión. Estos vectores constituyen la base del motor de recuperación semántica del sistema y permiten consultas contextualizadas con conocimiento técnico verificable.

IV-C. Pipeline de recuperación de información relevante

Para recuperar el contexto relevante dada una pregunta del usuario, el sistema transforma cada evento registrado en el SIEM en su representación vectorial empleando el mismo modelo de embeddings utilizado para indexar las fuentes en ChromaDB, compartiendo así el mismo espacio semántico que la base de conocimiento. A partir del vector de entrada v_i, el sistema busca los documentos más próximos para identificar fragmentos conceptualmente relacionados con las alertas recientes, recuperando los k documentos más cercanos (en esta implementación, $k = 5$) según un criterio de similitud basado en la distancia coseno, que mide el ángulo entre dos vectores normalizados.

$$\text{sim}(A, B) = \frac{A \cdot B}{\|A\| \, \|B\|}$$

Los resultados se ordenan por similitud y los fragmentos más relevantes se concatenan en un bloque contextual que acompaña a los logs y la pregunta del usuario en la consulta al LLM, favoreciendo el uso de información directamente relacionada con la alerta y reduciendo alucinaciones.

IV-D. Construcción del prompt y modelo LLM

Para implementar el sistema se emplea LLaMA 3.1:8B, utilizando sus versiones de embeddings y generación de texto como base para representar fragmentos y producir respuestas. Su elección se debe a: su eficiencia y ejecución local, que permite un buen rendimiento sin depender de servicios externos; a su naturaleza abierta y auditable, adecuada para entornos donde la privacidad es crítica; y a su capacidad multilingüe, necesaria para integrar consultas en español, logs en inglés y documentación técnica diversa. Además, al ser un modelo de código abierto, ofrece mayor flexibilidad y adaptación frente a alternativas propietarias.

El prompt de sistema actúa como interfaz entre la consulta de usuario, los datos del SIEM y el modelo LLM, especificando de forma explícita cómo deben interpretarse los logs, qué relevancia otorgar al contexto recuperado y cómo estructurar la respuesta. La metodología de diseño se basa en estándares de *prompt engineering*: definir claramente el rol del modelo, sus fuentes de información y el formato de salida esperado.

El prompt se compone de tres bloques diferenciados:

- Logs, conteniendo los eventos brutos procedentes del SIEM (Wazuh).
- Contexto, formado por los fragmentos más relevantes recuperados de ChromaDB mediante búsqueda semántica.
- Pregunta, formula la tarea de análisis o diagnóstico.

A continuación, se incluye el prompt utilizado en el sistema, adaptado para LLaMA 3.1. El prompt se diseña para minimizar alucinaciones y garantizar la trazabilidad entre logs y recomendaciones, usando el contexto solo como refuerzo sobre evidencias reales. Además, fija una plantilla de salida que estandariza las respuestas y facilita su interpretación por parte de los analistas.

```
Eres un asistente experto en ciberseguridad,
    especializado en la detección de amenazas
    y respuesta a incidentes.

Se te proporcionan los siguientes datos:
1. LOGS: Información extraída del sistema SIEM
    .
2. CONTEXTO: Información relevante obtenida de
     bases de datos y documentación de
    seguridad.
3. PREGUNTA: Un usuario solicita un análisis
    de seguridad y recomendaciones de mitigaci
    ón.

Tu tarea es:
- Analiza los logs de forma autónoma,
    identificando cualquier posible actividad
    sospechosa.
- Utiliza el contexto únicamente para reforzar
     tus hallazgos, vinculándolos con
    evidencias documentadas.
- Responde utilizando exactamente la
    estructura siguiente:

1. Detección de amenazas: Sí/No + descripción
2. Nivel de riesgo: Bajo/Medio/Alto/Crítico
3. Explicación técnica: Análisis detallado
    basado en los logs y contexto
4. Cursos de Acción (COAs): Recomendaciones
    específicas de mitigación

Asegúrate de que la respuesta sea clara,
    verificable y útil para la toma de
    decisiones de un analista de seguridad.
```

El diseño de este prompt busca minimizar las alucinaciones y garantizar la trazabilidad entre los logs y las recomendaciones. Al delimitar explícitamente el uso del contexto en un

papel de refuerzo, se orienta al modelo a fundamentar sus salidas en las evidencias observadas.

V. Escenarios de validación

Con el fin de evaluar el comportamiento del sistema, se han diseñado dos escenarios de validación que permiten analizar su desempeño en distintos niveles de complejidad y realismo. Ambos reproducen situaciones de ciberseguridad en las que se generan y procesan eventos a través del SIEM, lo que permite examinar la capacidad del modelo LLM-RAG para detectar, contextualizar y proponer respuestas adecuadas ante posibles amenazas. Esta sección describe los entornos experimentales y la metodología de validación empleada.

Los experimentos se ejecutaron en un entorno local con una CPU Intel Core i7-14700F, 32 GB de RAM y una GPU NVIDIA GeForce RTX 4070 SUPER. El modelo LLaMA 3.1 8B se ejecutó en formato GGUF mediante Ollama.

V-A. Escenario 1: CALDERA

Este primer escenario tiene como finalidad evaluar el rendimiento en un entorno controlado, donde cada evento y técnica de ataque está previamente caracterizado. El uso de Mitre CALDERA, una herramienta desarrollada para la emulación automatizada de adversarios, garantiza la reproducibilidad y trazabilidad de las acciones maliciosas. Este escenario se desarrolla en una única máquina virtual Windows donde se ejecuta un escenario de emulación de adversarios siguiendo el marco MITRE ATT&CK. Los ataques emulados incluyen las siguientes técnicas:

- T1197 - BITS Jobs: Uso malicioso del servicio BITS de Windows para descargar o ejecutar archivos en segundo plano, facilitando la persistencia del atacante.
- T1040 - Network Sniffing: Captura de tráfico de red para obtener información sensible como credenciales o datos no cifrados.
- T1021.001 - Remote Desktop Protocol: Acceso remoto a sistemas mediante Remote Desktop Protocol, utilizado para movimientos laterales o control total del equipo.

Tras definir algunas reglas de detección adicionales, durante la simulación se registran los eventos clave como modificaciones de cuentas de usuario, ejecución de tareas maliciosas y cambios en las configuraciones del sistema. El escenario de CALDERA, por tanto, representa la validación en condiciones ideales, centrada en la detección de comportamientos maliciosos concretos y en la generación de Courses of Action (COAs) a partir de evidencias directas en los logs, constituyendo la base metodológica sobre la que se comparará el segundo escenario.

V-B. Escenario 2: CTF

El segundo escenario introduce un entorno más complejo y realista, orientado a evaluar la capacidad del sistema en condiciones operativas cercanas a un entorno corporativo. A diferencia del escenario anterior, aquí los eventos provienen de un ataque ejecutado sobre una infraestructura multimáquina, donde se reproducen distintas fases de la cadena de ataque. Este escenario ejecuta un ejercicio de tipo Capture The Flag (CTF) emulando una intrusión interna por parte de un exempleado con acceso previo al entorno, siguiendo ejemplos realizados en proyectos anteriores [29].

El escenario se desarrolla sobre un entorno compuesto por tres máquinas virtuales (una máquina atacante, un host intermedio accesible y un servidor interno), configuradas para simular una red corporativa segmentada. El usuario solo tiene acceso inicial a la máquina atacante, desde la cual debe escalar privilegios y pivotar lateralmente para acceder a otras partes de la red. El entorno, descrito en la Figura 2, incluye servicios como SSH, FTP y aplicaciones web con posibles vulnerabilidades explotables (inyección SQL, movimientos laterales, descifrado de credenciales, etc.).

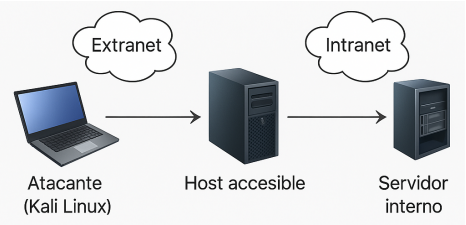


Figura 2. Entorno de ataque para el escenario CTF

En este escenario, los logs generados por Wazuh permiten evaluar cómo el sistema gestiona flujos simultáneos y eventos dispersos, poniendo a prueba tanto la recuperación semántica como la capacidad del LLM para sintetizar información en un entorno más ruidoso y heterogéneo. Además, su carácter práctico lo convierte en un marco útil para fines formativos, al permitir comparar el razonamiento de analistas con las recomendaciones generadas. Junto con el escenario controlado anterior, este caso ofrece una evaluación complementaria que contrasta precisión técnica y adaptabilidad en condiciones realistas.

V-C. Métricas de resultados

La metodología de evaluación se ha diseñado con el propósito de cuantificar el rendimiento y la utilidad práctica del sistema propuesto bajo dos condiciones operativas diferenciadas:

- Zero-Shot: el modelo de lenguaje opera exclusivamente con la información contenida en los logs.
- RAG: el modelo utiliza, además de los logs, información contextual recuperada desde una base de conocimiento estructurada.

Esta comparación permite analizar de forma sistemática el impacto que la integración del componente de recuperación tiene sobre la calidad y consistencia de las respuestas generadas.

En ambos escenarios se aplican las mismas métricas y procedimientos para garantizar la comparabilidad entre resultados. El objetivo es determinar en qué medida la contextualización introducida por RAG mejora la detección, clasificación y respuesta ante incidentes en entornos de distinta complejidad.

Las métricas de evaluación son:

- **Precisión y criticidad**: miden la capacidad del sistema para identificar correctamente la amenaza simulada. Se considera *acierto* cuando la técnica inferida coincide con la ejecutada (ground truth), y *fallo* en caso contrario, incluso si la técnica propuesta es coherente con el contexto.

Caso	Config.	Claridad	Relevancia	Aplicabilidad	Media Likert	Tiempo (s)
C1 (CALDERA)	Sin RAG	3.0	2.5	3.0	3.2	14.96
C1 (CALDERA)	Con RAG	4.5	4.7	4.5	4.6	9.37
C2 (CTF)	Sin RAG	3.0	2.3	3.0	2.8	15.75
C2 (CTF)	Con RAG	4.7	4.8	4.5	4.7	10.41

Tabla I
RESUMEN CUANTITATIVO DE LAS MÉTRICAS MEDIAS OBTENIDAS EN AMBOS ESCENARIOS Y CONFIGURACIONES.

- **Calidad de los COAs**: se evalúan **claridad**, **relevancia** y **aplicabilidad** mediante una escala de Likert (1–5). La puntuación final corresponde a la media de estas tres dimensiones.
- **Tiempo de respuesta**: tiempo total, en segundos, desde la recepción del log hasta la generación de la respuesta. Permite comparar la eficiencia del pipeline con y sin RAG.
- **Nivel de contextualización**: analiza si los documentos recuperados son relevantes y se incorporan adecuadamente en la respuesta. La validación se realiza mediante el análisis de trazas en LangSmith [30].

La precisión y criticidad se contrastan con el conocimiento previo de cada caso: TTPs definidas en CALDERA y trazabilidad completa del ataque en el CTF. En conjunto, estas métricas combinan aspectos cualitativos y cuantitativos, ofreciendo una evaluación equilibrada del rendimiento y utilidad del sistema para apoyar la toma de decisiones en ciberseguridad.

Para garantizar una evaluación consistente, cada configuración se aplicó sobre un conjunto reducido de casos en ambos escenarios, y las salidas fueron valoradas por dos evaluadores independientes, con conocimientos amplios en el ámbito de la ciberseguridad. Las diferencias entre evaluaciones fueron bajas (desviación estándar inferior a 0.5 en las métricas Likert), lo que indica consistencia en las puntuaciones. Dado el tamaño muestral limitado, no se incluyen pruebas estadísticas inferenciales (p-valores), si bien las medias y dispersión observadas permiten valorar la estabilidad de los resultados.

VI. ANÁLISIS DE RESULTADOS

Desde el punto de vista cualitativo, la evaluación mediante la escala de Likert muestra que el enfoque RAG supera al modelo base en claridad, relevancia y aplicabilidad. Los COAs generados con contextualización resultan más precisos y coherentes, mientras que las soluciones *zero-shot* tienden a ofrecer recomendaciones genéricas y poco técnicas. Este comportamiento confirma el efecto positivo de RAG asumido en la hipótesis H_2.

No obstante, esta mejora en las mitigaciones no se traduce en una mejor identificación de TTPs. En el escenario de CALDERA ninguna configuración detecta correctamente las técnicas ejecutadas, aunque el modelo con RAG menciona TTPs coherentes con ataques de inyección de *payload* (T1055). En el escenario CTF, las acciones detalladas (NTDS.dit, LSASS, procdump, Mimikatz) reflejan fielmente el ataque, pero no se vinculan explícitamente con técnicas concretas. Por lo tanto, no podemos realmente afirmar una mejora en la hipótesis H_1, pues si la recuperación supone una mejora en la clasificación, no se observa de forma estable. En ambos casos, el riesgo se clasifica sistemáticamente como alto; si bien RAG ofrece justificaciones técnicas más sólidas y mejor alineadas con los eventos observados.

En el plano cuantitativo, buscando evaluar la hipótesis H_3, el tiempo de respuesta con RAG es menor pese al proceso de recuperación previo, posiblemente porque el contexto reduce el esfuerzo de razonamiento del modelo y agiliza la generación de la respuesta final.

Respecto a las fuentes recuperadas, la mayoría procede de NIST 800-53, lo que explica la notable mejora en los COAs. Aunque también aparecen tácticas ATT&CK, su relevancia es menor, probablemente debido a la diferencia entre el lenguaje de las alertas del SIEM y la forma en que se describen las TTPs. En cambio, la redacción de los controles de NIST se aproxima más al vocabulario de los eventos correlacionados, facilitando su recuperación.

En conjunto, los resultados confirman que la recuperación aumentada aporta un valor claro en contextualización y utilidad práctica, pero también evidencian que la detección de TTPs requiere una mejor alineación semántica entre logs y base de conocimiento. Futuras mejoras deberán centrarse en este acoplamiento para aprovechar plenamente el potencial del enfoque LLM-RAG.

VII. DISCUSIÓN Y LÍNEAS FUTURAS

Los resultados muestran que la integración de LLMs con RAG es especialmente adecuada para la respuesta ante ciberamenazas, al reducir alucinaciones gracias al uso de evidencias verificadas y facilitar la adaptación a nuevas amenazas gracias a bases de conocimiento dinámicas.

La IA generativa está transformando el trabajo de los analistas al automatizar tareas de análisis y priorización, reduciendo carga cognitiva y acelerando la detección. En este contexto, el enfoque LLM–RAG responde tanto a un reto técnico como a una necesidad creciente del ecosistema de ciberseguridad.

Desde un punto de vista ético y operativo, la transparencia y auditabilidad de las decisiones automáticas son esenciales. El uso de modelos abiertos contribuye a la trazabilidad y sostenibilidad, aunque persisten riesgos como el *prompt injection*, el *data leakage* o la manipulación del contexto, que requieren mecanismos de mitigación específicos.

Una limitación del estudio es el uso exclusivo de LLaMA 3.1 8B; será necesario evaluar modelos de mayor tamaño y alternativas recientes para analizar su impacto en la identificación de TTPs y en el rendimiento del sistema. La evaluación estadística también queda limitada por el reducido número de casos y evaluadores, por lo que trabajos futuros deberán ampliar la muestra y aplicar técnicas estadísticas básicas para valorar la significancia de las diferencias observadas.

Las líneas futuras incluyen ampliar la base de conocimiento con fuentes multilingües, mejorar el pipeline de recuperación

mediante embeddings especializados y técnicas de re-ranqueo, y validar el sistema en entornos reales de operación (SOCs).

En conjunto, los resultados evidencian que la combinación de LLMs y RAG aporta mejoras claras en contextualización y utilidad práctica, y que reforzar su fiabilidad será clave para una automatización responsable en ciberseguridad.

Agradecimientos

Esta investigación es resultado del Proyecto Estratégico SPEDIA (PE_123/23), desarrollado en el marco del convenio de colaboración firmado entre el Instituto Nacional de Ciberseguridad (INCIBE) y la Universidad Politécnica de Madrid. Esta iniciativa se enmarca en el Plan de Recuperación, Transformación y Resiliencia, financiado por la Unión Europea (Next Generation). Además, parcialmente financiado por el Ministerio de Asuntos Económicos y Transformación Digital del Gobierno de España y por NextGenerationEU (Plan de Recuperación, Transformación y Resiliencia – PRTR), en el marco del proyecto del Programa UNICO I+D 6G, TSI-064100-2022-7.

Referencias

[1] G. d. J. C. da Silva and C. B. Westphall, "A survey of large language models in cybersecurity," *arXiv preprint arXiv:2402.16968*, 2024.

[2] V. Rawte, A. Sheth, and A. Das, "A survey of hallucination in large foundation models," *arXiv preprint arXiv:2309.05922*, 2023.

[3] S. S. Kumar, A. K. M. A. Khan, I. A. Banday, M. Gada, and V. V. Shanbhag, "Overcoming llm challenges using rag-driven precision in coffee leaf disease remediation," in *2024 ICETCS*. IEEE, 2024, pp. 1–6.

[4] MITRE, "MITRE ATT&CK Framework," https://attack.mitre.org/, accedido en 2025.

[5] U.S. Department of Commerce, "NIST," https://www.nist.gov/, 2025, accedido en 2025.

[6] J. Zhang, H. Bu, H. Wen, Y. Liu, H. Fei, R. Xi, L. Li, Y. Yang, H. Zhu, and D. Meng, "When llms meet cybersecurity: A systematic literature review," *Cybersecurity*, vol. 8, no. 1, p. 55, 2025.

[7] F. N. Motlagh, M. Hajizadeh, M. Majd, P. Najafi, F. Cheng, and C. Meinel, "Large language models in cybersecurity: State-of-the-art," *arXiv preprint arXiv:2402.00891*, 2024.

[8] Y. Yigit, W. J. Buchanan, M. G. Tehrani, and L. Maglaras, "Review of generative ai methods in cybersecurity," *arXiv preprint arXiv:2403.08701*, 2024.

[9] A. Shestov, R. Levichev, R. Mussabayev, E. Maslov, P. Zadorozhny, A. Cheshkov, R. Mussabayev, A. Toleu, G. Tolegen, and A. Krassovitskiy, "Finetuning large language models for vulnerability detection," *IEEE Access*, 2025.

[10] Z. Li, S. Dutta, and M. Naik, "Iris: Llm-assisted static analysis for detecting security vulnerabilities," *arXiv preprint arXiv:2405.17238*, 2024.

[11] M. Sultana, A. Taylor, L. Li, and S. Majumdar, "Towards evaluation and understanding of large language models for cyber operation automation," in *2023 IEEE Conference on CNS*. IEEE, 2023, pp. 1–6.

[12] M. M. Yamin, E. Hashmi, M. Ullah, and B. Katt, "Applications of llms for generating cyber security exercise scenarios," *IEEE Access*, 2024.

[13] (2024) Ingeniería de prompts vs ajuste fino vs rag. MyScale. Accedido en 2025. [Online]. Available: https://myscale.com/blog/prompt-engineering-vs-fine-tuning-vs-rag/

[14] Y. Singh, N. Patel, and S. K. Shandilya, "Enhancing security operations center efficiency throughmulti-model integration of large language models and siem systems," 2024.

[15] M. Kaheh, D. K. Kholgh, and P. Kostakos, "Cyber sentinel: Exploring conversational agents in streamlining security tasks with gpt-4," *arXiv preprint arXiv:2309.16422*, 2023.

[16] M. Lempinen, A. Juntunen, and E. Pyyny, "Chatbot for assessing system security with openai gpt-3.5," B.S. thesis, M. Lempinen; A. Juntunen; E. Pyyny, 2023.

[17] M. Simoni, A. Saracino, V. P, and M. Conti, "Morse: Bridging the gap in cybersecurity expertise with retrieval augmented generation," in *Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing*, 2025.

[18] J. Pan, W. S. Liang, and Y. Yidi, "Raglog: Log anomaly detection using retrieval augmented generation," in *2024 IEEE World Forum on Public Safety Technology (WFPST)*. IEEE, 2024, pp. 169–174.

[19] (2026) Wazuh. Wazuh. Accedido en 2025. [Online]. Available: https://wazuh.com/

[20] A. G. et al., "The llama 3 herd of models," 2024. [Online]. Available: https://arxiv.org/abs/2407.21783

[21] MITRE, "Caldera," https://caldera.mitre.org/, accedido en 2025.

[22] Chroma, "ChromaDB: Open-Source Embedding Database," https://www.trychroma.com/, 2025, accedido en 2025.

[23] U.S. Department of Commerce, "Security and privacy controls for information systems and organizations," https://doi.org/10.6028/NIST.SP.800-53r5, 2020.

[24] ——, "Incident response recommendations and considerations for cybersecurity risk management," https://doi.org/10.6028/NIST.SP.800-61r3, 2025.

[25] M. Souppaya and K. Kent, "Guide to computer security log management," https://doi.org/10.6028/NIST.SP.800-92, 2006.

[26] MITRE, "D3fend," https://d3fend.mitre.org/, 2025, accedido en 2025.

[27] ——, "Capec," https://capec.mitre.org/, 2025, accedido en 2025.

[28] ——, "Cwe: Common weakness enumeration," https://cwe.mitre.org/, 2025, accedido en 2025.

[29] A. M. X. L.-N. M. A.-C. V. A. V. David Alvarez, Luis Perez, "Diseno y generación de un conjunto de datos para entrenamiento de modelos de prevencion y deteccion de amenazas internas: Spedia dataset," in *Actas de las X Jornadas Nacionales de Investigaci on en Ciberseguridad*, 2025.

[30] LangChain, "Langsmith: Platform for llm debugging, evaluation, and monitoring," https://www.langchain.com/langsmith, 2025, accedido en 2025.

A Side-Channel Protected and High-Performance Hardware Implementation for EdDSA25519

Pablo Navarro-Torrero, Eros Camacho-Ruiz, Macarena C. Martínez-Rodríguez, Piedad Brox
Instituto de Microelectrónica de Sevilla (IMSE-CNM), CSIC / Universidad de Sevilla, Sevilla, Spain
{navarro, camacho, macarena, brox}@imse-cnm.csic.es

***Abstract*—This is an extended abstract from the article published in [1]. The original paper presents a high-performance and secure hardware implementation of the Edwards-Curve Digital Signature Algorithm (EdDSA25519). By employing fixed-base signed multi-comb and k-ary algorithms for scalar multiplication, the design achieves significant performance gains of 307% in key generation, 253% in signature generation, and 48% in signature verification compared to the fastest existing hardware implementations. Furthermore, when benchmarked against the software-based OpenSSL implementation, the hardware design demonstrates a remarkable 1000% to 2200% improvement in timing performance. In addition to its high speed, the implementation integrates robust countermeasures against Side-Channel Attacks (SCA). The effectiveness of these security features is validated through Test Vector Leakage Assessment (TVLA), which confirms increased resistance to both Simple Power Analysis (SPA) and Differential Power Analysis (DPA). This work, therefore, delivers a hardware-based solution that offers both high performance and enhanced security for modern cryptographic applications.**

***Index Terms*—Elliptic Curve Cryptography, EdDSA25519, Hardware Implementation, Side-Channel Attacks, TVLA**

I. Introduction

Elliptic Curve Cryptography (ECC) is a cornerstone of modern public-key cryptography, with the Edwards-Curve Digital Signature Algorithm (EdDSA) [2], particularly Ed25519 [3], being widely adopted for its high performance and security. However, while numerous hardware implementations exist [4], [5], many prioritize speed over security and lack rigorous validation against Side-Channel Attacks (SCAs) [6]. There is a notable gap in the literature regarding hardware designs that are systematically evaluated using standard methodologies like Test Vector Leakage Assessment (TVLA) [7], with only a few exceptions that focus on incomplete components or do not publish design details [8], [9].

This work addresses this gap by presenting a high-performance and secure hardware implementation of EdDSA25519. Our main contributions are:

- A novel architecture leveraging the fixed-base signed multi-comb algorithm [10] for scalar multiplication, achieving speedups of 10× to 22× over the OpenSSL library and outperforming state-of-the-art hardware accelerators.
- The integration of comprehensive SCA countermeasures, including scalar blinding, randomized projective coordinates, and a noise engine to protect against Simple and Differential Power Analysis (SPA/DPA).
- A systematic security validation using TVLA, providing one of the first public and detailed assessments for a complete EdDSA25519 hardware implementation.

All design files and evaluation data are made publicly available to foster reproducibility and further research [11].

II. Mathematical Background

This work implements EdDSA as specified in RFC 8032 [12], utilizing the twisted Edwards curve Ed25519. The curve is defined over the prime field $\mathbb{GF}(p = 2^{255} - 19)$ by the equation $ax^2 + y^2 = 1 + dx^2y^2$. Where $a = -1 \pmod p$ and $d = -\frac{121665}{121666} \pmod p$. EdDSA offers significant performance and security benefits, partly due to its complete and unified addition law, which avoids exceptional points and strengthens resistance to certain side-channel attacks.

Our design's performance is achieved through a hierarchical optimization strategy, as illustrated in Fig. 1, targeting every computational layer from basic field arithmetic up to the core scalar multiplication.

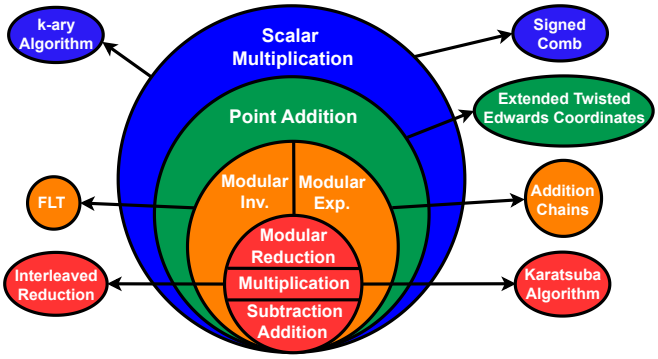


Fig. 1. ECC Operations Hierarchy and Optimization Algorithms.

A. Field-Level Optimizations

At the lowest level, multi-precision integer multiplication, the most frequent operation, is accelerated using the **Karatsuba algorithm** [13], which reduces the number of expensive multiplications at the cost of cheaper additions. Modular reduction is specialized for the fixed prime $p = 2^{255} - 19$, using **interleaved reduction** techniques that exploit the prime's specific structure [14]. Finally, modular inversion is implemented using modular exponentiation via **Fermat's Little Theorem** ($x^{-1} \equiv x^{p-2} \pmod p$). The exponentiations themselves are optimized using pre-calculated **Addition Chains** to minimize the total number of multiplications required.

B. Point-Level Optimizations

To accelerate point addition and doubling, our architecture avoids affine coordinates, whose group law requires a costly modular inversion for every operation. Instead, we use **Extended Twisted Edwards Coordinates** (X, Y, Z, T). This projective system replaces the single modular inversion

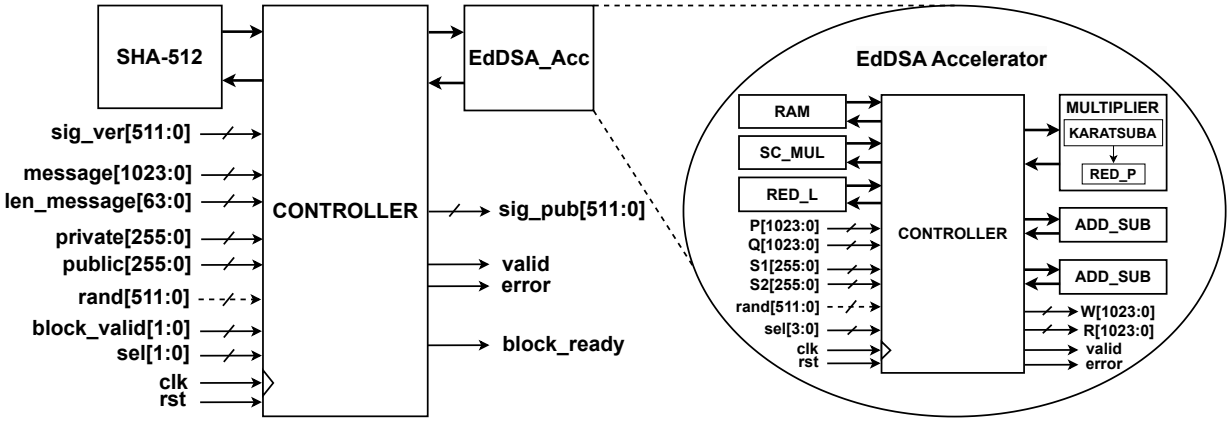


Fig. 2. Block diagram of the unified EdDSA25519 Cryptocore, showing the main components and data paths.

with a sequence of ten field multiplications and seven additions/subtractions. While more field operations are needed per point addition, they are collectively far cheaper than a single inversion. An inversion is only required once at the very end of a scalar multiplication to convert the final point back to its affine representation.

C. Scalar Multiplication Optimizations

Scalar multiplication is the most computationally intensive part of EdDSA. We employ two distinct algorithms to maximize performance across all operations:

- **Signed Multi-Comb Algorithm:** For key generation and signature generation, the scalar multiplication is performed on a fixed base point $\mathcal{G}$. We leverage the high-performance fixed-base signed multi-comb algorithm [10]. This method relies on a large precomputed table of multiples of $\mathcal{G}$, allowing it to combine multiple bits of the scalar in a single step. This dramatically reduces the total number of point operations, making it significantly faster than traditional methods like the Montgomery ladder for fixed-base scenarios.
- **k-ary Algorithm:** During signature verification, a double-scalar multiplication is required. One part uses the fixed base point $\mathcal{G}$, while the other uses the public key, a variable point. For the variable-base component, we use the standard k-ary (windowed) method, which offers a good balance between performance and precomputation cost for non-fixed inputs. These two scalar multiplications are performed in parallel to optimize the verification process.

III. Hardware Implementation

Our EdDSA25519 hardware implementation is a unified cryptographic core designed for high performance and configurable security. The architecture, shown in Fig. 2, consists of a central Logic Controller that orchestrates a custom EdDSA Accelerator and a hash function SHA-512 module. The design process involved a detailed exploration of key architectural trade-offs to optimize performance at every level.

A. Architectural Design and Optimization

The core of our design is the EdDSA Accelerator, which was optimized based on the following key decisions:

- **Karatsuba Multiplier:** We evaluated multipliers with recursion levels from one to four. A two-level recursive Karatsuba multiplier was selected as it offered the best balance between resource consumption (144 DSPs), latency (3 cycles), and maximum operating frequency. A pipelining strategy allows subsequent multiplications to be issued every clock cycle.
- **Pipelined Point Addition:** The point addition unit, which uses the extended twisted Edwards coordinates, is heavily pipelined. It leverages the Karatsuba multiplier and two parallel adder/subtractor modules to complete a single point addition in just 17 clock cycles. The schedule is further optimized to support a double-point addition in only 22 cycles, significantly accelerating the signature verification process.
- **Scalar Multiplication Parameters:** The parameters for the scalar multiplication algorithms were chosen to maximize performance. For the variable-base k-ary algorithm, a 4-bit window size ($k = 4$) was found to be optimal, minimizing the operation count without excessive memory usage. For the fixed-base signed multi-comb algorithm, we adopted the parameters $(n = 3, t = 5, s = 17)$ [15], which require 67 point operations per scalar multiplication.

B. Integrated Side-Channel Countermeasures

A primary contribution of this work is the integration and validation of a comprehensive suite of SCA countermeasures. These are layered to provide increasing levels of security.

- **Noise Engine:** To hide power consumption, the hardware modules not in active use (e.g., the SHA-512 core during scalar multiplication) are tasked with performing pseudo-random operations. This generates a constant level of background noise, making it harder to isolate the power traces of sensitive computations.
- **Randomized Projective Coordinates:** Before and after each point addition/doubling, the projective coordinates (X, Y, Z, T) are randomized by multiplying them with a random non-zero scalar $\lambda \in \mathbb{GF}(p)$. This ensures that intermediate values are different in every execution, effectively masking the data and thwarting DPA attacks. This adds 4 clock cycles per point operation.

TABLE I
OPERATIONAL PERFORMANCE AND POWER CONSUMPTION OF EDDSA25519 CRYPTOCORE

SCA	Operation	Cycles	XC7Z020 @ 70 MHz				ZCU104 @ 280-320 MHz			
			T [μs]	OP/s	P [mW]	E [μJ]	T [μs]	OP/s	P [mW]	E [μJ]
+	Key Gen.	2,738	39	25,566	571	22	9	116,874	2,333	21
	Sign	3,147	45	22,243	532	24	10	101,684	2,193	22
	Verify	9,237	132	7,578	730	96	29	34,643	2,934	85
++	Key Gen.	3,003	43	23,310	1,082	47	11	93,240	3,469	38
	Sign	3,416	49	20,492	1,095	54	12	81,967	3,520	42
	Verify	9,245	132	7,572	1,138	150	33	30,287	3,623	120
+++	Key Gen.	4,167	60	16,799	1,166	70	15	67,195	3,625	54
	Sign	4,613	66	15,175	1,174	77	16	60,698	3,659	59
	Verify	9,519	136	7,354	1,193	162	34	29,415	3,692	126

- **Scalar Blinding:** To prevent attacks on the static private key, the scalar k is masked before each use by adding a 128-bit random multiple of the group order L, yielding $k' = k + r \cdot L$. Since $[k]\mathcal{P} = [k']\mathcal{P}$, the result is identical, but the scalar processed by the hardware is randomized for each signature. This requires adjusting the comb parameters $(n = 4, t = 4, s = 24)$ [15] to handle the larger scalar size and result in a total of 119 point operations per scalar multiplication.
- **Modular Reduction Masking:** Similarly, the 512-bit hash outputs are masked with a 256-bit random multiple of L before being reduced modulo L.

C. Unified Cryptocore and Security Levels

The final cryptocore integrates the EdDSA Accelerator and SHA-512 module into a single, cohesive unit. To systematically evaluate the trade-offs between performance and security, we developed three versions of the core, each with a different level of SCA protection:

- **SCA Security (+):** The baseline version, designed for maximum performance. It ensures constant-time execution to resist timing attacks and basic SPA.
- **SCA Security (++):** This version adds the noise engine and randomized projective coordinates on top of the baseline protections, providing strong resistance against DPA.
- **SCA Security (+++):** The most robust version, incorporating all countermeasures, including scalar blinding and modular reduction masking, for the highest level of security against DPA.

This configurable design allows for a direct comparison of the overhead introduced by each set of countermeasures, providing a clear picture of the performance-security landscape for EdDSA hardware.

IV. RESULTS AND STATE-OF-THE-ART COMPARISON

The proposed EdDSA25519 cryptocore was implemented on two distinct FPGA platforms: a Xilinx Zynq-7000 (XC7Z020), representing a mid-range device, and a Zynq UltraScale+ (ZCU104) for high-performance applications.

A. Implementation and Performance Results

Table II summarizes the hardware resource utilization for our three security configurations. As expected, enabling stronger SCA countermeasures incurs a resource overhead. On the ZCU104, moving from the baseline (+) to the fully protected (+++) version increases the logic utilization (LUTs) by 10%. The table also highlights the significant difference in achievable clock frequency, with the ZCU104 reaching up to 320 MHz, compared to 70 MHz on the Zynq-7000 device.

TABLE II
RESOURCE UTILIZATION OF EDDSA25519 CRYPTOCORE

Platform	SCA	LUTs	FFs	DSPs	BRAM	Freq. [MHz]
XC7Z020	(+)	47,241	18,228	144	26.5	70
	(++)	51,257	18,323			
	(+++)	52,949	18,514			
ZCU104	(+)	48,001	19,338	144	11	320
	(++)	51,933	19,419			280
	(+++)	56,006	20,355			280

The operational performance metrics are detailed in Table I. The data quantifies the trade-off between security and speed. The addition of scalar blinding in the (+++) version increases the clock cycle latency for key generation and signing by 30-40% compared to the baseline. Despite this overhead, the throughput remains exceptionally high across all configurations.

B. Comparison against Software and Hardware

When benchmarked against a standard software implementation (OpenSSL v3.0.2), our hardware accelerator provides a dramatic speedup. On the ZCU104, the fastest (+) configuration is **11× to 22× faster than software**. Even our most secure (+++) design maintains a significant lead with a speedup of nearly 10×.

Critically, our design sets a new performance benchmark for EdDSA2519 hardware. On the widely-used XC7Z020 platform, our baseline (+) implementation (see Table I) is **307% faster in key generation, 253% faster in signing, and 48% faster in verification** than the best previously reported result on the same device [14].

V. SIDE-CHANNEL EVALUATION

To validate the effectiveness of our countermeasures, we conducted a rigorous SCA on a SAKURA-X hardware evaluation board. Power consumption traces were captured during signature generation for all three security configurations. A TVLA, based on Welch's t-test, was performed to statistically detect information leakage, using a corrected confidence threshold of $|t| \geq 6.22$.

The evaluation revealed a clear progression of security. The baseline (+) version, protected only by constant-time execution, exhibited significant leakage across all operations. The (++) version, which adds a noise engine and randomized coordinates, successfully eliminated leakage during hash computations but still showed vulnerabilities during the core scalar multiplication. The final (+++) configuration, which incorporates scalar blinding, demonstrated strong resistance.

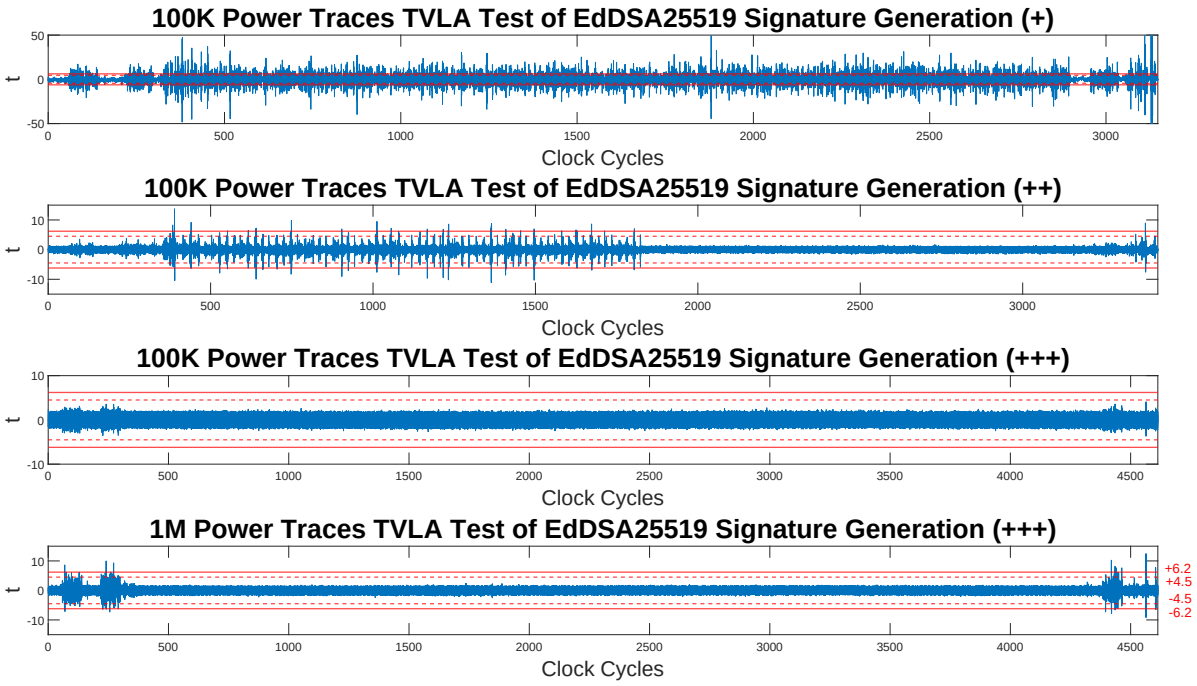


Fig. 3. TVLA Test comparison between different EdDSA25519 implementations during signature generation

As shown in Fig. 3, with 100,000 power traces, the t-statistic remains well below the leakage threshold, indicating no detectable information leakage. This confirms the effectiveness of our complete countermeasure suite against DPA attacks under typical assessment conditions. A more extensive test with one million traces did reveal minor, residual leakage, confirming that while not perfect, the countermeasures drastically increase the attacker's required effort.

VI. Conclusion

This paper presented a high-performance, secure hardware implementation of EdDSA25519. By leveraging a signed multi-comb scalar multiplication architecture, our design outperforms prior state-of-the-art hardware and standard software libraries. Crucially, we integrated a comprehensive suite of layered SCA countermeasures and provided one of the first systematic TVLA evaluations for a complete EdDSA hardware core. Our design demonstrated increased resistance to SCA and established a new benchmark for cryptographic hardware that balances speed with security.

Acknowledgements

This work was supported in part by the QUBIP Project through the European Union (EU) Horizon Europe Research and Innovation Program under Grant 101119746.

References

[1] P. Navarro-Torrero, E. Camacho-Ruiz, M. C. Martínez-Rodríguez, and P. Brox-Jiménez, "A side-channel protected and high-performance hardware implementation for eddsa25519," *IEEE Access*, vol. 13, pp. 115 748–115 765, 2025, doi: 10.1109/ACCESS.2025.3584696.

[2] D. J. Bernstein, P. Birkner, M. Joye, T. Lange, and C. Peters, "Twisted edwards curves," in *Progress in Cryptology – AFRICACRYPT 2008*, S. Vaudenay, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 389–405, doi: 10.1007/978-3-540-68164-9_26.

[3] D. J. Bernstein, "Curve25519: new diffie-hellman speed records," in *Public Key Cryptography-PKC 2006: 9th International Conference on Theory and Practice in Public-Key Cryptography, New York, NY, USA, April 24-26, 2006. Proceedings 9*. Springer, 2006, pp. 207–228, doi: 10.1007/11745853_14.

[4] B. Kieu-Do-Nguyen, C. Pham-Quoc, N.-T. Tran, C.-K. Pham, and T.-T. Hoang, "Low-cost area-efficient fpga-based multi-functional ecdsa/eddsa," *Cryptography*, vol. 6, no. 2, 2022, doi: 10.3390/cryptography6020025. [Online]. Available: https://www.mdpi.com/2410-387X/6/2/25

[5] F. Turan and I. Verbauwhede, "Compact and Flexible FPGA Implementation of Ed25519 and X25519," *ACM Trans. Embed. Comput. Syst.*, vol. 18, no. 3, apr 2019, doi: 10.1145/3312742.

[6] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO'99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19*. Springer, 1999, pp. 388–397, doi: 10.1007/3-540-48405-1_25.

[7] B. J. Gilbert Goodwill, J. Jaffe, P. Rohatgi *et al.*, "A testing methodology for side-channel resistance validation," in *NIST non-invasive attack testing workshop*, vol. 7, 2011, pp. 115–136.

[8] P. Ravi, B. Jungk, and S. Bhasin, "Single trace electromagnetic side-channel attacks on fpga implementation of elliptic curve cryptography," in *2019 Joint International Symposium on Electromagnetic Compatibility, Sapporo and Asia-Pacific International Symposium on Electromagnetic Compatibility (EMC Sapporo/APEMC)*. IEEE, 2019, pp. 1–4, doi: 10.23919/EMCSapporo/APEMC44270.2019.9320846.

[9] Ł. Chmielewski, P. M. C. Massolino, J. Vliegen, L. Batina, and N. Mentens, "Completing the complete ecc formulae with countermeasures," *Journal of Low Power Electronics and Applications*, vol. 7, no. 1, p. 3, 2017, doi: 10.3390/jlpea7010003.

[10] M. Hamburg, "Fast and compact elliptic-curve cryptography," *Cryptology ePrint Archive*, 2012. [Online]. Available: https://eprint.iacr.org/2012/309

[11] P. N. Torrero, "EDDSA_HPCM: Side-channel protected and high performance hw implementation for eddsa25519," https://gitlab.com/hwsec/eddsa_hpcm, 2025, accessed: 2025-03-20.

[12] S. Josefsson and I. Liusvaara, "Edwards-Curve Digital Signature Algorithm (EdDSA)," RFC 8032, Jan. 2017, doi: 10.17487/RFC8032.

[13] A. Karatsuba and Y. Ofman, "Multiplication of many-digital numbers by automatic computers," *Dokl. Akad. Nauk SSSR*, vol. 145, no. 2, pp. 293–294, 1962. [Online]. Available: http://mi.mathnet.ru/dan26729

[14] M. Bisheh-Niasar, R. Azarderakhsh, and M. Mozaffari-Kermani, "Cryptographic accelerators for digital signature based on ed25519," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 7, pp. 1297–1305, 2021, doi: 10.1109/TVLSI.2021.3077885.

[15] D. Owens, R. El Khatib, M. Bisheh-Niasar, R. Azarderakhsh, and M. M. Kermani, "Efficient and side-channel resistant ed25519 on arm cortex-m4," *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2024, doi: 10.1109/TCSI.2024.3384414.

NoRDEx: Un protocolo descentralizado optimista de no repudio para intercambios de datos

Fernando Román-García
Universitat Politècnica de Catalunya (UPC)
fernando.roman@upc.edu

Antonio Alarcón
Universitat Politècnica de Catalunya (UPC)
antonio.alarcon@upc.edu

Juan Hernández Serrano
Universitat Politècnica de Catalunya (UPC)
j.hernandez@upc.edu

Oscar Esparza
Universitat Politècnica de Catalunya (UPC)
oscar.esparza@upc.edu

***Resumen*—El artículo presenta el protocolo NoRDEx, diseñado para garantizar el no repudio en intercambios de datos. NoRDEx puede considerarse descentralizado, ya que elimina la necesidad de un TTP centralizado mediante el uso de un DLT para almacenar pruebas criptográficas sin revelar el mensaje intercambiado. NoRDEx es un protocolo de no repudio optimista, ya que solo utiliza la DLT en caso de disputa. El protocolo ha sido implementado y probado en entornos del mundo real, con evaluaciones de rendimiento que cubren costos, sobrecarga y tiempo de ejecución. Un análisis formal de seguridad utilizando el modelo lógico SVO demuestra la capacidad de NoRDEx para resolver disputas de manera segura.**

***Index Terms*—No repudio, Intercambio Justo, Distributed Ledger, Blockchain**

I. Introducción

Los mercados de datos se han consolidado como plataformas para la compraventa eficiente de información [1]. En este contexto, se debe solucionar el clásico problema del intercambio justo [2]: si el proveedor revela los datos antes de cobrar, el consumidor podría negarse a pagar; y si el consumidor paga por adelantado, el proveedor podría no entregarlos. Resolver disputas exige evidencias verificables y auditables que permitan a un tercero arbitrar sin ambigüedades.

Los protocolos de no repudio proporcionan no repudio de origen (NRO) y de recepción (NRR) [3], [4]; sin embargo, necesitan de un tercero de confianza (TTP) [5] que notarice el intercambio. Es evidente en este contexto la aplicabilidad de las DLTs para notarizar el intercambio sin necesidad de la TTP; sin embargo, existen retos que resolver para su aplicación en términos de coste [6], latencia y privacidad, entre otros. Además, la mayor parte de propuestas recientes acoplan pago y entrega en un mismo flujo (p. ej., con criptomonedas), lo que restringe el medio de pago y dificulta su adopción en dominios donde no siempre existe pago o este se realiza por canales tradicionales; por ejemplo, en sanidad, un hospital (proveedor) puede necesitar garantizar la entrega de un informe al asegurador (consumidor) sin que medie una transacción económica inmediata, simplemente garantizando que se podrá facturar dicho intercambio.

Este trabajo resume NoRDEx, un protocolo de no repudio descentralizado y optimista desarrollado en el marco de i3-MARKET [7]. NoRDEx es descentralizado porque sustituye a la TTP centralizada por una DLT; y es optimista porque el coste asociado a la DLT solo afecta al intercambio en caso de disputa. Además, NoRDEx se ha diseñado para mantener la privacidad del intercambio incluso en el caso de usar una DLT pública.

Es importante recalcar que NoRDEx se centra exclusivamente en garantizar el no repudio de los intercambios de datos independientemente del sistema de pago, lo que permite que proveedores y consumidores puedan elegir su método de pago preferido, ya sea por adelantado o a posteriori, con dinero fiduciario o criptomonedas. Las pruebas criptográficas generadas durante su ejecución permiten tanto una facturación justa y auditable, como la eventual resolución de disputas.

La implementación completa del protocolo NoRDEx está disponible para su descarga gratuita en [8], y el contrato inteligente de no repudio para una DLT tipo Ethereum en [9], ambos bajo la licencia de software libre EUPL v. 1.2. El artículo original [10] presenta todos los detalles del protocolo, un análisis formal de seguridad basado en el modelo lógico SVO, y una evaluación exhaustiva del rendimiento de la implementación de NoRDEx.

II. Motivación

Durante nuestro análisis en el proyecto i3-MARKET, identificamos cinco requisitos que un protocolo de no repudio/intercambio justo debía cumplir para garantizar un intercambio de datos seguro y eficiente.

1. **Sistema de pago desacoplado**. En escenarios heterogéneos, el consumidor y el proveedor deben poder decidir qué medio de pago (criptomonedas, tokens y/o dinero fiduciario) se utilizará durante el intercambio de datos. Desacoplar el pago proporciona flexibilidad para que el proveedor y el consumidor acuerden el método de pago que consideren más conveniente.
2. **Resolución de disputas**. Por motivos legales, consideramos esencial proporcionar auditorías fiables de cada intercambio de datos, mediante facturas verificables [11], que puedan usarse para resolver disputas de manera formal y legal.
3. **Descentralizado**. Es esencial evitar el uso de TTPs centralizados, ya que el protocolo no debe depender de la fiabilidad de una única entidad. Al utilizar un TTP descentralizado, el poder se distribuye entre múltiples partes, lo que facilita intercambios más fiables y justos.
4. **Ejecución optimista**. Si consumidor y proveedor actúan honestamente, no se requiere ninguna otra entidad en el intercambio de datos, y el TTP solo es necesario en caso de disputa. Al usar una DLT, esta propiedad es especialmente ventajosa, ya que los problemas de latencia asociados a su uso pueden ser significativos [12].
5. **Privacidad**. La información pública sobre el intercambio de datos no debe dar información sobre los datos

intercambiados o las partes intervinientes.

III. Estado del arte

Las primeras implementaciones de soluciones de no repudio e intercambio justo dependían de TTPs centralizados que "notarizaban" la entrega de los datos. Entre las propuestas iniciales destaca PRE [13], que utiliza varios proxies para retener los mensajes intercambiados a fin de proporcionar no repudio. También es reseñable que, dado que el uso de un notario introducía problemas de latencia evidentes; desde buen principio ha habido soluciones que usaban la TTP solo en el caso no optimista; entre ellas, cabe destacar el protocolo MPNR [14], y el protocolo OFE [15].

Hubo también algunos intentos de eliminar por completo los TTPs de los protocolos de no repudio o intercambio justo. Sin embargo, solo se lograron soluciones probabilísticas, lo que significa que estas soluciones no garantizan con certeza que el intercambio haya ocurrido, pero proporcionan una cierta probabilidad de que haya tenido lugar. Por ejemplo, en el caso de PNR [16], el proveedor divide el mensaje en n bloques iguales, donde n es desconocido para el proveedor; por lo tanto, si el consumidor desea obtener el mensaje completo, debe enviar un ACK firmado por cada bloque, confirmando generalmente la recepción de todos los bloques. Sin embargo, esta solución solo se considera válida si el consumidor no puede inferir el mensaje a partir de un fragmento, ni puede determinar el número total de bloques n.

Más recientemente, se ha introducido el cifrado retardado como una alternativa probabilística para el intercambio justo [17]. Esta técnica oculta el mensaje durante un período predefinido, asegurando que el intercambio ocurra con cierta probabilidad en lugar de certeza absoluta.

Tras la aparición de las DLTs, se han desarrollado muchos protocolos basados en esta tecnología. El SDDP [18], propuesto por Wang et al., presenta una novedosa DLT que incluye NRR en su protocolo de consenso, siendo una de las muy pocas propuestas que combinan DLT con no repudio. La mayoría de las propuestas basadas en DLTs intentan proporcionar intercambio justo. En FairSwap [19], los autores proponen un protocolo de cinco pasos que proporciona intercambio justo utilizando árboles de Merkle para generar pruebas eficientes que un contrato inteligente puede verificar fácilmente. Además, Delgado-Segura et al. [20] sugirió un protocolo que utiliza Bitcoin para el intercambio justo. En esta solución, el *Emisor* envía un mensaje cifrado fuera de la cadena al *Receptor*, y utiliza la DLT para enviar la clave de descifrado una vez que el *Receptor* ha realizado el pago. Publicaciones recientes buscan mejorar la eficiencia de los protocolos de intercambio justo. Por ejemplo, DFSE [21] es un protocolo que optimiza el intercambio de dos firmas digitales, común en transacciones que involucran Non-Fungible Tokens. Por el contrario, FairShare [22] se centra en optimizar el caso de uso del IoT. Al combinar PRE y DLTs. En FDE [23], el objetivo es mejorar FairSwap introduciendo un nuevo primitivo criptográfico llamado VECK. Finalmente, Fair^{2}Trade [24] introduce un método de verificación de claves completamente nuevo que reduce el consumo de gas en comparación con soluciones anteriores del estado del arte.

A pesar de las contribuciones anteriores, y hasta donde sabemos, no existe un protocolo que aborde el problema del intercambio justo de forma eficiente que permita desacoplar el sistema de pago a la par que elimine la dependencia de un TTP centralizado.

IV. Resumiendo NoRDEx

Como se ha mencionado en I, NoRDEx es un protocolo optimista de no repudio que proporciona un mecanismo para contabilizar de manera fiable un intercambio de datos, permitiendo que el proveedor y el consumidor verifiquen cuándo tuvo lugar el intercambio, la integridad de los datos recibidos y su recepción dentro del plazo acordado.

El funcionamiento de NoRDEx está desacoplado del sistema de pago, dejando que proveedor y consumidor lo acuerden libremente. El pago real puede realizarse por adelantado o al finalizar, y puede ser tan simple como una transferencia bancaria directa o tan complejo como un contrato inteligente que transfiera fondos automáticamente. La contabilización fiable del intercambio de datos que proporciona NoRDEx permite una facturación justa y auditable, además de aportar pruebas que pueden usarse para resolver disputas judiciales si la facturación no es adecuada o los datos intercambiados no cumplen las expectativas de alguna de las partes.

Para ilustrar el funcionamiento general del protocolo, usaremos una analogía. Imaginemos que una compradora, llamada Carla, desea adquirir una gema. El vendedor, Pedro, guarda la gema en un cofre indestructible y se lo entrega a Carla junto con un recibo que certifica la entrega. Una vez que Carla recibe el cofre y el recibo, emite un acuse de recibo a Pedro confirmando la entrega. En este momento, Pedro deposita la llave del cofre bajo la custodia de un notario. Finalmente, Carla puede recoger la llave del notario y acceder a la gema.

Una vez que comprador y vendedor completan este procedimiento, ambos pueden demostrar ante un juez que el intercambio se realizó. Si Carla no recibió la gema puede demostrárselo al juez si: 1) no generó el acuse de recibo; 2) la llave para descifrar el cofre no se depositó en el notario; o 3) la llave que se depositó en el notario no sirve para abrir el cofre. Pero nótese que solo el caso 3 puede ocurrir si se finaliza el protocolo. Si Pedro necesita probar que entregó la gema, puede demostrárselo al juez con: 1) el acuse de recibo que Carla le dio al recibir el cofre; y 2) con el acuse de depósito en el notario de la llave para abrirlo. Dado que Carla había recibido el cofre, había firmado su voluntad de abrirlo (acuse de recibo), y la llave estaba en el notario, Carla puede acceder a la gema.

En el contexto de nuestro estudio (intercambios de datos en mercados de datos), Carla representa al consumidor y Pedro al proveedor. Dado que tratamos con datos incluidos en un mensaje con un formato concreto, el proveedor puede usar un algoritmo de cifrado para garantizar que la compradora no acceda al mensaje hasta que haya emitido el correspondiente acuse de recibo. Tanto el recibo como los acuses de recibo pueden generarse mediante firmas digitales ligadas al texto cifrado. Finalmente, el papel del notario se modela mediante una DLT.

IV-A. Premisas

En esta sección se introducen tres conceptos fundamentales para comprender el protocolo: el acuerdo, el intercambio de datos y las pruebas. Es importante definirlos explícitamente,

ya que pequeñas variaciones pueden afectar tanto la seguridad como la claridad del protocolo.

IV-A1. Acuerdo:

Para ejecutar el protocolo NoRDEx, es necesario que tanto Pedro como Carla hayan establecido previamente un acuerdo que regule el intercambio de datos. Este acuerdo define las identidades digitales de ambas partes, las condiciones básicas del servicio, la interacción con la DLT, los algoritmos criptográficos que se utilizarán y otros parámetros opcionales.

IV-A2. Intercambio de datos:

Una vez que Carla y Pedro tienen un acuerdo, el contrato inteligente de NoRDEx puede asegurar el intercambio de uno o varios mensajes bajo las condiciones especificadas en dicho acuerdo. Para identificar de forma única e inequívoca cada intercambio de mensajes, se define el identificador único de intercambio *eid*, calculado como el hash de la concatenación del acuerdo asociado, el hash del texto cifrado *hash(c)* y el hash de la clave de descifrado *hash(k)*.

El intercambio queda así definido como una tupla que contiene *eid*, el acuerdo, y nuevamente *hash(c)* y *hash(k)*. La inclusión de *hash(c)* y *hash(k)* como parte del intercambio y de su *eid* constituye un compromiso de los datos cifrados y la clave de descifrado que se usará en dicho intercambio. Estas vinculaciones son esenciales para garantizar la integridad del intercambio de datos.

Esta estructura forma parte de las afirmaciones contenidas en las pruebas de NoRDEx, y es crucial para verificar que tanto el mensaje cifrado como la clave corresponden al intercambio. La frescura del intercambio y del identificador *eid* depende directamente de la frescura de la clave utilizada por NoRDEx para cifrar el mensaje. Esto significa que la clave debe usarse una sola vez.

IV-A3. Pruebas:

La ejecución del protocolo requiere que tanto Carla como Pedro generen y verifiquen pruebas criptográficas, o simplemente *pruebas*.

Una prueba π_x es un documento firmado digitalmente compuesto por un conjunto de afirmaciones ϕ_x, que sirven como evidencia de un evento. Las afirmaciones pueden incluir diversa información: tipo de documento, fecha de creación, o valores que comprometan al emisor a realizar una acción concreta. La firma digital garantiza la integridad y autenticidad del documento.

A continuación se describen las tres pruebas utilizadas en el funcionamiento de NoRDEx:

- Prueba de Origen ($\boldsymbol{\pi_{PoO}}$): generada por Pedro y retenida por Carla. Evita que Pedro niegue haber enviado m.
  Las afirmaciones en π_{PoO} son la tupla $\phi_{PoO} := (type, iss, iat, e)$. El primer campo, *type*, indica que el documento firmado es una prueba de origen. *iss* especifica el emisor (Pedro), *iat* el momento de creación de la prueba, y *e* representa el intercambio, que incluye los compromisos *hash(c)* y *hash(k)*.
- Prueba de Interés ($\boldsymbol{\pi_{PoI}}$): generada por Carla y retenida por Pedro. Confirma el interés de Carla en adquirir m, y evita que niegue haberlo recibido.
  En este caso, $\phi_{PoI} := (type, iss, iat, \pi_{PoO})$, con los campos *type* (prueba de interés), *iss* (Carla) e *iat* (marca temporal), así como la prueba de origen asociada π_{PoO}.
- Prueba de Clave ($\boldsymbol{\pi_{PoK}}$): generada por Pedro y retenida por Carla. Evita que Pedro niegue haber enviado una clave concreta.
  Las afirmaciones son $\phi_{PoK} := (type, iss, iat, \pi_{PoI}, k)$. Aparte de los campos comunes (*type*, *iss*, *iat*), incluye la prueba de interés asociada π_{PoI} y la clave k que permite a Carla descifrar y obtener m.
  Además de servir como prueba, PoK reduce el tiempo de ejecución en los casos optimistas, es decir, cuando ambas partes actúan de buena fe. Al incluir la clave k, Carla no necesita esperar a que se publique en la DLT.

IV-B. Intercambio de datos simplificado de NoRDEx

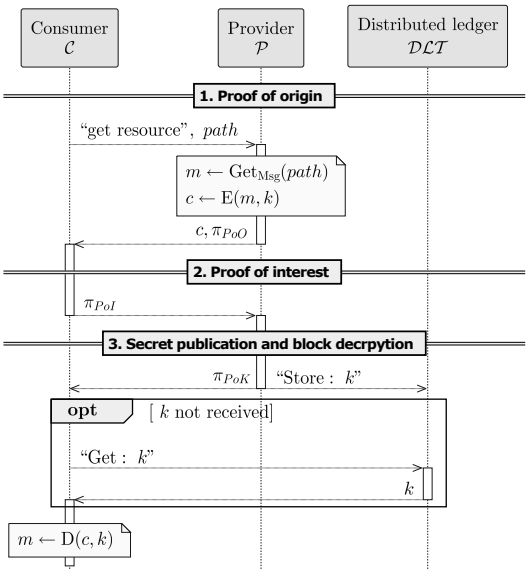


Figura 1. Simplified outline of the NoRDEx operation

La Figura 1 muestra una representación simplificada del protocolo NoRDEx. Aunque esta vista es una simplificación, resulta útil para entender su estructura básica.

El protocolo consta de tres fases distintas que involucran a tres entidades: el consumidor (Carla), el proveedor (Pedro) y la DLT.

1. **Prueba de origen.** Carla solicita un mensaje m en una ruta $path$ a Pedro. En lugar de enviar m directamente, Pedro envía el mensaje cifrado c con una nueva clave k, junto con la prueba de origen π_{PoO}, que permite a Carla demostrar que Pedro envió c.
2. **Prueba de interés.** Carla responde con una prueba de interés, que permite a Pedro demostrar que Carla ha recibido c y se compromete a descifrarlo más adelante. La PoI hace que Carla sea responsable de obtener la clave k desde la DLT si no la recibe por el canal directo en la siguiente fase.
3. **Publicación del secreto y descifrado.** Pedro envía la PoK (que contiene k) a Carla, de modo que ella puede descifrar c y obtener m. Al mismo tiempo, Pedro publica la clave k en la DLT, impidiendo que Carla niegue su recepción, ya que, en el peor de los casos, puede obtenerla directamente de la DLT.

NoRDEx está diseñado para ofrecer no repudio, pero su funcionamiento no incluye protección de datos entre provee-

dores y consumidores. Se asume que las comunicaciones entre ellos están protegidas extremo a extremo (por ejemplo, mediante TLS). De no ser así, cualquier atacante que intercepte el canal podría obtener el mensaje m al final del protocolo.

Los únicos parámetros que serán públicos en la DLT son el identificador del proveedor, el *eid* y la clave k. El resto se mantiene privado o se intercambia solo entre consumidor y proveedor a través de un canal seguro. Como c no se comparte públicamente y tanto k como c son distintos en cada intercambio, el *eid* no revela información sobre el contenido o las partes involucradas. Por tanto, la única información pública en cada intercambio es el proveedor.

V. Conclusiones

A lo largo de este resumen hemos presentado NoRDEx, un protocolo que genera evidencias para garantizar el no repudio en los intercambios de datos, protegiendo tanto al proveedor —que puede demostrar que el consumidor recibió los datos— como al consumidor —que puede probar qué datos recibió y en qué momento—. Estas evidencias permiten una contabilidad y facturación justas, y pueden emplearse como soporte en la resolución de posibles disputas.

NoRDEx ha sido diseñado para proporcionar no repudio sin necesidad de un TTP. De hecho, un contrato inteligente ejecutado sobre una DLT, intrínsecamente distribuida y fiable, asume el papel del TTP. Dado que la conexión con la infraestructura externa puede ser una fuente potencial de latencia, nuestro protocolo utiliza la DLT únicamente de forma optimista, lo que significa que las conexiones que afecten al intercambio solo serán necesarias en casos poco probables, como fallos de canal o comportamientos maliciosos por parte de alguno de los participantes.

En el artículo original [10] se presentan todos los detalles del protocolo, así como un análisis formal de seguridad basado en el modelo lógico SVO, que demuestra la capacidad de NoRDEx para resolver disputas de manera segura. Además de un análisis detallado del rendimiento de la implementación de NoRDEx, evaluando el coste económico y la sobrecarga derivada de su uso en términos de datos transmitidos y tiempo de cómputo.

Consideramos que NoRDEx ofrece una solución flexible, segura y eficiente que puede impulsar el intercambio de datos en los mercados y otros entornos digitales.

Agradecimientos

Esta iniciativa se realiza en el marco de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiadas por la Unión Europea (Next Generation), dentro de la cátedra INCIBE CARISMATICA (C066.23_UPC), así como por el Ministerio de Ciencia, Innovación y Universidades bajo el proyecto VERIFICO (PID2024-159480OB-I00), y por la Generalitat de Catalunya bajo el proyecto 2021-SGR-00594.

Referencias

[1] B. H. Wixom and J. W. Ross, "How to Monetize Your Data — sloanreview.mit.edu," 2017, January 09, [Accessed 29-07-2024]. [Online]. Available: https://sloanreview.mit.edu/article/how-to-monetize-your-data/

[2] M. Arenas, P. Barceló, L. Libkin, and F. Murlak, *Foundations of Data Exchange*. Cambridge University Press, 2014.

[3] J. Zhou and D. Gollmann, "Evidence and non-repudiation," *Journal of Network and Computer Applications*, vol. 20, no. 3, pp. 267–281, 1997. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804597900564

[4] B. Schneier, *Foundations*. John Wiley & Sons, Ltd, 2015, ch. 1, pp. 1–18.

[5] H. Pagnia, F. C. Gärtner *et al.*, "On the impossibility of fair exchange without a trusted third party," Citeseer, Tech. Rep., 1999.

[6] V. Buterin and M. Swende, "EIP-2929: Gas cost increases for state access opcodes," 2020, accessed: 2023-09-05. [Online]. Available: https://eips.ethereum.org/EIPS/eip-2929

[7] European Union, "Intelligent, Interoperable, Integrative and deployable open source MARKETplace with trusted and secure software tools for incentivising the industry data economy - i3-MARKET. H2020-ICT-2019-2 871754," 2023. [Online]. Available: https://www.i3-market.eu/

[8] i3 MARKET, "NoRDEx git repository," 2024. [Online]. Available: https://github.com/i3-Market-V3-Public-Repository/SP3-SCGBSSW-CR-NonRepudiationLibrary

[9] ——, "NRP contract git repository," 2024. [Online]. Available: https://github.com/i3-Market-V3-Public-Repository/SP3-SCGBSSW-CR-NonRepudiationProtocolSmartContract

[10] F. Román-García, J. Hernández-Serrano, and O. Esparza, "Nordex: A decentralized optimistic non-repudiation protocol for data exchanges," *Journal of Network and Computer Applications*, vol. 243, p. 104291, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804525001882

[11] G. Ghosh, "Adoption of digital payment system by consumer: a review of literature," *International Journal of Creative Research Thoughts*, vol. 9, no. 2, pp. 2320–2882, 2021.

[12] Coinbase, "Bitcoin block reward, block size, block time: what's the difference?" 2024. [Online]. Available: https://www.coinbase.com/en-es/learn/crypto-basics/bitcoin-block-reward-block-size-block-time-whats-the-difference

[13] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in *Advances in Cryptology — EUROCRYPT'98*, K. Nyberg, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 127–144.

[14] J. A. Onieva, J. Zhou, and J. Lopez, "Non-repudiation protocols for multiple entities," *Computer Communications*, vol. 27, no. 16, pp. 1608–1616, 2004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366404002269

[15] N. Asokan, M. Schunter, and M. Waidner, "Optimistic protocols for fair exchange," in *Proceedings of the 4th ACM Conference on Computer and Communications Security*, 1997, pp. 7–17.

[16] O. Markowitch and Y. Roggeman, "Probabilistic non-repudiation without trusted third party," in *Second Conference on Security in Communication Networks*, vol. 99, 1999, pp. 25–36.

[17] I. Maffei and A. W. Roscoe, "Optimally-fair exchange of secrets via delay encryption and commutative blinding," in *International Conference on Financial Cryptography and Data Security*. Springer, 2023, pp. 94–111.

[18] L. Wang, J. Liu, and W. Liu, "Staged data delivery protocol: A blockchain-based two-stage protocol for non-repudiation data delivery," *Concurrency and Computation: Practice and Experience*, vol. 33, 03 2021.

[19] S. Dziembowski, L. Eckey, and S. Faust, "Fairswap: How to fairly exchange digital goods," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 967–984.

[20] S. Delgado-Segura, C. Pérez-Solà, G. Navarro-Arribas, and J. Herrera-Joancomartí, "A fair protocol for data trading based on bitcoin transactions," *Future Generation Computer Systems*, vol. 107, pp. 832–840, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X17318344

[21] J. Chen, H. Xiao, M. Hu, and C.-M. Chen, "A blockchain-based signature exchange protocol for metaverse," *Future Generation Computer Systems*, vol. 142, pp. 237–247, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X22004356

[22] J. Sengupta, S. Ruj, and S. Das Bit, "Fairshare: Blockchain enabled fair, accountable and secure data sharing for industrial iot," *IEEE Transactions on Network and Service Management*, vol. 20, no. 3, pp. 2929–2941, 2023.

[23] E. N. Tas, I. A. Seres, Y. Zhang, M. Melczer, M. Kelkar, J. Bonneau, and V. Nikolaenko, "Atomic and fair data exchange via blockchain," Cryptology ePrint Archive, Paper 2024/418, 2024.

[24] C. Chenli, W. Tang, H. Lee, and T. Jung, "Fair2trade: Digital trading platform ensuring exchange and distribution fairness," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–16, 2024.

Nonlinear Approximations in Linear Cryptanalysis and Hadamard matrices

José Andrés Armario
Depto Matemática Aplicada I
Universidad de Sevilla (Spain)
armario@us.es

***Abstract*—Some relevant notions in Cryptography such as bent sequences, almost bent functions, and almost perfect nonlinear functions have recently been extended in new directions by means of Hadamard matrices (beyond of Sylvester type). Here, we go one step further in this project by extending the notion of *Correlation Matrices* introduced by Daemen et al. in the 1990s.**

The current paper makes some progress in the understanding of the significance of these extensions, in particular, in the context of nonlinear approximations in Linear Cryptanalysis.

***Index Terms*—Linear Cryptanalysis, Nonlinear approximations, Hadamard matrices, Walsh Hadamard transform.**

I. Introduction

Linear cryptanalysis [13] was introduced by Matsui in 1993 and it has become a standard attack method. The idea behind, can be phrased as approximating a linear Boolean function in the cipher's output by a linear Boolean function in the input. Concretely, let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be a vectorial Boolean function (e.g., a block cipher with a fixed key). Given two n-bit non-constant linear Boolean functions g and h, we want to "approximate" a Boolean function h in the output by a Boolean function g in the input with **high absolute bias**. The measurement of the bias of such an approximation comes from its *correlation*,

$$Cor_F(g,h) := 2\,Prob_x[g(x) = h(F(x))] - 1,$$

and it can be computed by

$$Cor_F(g,h) := 2^{-n} \sum_{x\in\mathbb{F}_2^m} (-1)^{g(x)+h(F(x))} \in [-1,1].$$

Generalization to nonlinear approximations (i.e. g and h being two n-bit non-constant Boolean functions g and h with the same Hamming weight), was originally discussed in [10], [11]. The main problems that come along with this generalization are the much larger search space compared to linear functions and a rather complex key dependency. These nonlinear approximations were rediscovered in the context of invariant attacks, i.e., invariant subspace attacks [12] and the nonlinear invariant attack [17]. Concretely, the existence of a nonlinear invariant for E_k (a n-block cipher with a fixed key), in many practical cases, or the existence of an invariant subspace for E_k implies the existence of highly-biased linear approximations of the cipher, i.e., $|Cor_{E_k}(l_a, l_b)| >> 2^{-n/2}$ for the linear function $l_a(x) = a^\top \cdot x$ with $a, x \in \mathbb{F}_2^n$. Hence, a usual strategy to look for highly-biased linear approximations consists of studing nonlinear invariants or invariant subspaces as a first step.

More recently, on one hand in 2018 [4], this line of work has been revised and extended to study deterministic and non-deterministic nonlinear approximations for cryptanalysis of block ciphers and cryptographic permutations and embedded it in the well-understood framework of linear cryptanalysis. On the other hand, in 2021 [5], Beyne proposed a new interpretation of linear cryptanalysis by means of the so-called geometric approach, which unifies all common variants of linear cryptanalysis where the basic tools are Fourier analysis on abelian groups and the theory of inner product space of (linear) functions. Let us point out that the theory of Hadamard bent sequences (see [16] and the references inside), which we will use here, does not fit in the former approaches because a more general transform than the Walsh-Hadamard is considered. The aim of this paper is to progress in the understanding of the significance of this theory, in particular, in the context of nonlinear approximations in Linear Cryptanalysis.

II. Preliminaries

For every positive integer n, $\mathbb{F}_2^n$ denotes the vector space of all vector of length n over the finite field with 2 elements $\mathbb{F}_2$.

Let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ be a *Boolean* function, we denote by $(-1)^{f(x)}$ the ±1-representation of f. The *Walsh-Hadamard transform* of f, denoted by $\mathcal{W}_f$, is the function from $\mathbb{F}_2^n$ into the ring of integer numbers $\mathbb{Z}$ given by

$$\mathcal{W}_f(a) = \sum_{x\in\mathbb{F}_2^n} (-1)^{a\cdot x + f(x)}$$

where $a \cdot x$ denotes the canonical inner product $a^\top x$ of a and x. The multiset $\{\mathcal{W}_f(a)\colon a \in \mathbb{F}_2^n\}$ is called the *Walsh spectrum* of the function f. The Boolean function f is *bent* if $|\mathcal{W}_f(a)|^2 = 2^n$ for all $a \in \mathbb{F}_2^n$. Hence, f can be (Boolean) bent only if n is even. More generally, bent functions have been studied in connection with group-invariant Hadamard matrices and certain (relative) difference sets, see, e.g., [3].

A square matrix H of order n with entries ±1 is said to be *Hadamard* if $HH^\top = nI$ where the $\top$ denotes the transpose and I is the identity matrix of order n. It is well known that a Hadamard matrix of order n can exist only if $n = 1, 2$ or $n \equiv 0 \mod 4$. The Hadamard conjecture states that these necessary conditions are also sufficient. The smallest open case where the existence of a Hadamard matrix is uncertain is $n = 668$. Two Hadamard matrices are equivalent if one can be obtained from the other by permuting rows or columns and by multiplying rows or columns by ±1. A Hadamard matrix is said to be *normalized* if the first row and column have all entries 1s. We always assume that our Hadamard matrices are normalized. In order 32 there are more than two million inequivalent Hadamard matrices. Historically, the first infinite

sequence of Hadamard matrices constructed was by Sylvester; they arise as Kronecker powers of the matrix

$$S_n = \otimes^n \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

It is easy to see that $S_n = ((-1)^{x\cdot y})_{x,y\in\mathbb{F}_2^n}$ and, as consequences:

- S_n can be seen as a look up table of (the ±1-representation of) the canonical inner product.
- Each of its rows corresponds to the ±1-representation of one of the 2^n linear Boolean functions.
- The entries of the vector

  $$y = S_n\, x$$

  where $x = [(-1)^{f(\mathbf{a})}]^\top_{\mathbf{a}\in\mathbb{F}_{2^n}}$ (the ±1-representation of f) with $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$, determine the Walsh spectrum of f.

In what follows, we highlight the connection between Walsh-Hadamard transform and important notions in Cryptography as the matrix of correlation [7], almost perfect nonlinear and almost bent functions.

Definition 1: Let $\mathbb{F}_2^n \to \mathbb{F}_2^n$ a vectorial Boolean function. The *correlation matrix* $C^F \in \mathbb{R}^{2^n\times 2^n}$ has entries

$$C^F_{a,b} = \frac{1}{2^n}\mathcal{W}_{bF}(a) = \frac{1}{2^n}\sum_{x\in\mathbb{F}_2^n}(-1)^{a\cdot x + b\cdot F(x)}$$

for all a and b in $\mathbb{F}_2^n$.

Remark 1: It is immediate:

- $C^F_{a,b} = Cor_F(g,h)$ for $g = l_a$ and $h = l_b$ (where $l_\alpha = \alpha\cdot x$ with $\alpha\in\mathbb{F}_2^n$). Hence, $-1 \le C^F_{a,b} \le 1$.
- $C^F = \frac{1}{2^n} S_n P S_n$ where P has entries

  $$P_{a,b} = \begin{cases} 1 & b = F(a) \\ 0 & \text{Otherwise} \end{cases} \tag{1}$$

  for all a and b in $\mathbb{F}_2^n$ where the rows and columns of the matrices involved, are indexed by the elements of $\mathbb{F}_2^n$ in the lexicographic order. Let us point out that there is a correspondence between P and the *transition matrix* introduced in [5].
- If F is a bijection then C^F is orthogonal.
- The "linear approximation table" of linear cryptanalysis for F (a block cipher with a fixed key) coincies with $\frac{1}{2}S_nPS_n$.

The following definitions have extracted from [15].

Definition 2: A function $F\colon \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$ (or $\mathbb{F}_2^n \to \mathbb{F}_2^n$) is called *almost perfect nonlinear* or APN if the mapping $x \mapsto F(x+a) - F(x)$ are 2-to-1 for all $a \neq 0$; that means $F(x+a) - F(x) = b$ has 0 or 2 solutions for $a, b \in F_{2^n}$, $a \neq 0$.

The function $F(x) = x^{-1}$ (here we use the convention $0^{-1} = 0$) is almost perfect nonlinear in $\mathbb{F}_{2^n}$ if n is odd. When n is even the the mapping $x \mapsto F(x+a) - F(x)$ are 4-to-1 for all $a \neq 0$. Partly due to its good "differential" behaviour, this function for $n = 8$ is used in the S-boxes of the Advanced Encryption Standard AES [8, Subsection 5.1.1].

Definition 3: A function $F\colon \mathbb{F}_{2^n} \to F_{2^n}$ (or $\mathbb{F}_2^n \to \mathbb{F}_2^n$) is called *almost bent* or AB if

$$\mathcal{W}_F(a,b) = \mathcal{W}_{bF}(a) = \sum_{x\in\mathbb{F}_2^n}(-1)^{a\cdot x + b\cdot F(x)} \in \{0, \pm 2^{(n+1)/2}\}$$

for all $b \neq 0$.

Remark 2: $\mathcal{W}_{bF}(a)$ can be displayed in matrix form as

$$S_n P S_n$$

where $a, b \in \mathbb{F}_2^n$ and P as in (1).

Obviously, AB functions can exist only if n is odd. The following result is immediate.

Theorem 1: [15, Theorem 7.4] Let n be odd. Then any AB function on $\mathbb{F}_2^n$ is APN.

The converse of this theorem does not hold. See [15, Theorem 7.6] for examples of APN functions which are not AB. The function $F(x) = x^{-1}$ (mentioned before) is not AB (see [6, Example 3]. All of these examples are for $n \ge 5$. For $n = 3$ all APN functions are AB.

Remark 3: For n odd, the set of all AB functions on $\mathbb{F}_2^n$, denoted by $\mathcal{AB}_n$, is a subset of the set of all APN functions on $\mathbb{F}_2^n$, denote by $\mathcal{APN}_n$. That is $\mathcal{AB}_n \subseteq \mathcal{APN}_n$. If $n \ge 5$ then $\mathcal{AB}_n \subsetneq \mathcal{APN}_n$.

III. The Walsh-Hadamard transform beyond of Sylvester type

The connection between the Sylveter Hadamard matrices and the Walsh-Hadamard transform suggest us the following study.

Definition 4: Let f be a Boolean function over $\mathbb{F}_2^n$ and $\mathbf{x}$ be the $\pm$-representation of f and H be a Hadamard matrix of order 2^n. The *Walsh spectrum of f attached to H* is defined by

$$\hat{\mathbf{x}} = H\mathbf{x}.$$

And the *Walsh-Hadamard transform* of f attached to H, denoted by $\mathcal{W}_f^H$, is the function from $\mathbb{F}_2^n$ into the integer numbers $\mathbb{Z}$ of the form

$$\mathcal{W}_f^H(w) = \sum_{x\in\mathbb{F}_2^n}(-1)^{f(x)+g(w,x)}$$

where the matrix $H = \left((-1)^{g(x,w)}\right)_{x,w\in\mathbb{F}_2^n}$.

The Walsh-Hadamard transform attached to H satisfies properties similar to the regular one.

Lemma 1: Under the same hypothesis than Definition 4.

1) $\tilde{\hat{\mathbf{x}}} = 2^n\mathbf{x}$ where $\tilde{\mathbf{x}} = H^\top\mathbf{x}$ is the anti-transform. If H is symmetric, then the anti-transform is the same as the transform.
2) (Parseval's identity)

   $$\sum_i |\hat{\mathbf{x}}(i)|^2 = 2^{2n}.$$

3) $\sum_i \mathbf{x}(i) = \hat{\mathbf{x}}(0)$.
4) If g is bilinear, then

   $$(\hat{f})^2 = \widehat{f\otimes f},$$

   where $f\otimes f$ is the convolutional product

   $$(f\otimes f)(w) = \sum_{x\in\mathbb{F}_2^n} f(x)f(w\oplus x).$$

5)

   $$\mathrm{dist}(f, r_w) = 2^{n-1} - \frac{1}{2}\mathcal{W}_f^H(w), \quad \forall w\in\mathbb{F}_2^n;$$

   where $\mathrm{dist}(f, r_w)$ is the Hamming distance between the ±1-representation of f and the w-th row of H.

Proof:

1) $\tilde{\hat{\mathbf{x}}} = H^*\hat{\mathbf{x}} = H^*H\mathbf{x} = 2^n\mathbf{x}$. Here we have applied that H is Hadamard.
2) On the one hand, we have $\hat{\mathbf{x}}^*\hat{\mathbf{x}} = \sum_i |\hat{\mathbf{x}}(i)|^2$. On the second hand, $\hat{\mathbf{x}}^*\hat{\mathbf{x}} = \mathbf{x}^*H^*H\mathbf{x} = p^n\sum_i |\mathbf{x}(i)|^2 = 2^{2n}$. We have applied that H is Hadamard again and the entries of $\mathbf{x}$ are 2-roots of the unity.
3) This identity follows from H to be dephased.
4) It is similar to the proof of [3, Lemma 2].
5) Immediate.

■

Remark 4: The hypothesis of property 4 is quite restrictive. An important challenge will be to adapt this result to a more flexible framework. Here, we have considered two possible ways:

1) This line relies on the connection between Hadamard matrices and Hadamard codes, which implies that the hypothesis of 4) can be rephrased, as the Hadamard code defined by H is linear. Then, we propose to extend to Hadamard matrices associated to the $\mathbb{Z}_{2^s}$-linear Hadamard codes (i.e., binary codes which are the image by the Carlet Gray map of some kind of codes over finite rings, see Proposition 1 of [1]).
2) In this line, some kind of 'twisted' convolution product is considered.

As an example of the 2nd line, we define $G : \mathbb{F}_2^n \times \mathbb{F}_2^n \to \mathbb{F}_2$ by $G(u,v) = h(u,v) + u\cdot v \mod 2$, i.e., G displays as the matrix logaritmic form of $H \circ S_{2^n}$, where $\circ$ denotes the Hadamard product.

Definition 5: Let f and g be two Boolean functions, we denote by $f \otimes_H g$ the convolutional product with respect to H by

$$(f \otimes_H g)(u,v) = \sum_{x\in\mathbb{F}_2^n} (-1)^{f(x)+g(x+u)+G_v(x)+G_v(x+u)+G_v(u)}$$

where $u, v \in \mathbb{F}_2^n$.

Remark 5: $(f \otimes_H g)(u)$ denotes $(f \otimes_H g)(u,u)$.

Proposition 1:

$$\widehat{(f \otimes_H g)}(u) = \widehat{f}(u)\cdot\widehat{g}(u), \quad \forall u \in \mathbb{F}_2^n.$$

Proof: Assuming $u = v$, the next identity holds

$$\widehat{(f \otimes_H g)}(u,v) =$$

$$\sum_u (-1)^{G_v(u)} \Big(\sum_x (-1)^{f(x)+g(x+u)+G_v(x)+G_v(x+u)}\Big)(-1)^{h(v,u)}$$

After replacing u by $u - x$, the double summation becomes

$$\sum_u \Big(\sum_x (-1)^{f(x)+g(u)+G_v(x)+G_v(u)}\Big)(-1)^{v\cdot(u-x)} =$$

$$\sum_u (-1)^{g(u)+v\cdot u+G_v(u)} \sum_x (-1)^{f(x)+v\cdot x+G_v(x)} =$$

$$\widehat{f}(v)\cdot\widehat{g}(v)$$

■

IV. Bent sequences, almost bent function and correlation matrices for Hadamard matrices

We will review the notions of Bent sequences and almost bent functions for Hadamard matrices. In the same way, we will introduce the analog notion for the correlation matrices.

Motivated by the *physical unclonable function problem*, Solé [14] introduced a new notion of bent sequences as solutions $(\mathbf{x}, \mathbf{y})$ to the system

$$\frac{1}{\sqrt{n}} H\mathbf{x}^\intercal = \mathbf{y}^\intercal,$$

where H is a real Hadamard matrix of order n and $\mathbf{x}, \mathbf{y} \in \{\pm 1\}^n$. In this context, $\mathbf{x}$ is called a bent sequence for H and $\mathbf{y}$ is called the dual of $\mathbf{x}$. If $\mathbf{x} = \mathbf{y}$, then $\mathbf{x}$ is called self-dual, and it is an eigenvector of H. The notion of (self-dual) H-bent sequences has been extended to Butson Hadamard matrices and related to spherical codes [16].

Definition 6: [2] We say that $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ is almost bent for H if

$$\mathcal{W}_F^H(a,b) \in \{0, \pm 2^{(n+1)/2}\}, \text{ for } (a,b) \in \mathbb{F}_2^n \times \mathbb{F}_2^n \setminus \{(0,0)\}$$

and $\mathcal{W}_F^H(0,0) = 2^n$, where

$$\mathcal{W}_F^H(a,b) = (HPH^\top)_{a,b} = \sum_{x\in\mathbb{F}_2^n} (-1)^{h(a,x)+h(b,F(x))}$$

with

$$(-1)^{h(x,y)} = H_{x,y}, \ \forall x, y \in \mathbb{F}_2^n.$$

and P is the matrix of (1).

Remark 6: The notion of Walsh transform can be extended to a pair (H_1, H_2) of Hadamard matrices in the following way,

$$\mathcal{W}_F^{H_1,H_2} = H_1 P H_2^\top$$

The MAGMA system includes a database of Hadamard matrices (HMs), which in particular contains 23 representatives of equivalence classes of HMs of order 32. For $1 \leqslant i \leqslant 23$ we call these representatives HM$(32,i)$. It turns out that HM$(32,1)$ is equivalent to S_5. So 5-variable AB functions for HM$(32,1)$ are precisely the classic AB functions. We now consider $H =$ HM$(32,2)$.

Example 1: [2] Given $F\colon \mathbb{F}_2^5 \to \mathbb{F}_2^5$ with coordinate functions

$y_1 = 1 + x_1x_2 + x_3 + x_2x_3 + x_2x_4 + x_3x_4 + x_2x_3x_4 + x_2x_5 + x_1x_2x_5 + x_3x_5 + x_1x_3x_5 + x_4x_5.$

$y_2 = 1 + x_3 + x_1x_3 + x_4 + x_2x_4 + x_1x_3x_4 + x_5 + x_1x_5 + x_2x_5 + x_3x_5 + x_1x_3x_5 + x_2x_3x_5 + x_2x_4x_5 + x_3x_4x_5.$

$y_3 = x_2 + x_1x_2 + x_3 + x_2x_3 + x_1x_2x_3 + x_4 + x_1x_4 + x_1x_3x_4 + x_1x_5 + x_1x_2x_5 + x_2x_3x_5 + x_2x_4x_5 + x_3x_4x_5.$

$y_4 = 1 + x_1 + x_2 + x_1x_2 + x_4 + x_1x_4 + x_2x_4 + x_1x_2x_4 + x_3x_4 + x_5 + x_2x_5 + x_1x_2x_5 + x_3x_5 + x_1x_3x_5 + x_2x_3x_5.$

$y_5 = 1 + x_1 + x_2 + x_1x_2x_3 + x_4 + x_1x_4 + x_2x_4 + x_1x_2x_4 + x_2x_3x_4 + x_5 + x_2x_5 + x_2x_3x_5 + x_4x_5 + x_2x_4x_5 + x_3x_4x_5.$

is a AB function for (H, S_5).

Definition 7: Let $\mathbb{F}_2^n \to \mathbb{F}_2^n$ be a vectorial Boolean function. The *correlation matrix* $C^{F,H} \in \mathbb{R}^{2^n\times 2^n}$ has entries

$$C_{a,b}^{F,H} = \frac{1}{2^n}\mathcal{W}_F^H(a,b)$$

■ for all a and b in $\mathbb{F}_2^n$.

Let us observer that for all $a, b \in \mathbb{F}_2^n$

$$C_{a,b}^{F,H} = 2\,Prob_x[h(a,x) = h(b,F(x))] - 1$$

with

$$(-1)^{h(x,y)} = H_{x,y},\ \forall x, y \in \mathbb{F}_2^n.$$

In other words, each entry of the correlation matrix with respect to H is the bias of the (nonlinear) approximations defined by the rows of H.

Let $M = (m_{i,j})_{1 \le i,j \le 2^n}$ be a square matrix, we define

$$\Lambda(M) = \sup_{i,j \neq 0} |m_{i,j}|.$$

Question 1: Given $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ a vectorial Boolean function with correlation matrix C^F. Design a strategy to find a Hadamard matrix H such that $\Lambda(C^F) < \Lambda(C^{F,H})$. Looking for a matrix with maximum Λ is a combinatorial optimization problem.

Focusing on Ascon's 5-bit S-box, $\mathcal{S}$ (see page 27 of [9]). We have found a matrix H (see http://neilsloane.com/hadamard/had.32.t3.txt), where

$$\Lambda(C^{\mathcal{S},H}) = 1$$

the optimal one. Observe that $\Lambda(C^{\mathcal{S}}) = 1/2$ where $16\,C^{\mathcal{S}}$ is Ascon's linear approximation table (see [9, Table 11, p. 27]). Labeling the input bits to $\mathcal{S}$ as $x_4, \ldots, x_0$ and the output bits as $y_4, \ldots, y_0$ the approximation $1 \oplus x_4 \oplus x_4 \cdot x_3 \oplus x_2 \oplus x_1 = y_2$ holds with probability 1. Surprisingly (or may be, not), this is the third coordinate function of $\mathcal{S}$.

Lemma 2: Let $f\colon \mathbb{F}_2^n \to \mathbb{F}_2$ be a balanced Boolean function with $f(\mathbf{0}) = 0$. If H is a Hadamard matrix of order 2^n, then there exists an equivalent Hadamard matrix where one of its rows is equal to $(-1)^f$.

Proof: Immediate. ∎

We believe that it is worth asking whether these deterministic approximations have any consequence on dominant (nonlinear) trails of ciphers. Common ciphers are designed as round-iterated functions, and this structure is used to compute their correlation.

Definition 8: Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an iterated function of the form $F = F_t \circ \cdots \circ F_2 \circ F_1$ with $F_i\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Given $(t+1)$ Boolean functions $g_0, \ldots, g_t$ corresponding to some rows of H (a Hadamard matrix), the tuple $(g_0, g_1, \ldots, g_t)$ is said to be a *H-trail* over F and its correlation is defined as

$$\begin{aligned} Cor_{F_1,\ldots,F_t}(g_0, g_1, \ldots, g_t) &= \prod_{i=1}^t Cor_{F_i}(g_{i-1}, g_i) \\ &= \frac{1}{2^{t \cdot n}} \prod_{i=1}^t \left(HP_iH^\top\right)_{i-1,i} \end{aligned}$$

where P_i is defined as (1). If H is equal to the Sylverster-Hadamard matrix S_n then, we get the usual *linear trail*.

Theorem 2: Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an iterated function of the form $F = F_t \circ \cdots \circ F_2 \circ F_1$ with $F_i\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ and H be a Hadamard matrix of order 2^n. Then, the correlation of the H-nonlinear approximation (g_0, g_t) over F is given by the sum of the correlations of all its constituent nonlinear trails, i.e.,

$$Cor_F(g_0, g_t) = \sum_{g_1,\ldots,g_{t-1} \in \mathcal{R}_H} Cor_{F_1,\ldots,F_t}(g_0, g_1, \ldots, g_t),$$

where $\mathcal{R}_H$ denotes the set of Boolean functions related to the rows of H. If H is equal to the Sylverster-Hadamard matrix S_n then, we get the usual linear approximation.

Proof:

$$Cor_F(g_0, g_t) = \frac{1}{2^n}\left(H\,(\prod_{i=1}^t P_i)\,H^\top\right)_{0,t} = \frac{1}{2^{t \cdot n}}\left(\prod_{i=1}^t HP_iH^\top\right)_{0,t}$$

By a simple comprobation.

$$\sum_{g_1,\ldots,g_{t-1} \in \mathcal{R}_H} \prod_{i=1}^t \left(HP_iH^\top\right)_{i-1,i} = \left(\prod_{i=1}^t HP_iH^\top\right)_{0,t}.$$

∎

In the next result, we study the correlation oer the S-box layer, one of the buliding blocks of iterated ciphers.

Proposition 2: Let $\mathcal{S}\colon (\mathbb{F}_2^b)^s \to (\mathbb{F}_2^b)^s$ be the s-time parallel application of a function $S\colon F_2^b \to \mathbb{F}_2^b$. Then,

$$Cor_{\mathcal{S}}(g, h) = \prod_{i=0}^{s-1} Cor_S(g_i, h_i).$$

Proof: It is easy to check that

$$Cor_{\mathcal{S}}(g, h) = \left(\bigotimes_1^s HPH^\top\right)_{g,h}$$

where $\otimes$ denotes the Kronecker product of matrices.

∎

The next result about nonlinear trail composition is known, but we give the idea of the proof into a matrix language, which in turn, give two ways to compute these trails.

Theorem 3: [4, Theorem 3] Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ and let g, h be n-bit balanced Boolean functions. Then,

$$Cor_F(g, h) = \sum_{\gamma, \gamma' \in \mathbb{F}_2^n} Cor_g(\gamma) Cor_F(\gamma, \gamma') Cor_h(\gamma'),$$

where $Cor_g(\gamma) = Cor_{Id}(g, \gamma) = Cor_{Id}(\gamma, g)$. By abuse of notation, γ denotes the linear function $l_\gamma(x) = \gamma^\top \cdot x$.

Proof:

$$\frac{1}{2^n}(HPH^\top)_{i,j} = \frac{1}{2^n}(HS_n)\,\frac{1}{2^n}(S_nPS_n)\,\frac{1}{2^n}(HS_n)^\top$$

where H is a Hadamard matrix where its i-th row (respectively, j-th row) corresponds with g (resp. h). ∎

Theorem 4: Let $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$ and let g_1, g_2 be n-bit balanced Boolean functions. Then,

1) There is Q_i a permutation matrix such that the second row of $H_i = S_nQ_i$ is $(-1)^{g_i}$ for $i = 1, 2$.
2) $$C^{F,\,(H_1,H_2)} = C^{G_1 F G_2^{-1}}$$
where G_i is a bijection associated to Q_i in the way of (1).

Proof: The proof follows from Lemma 2 and the identity

$$H_1PH_2^\top = S_n(Q_1PQ_2^\top)S_n.$$

∎

The last result is consistent with subsection 4.1 of [4].

V. Conclusion

We have made some progress in the understanding of the significance of the theory of Hadamard bent sequences, in particular, in the context of nonlinear approximations in Linear Cryptanalysis. In this line, we have rediscovered some results of [4]. But this is a work in progress, and advances in Question 1 would be priority in the near future.

Appendix

As consequences of Lemma 1.

Remark 7: Let H be a Hadamard matrix of order n. We have:

- By 5), $L^H(f) = \max\{|\mathcal{W}_f^H(w)|\colon w \in \mathbb{F}_2^n\}$ can serve as a measure of the distance of the set of Boolean functions describes by the rows of H (and their complements) to f.
- By 2), $L^H(f) \geq 2^{n/2}$. Let $\mu^H(n)$ be the smallest value of $L^H(f)$ among the Boolean functions on $\mathbb{F}_2^n$. Then the covering radius of Hadamard code constituted by the rows of H (and their complements) is equal to $2^{n-1} - \frac{1}{2}\mu^H(n)$.
- For $F\colon \mathbb{F}_2^n \to \mathbb{F}_2^n$. Let

  $$\mathcal{W}^H(F) = \{\mathcal{W}_{bF}^H(w)\colon b \in \mathbb{F}_2^n \setminus \{0\},\, w \in \mathbb{F}_2^n\}$$

  and let

  $$L^H(F) = \max\{L^H(bF)\colon b \in \mathbb{F}_2^n \setminus \{0\}\}.$$

  Then $L^H(F)$ can serve as measure of the (Hamming) distance of F (component by component) to the set of Boolean functions describes by the rows of H (and their complements).
- Let $\mu^H(n,n)$ be the smallest value of $L^H(F)$ among all the vectorial Boolean functions from $\mathbb{F}_2^n$ to $\mathbb{F}_2^n$. We have not found yet an interpretation of $\mu^H(n,n)$ in terms of covering radius of a code.

After doing some experimentation and computation. We believe,

Conjecture 1: Let n be an odd integer.

1) $\mu^H(n,n) \geq 2^{\frac{n+1}{2}}$.
2) Let H be the Paley (type I) Hadamard matrix of order 2^n (that is, $2^n - 1$ is Mersenne prime) then $\mu^H(n,n) \neq 2^{\frac{n+1}{2}}$.
3) The following statements are equivalent:
   a) $\mu^H(n,n) = 2^{\frac{n+1}{2}}$.
   b) The covering radius of the linear code generated by H is equal to the covering radius of the first order Reed-Muller code $R(1,n)$.

Let us point out that $R(1,n)$ is the Hadamard code associated to $S_{2,n}$ (the Sylvester Hadamard matrix of order 2^n) which is linear.

Acknowledgements

The author would like to thank Kristeen Cheng for her reading of this manuscript.

This research was partially supported by the Project PID2024-160735NB-I00 from the Ministry of Science and Innovation of the Government of Spain.

References

[1] J. A. Armario, I. Bailera, R. Egan: "Butson full propelinear codes", *Des. Codes Cryptogr.* Vol. 91, pp. 333–351, 2023.

[2] J. A. Armario, S. Barrera-Acebedo: "Almost Bent functions and Almost Perfect Non-linear functions for Hadamard matrices". In preparation.

[3] J. A. Armario, R. Egan, D. L. Flannery: "Generalized partially bent functions, generalized perfect arrays and cocyclic Butson matrices", *Cryptogr. Commun.*, vol. 16, pp. 323–337, 2024.

[4] Christof Beierle, Anne Canteaut, and Gregor Leander: "Nonlinear Approximations in Cryptanalysis Revisited", *IACR Transactions on Symmetric Cryptology*, 4, pp. 80–101, 2018.

[5] Tim Beyne: "A Geometric Approach to Linear Cryptanalysis", In: M. Tibouchi, H. Wang (eds) Advances in Cryptology – ASIACRYPT 2021, ASIACRYPT 2021, Lecture Notes in Computer Science, vol 13090. Springer, Cham, 2021.

[6] F. Chabaud, and S. Vaudenay: "Links between differential and linear cryptoanalysis", in:*Advances in Cryptology-EUROCRYPT'94 (Perugia)*. LNCS, vol. 950, pp. 356–356, Springer, Berlin, 1995.

[7] Joan Daemen, René Govaerts, and Joos Vandewalle: "Correlation matrices", In Bart Preneel, editor, FSE'94, volume 1008 of LNCS, pages 275–285. Springer, Heidelberg, December 1995.

[8] M. Dworkin et al: "Advanced Encryption Standard (AES)", *Federal Inf. Process. Stds. (NIST FIPS)*, National Institute of Standards and Technology, Gaithersburg, MD, 2001, [online], https://doi.org/10.6028/NIST.FIPS.197 (Accessed June 21, 2024)

[9] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schlaffer: "Ascon v1.2: Lightweight Authenticated Encryption and Hashing", *Journal of Cryptology*, vol. 34, n. 3, 33, 2021.

[10] Carlo Harpes, Gerhard G. Kramer, and James L. Massey: "A generalization of linear cryptanalysis and the applicability of Matsui's piling-up lemma", In Louis C. Guillou and Jean-Jacques Quisquater, editors, EUROCRYPT'95,volume 921 ofLNCS, pages 24–38. Springer, Heidelberg, May 1995.

[11] Lars R. Knudsen and Matthew J. B. Robshaw: "Non-linear approximations in linear cryptoanalysis". In Ueli M. Maurer, editor, EUROCRYPT'96, volume1070 of LNCS, pages 224–236. Springer, Heidelberg, May 1996.

[12] Gregor Leander, Mohamed Ahmed Abdelraheem, Hoda AlKhzaimi, and ErikZenner: "A cryptanalysis of PRINTcipher: The invariant subspace attack", In Phillip Rogaway, editor, CRYPTO 2011, volume 6841 of LNCS, pages 206–221.Springer, Heidelberg, 2011.

[13] Mitsuru Matsui: "The first experimetal cytanalysis of data encryption standard", In Yvo Desmedt, editor, CRYPTO'94, volume 839 of LNCS, pages 1–11. Springer, Heidelberg, August 1994.

[14] P. Solé, W. Cheng, S. Guilley, and O. Rioul: "Bent sequences over Hadamard codes for physically unclonable functions", *IEEE International Symposium on Information Theory (ISIT)*, Melbourne, Australia, 801-806, 2021.

[15] A. Pott: "Almost pefect and planar functions", *Des. Codes Cryptogr.*, vol. 78, n. 1, pp. 141–195, 2016.

[16] M. Shi, D. Lu, J.A. Armario, R. Egan, F. Ozbudak, and P. Solé: "Hadamard bent sequences and spherical designs". *J. Algebr. Comb.*, Vol. 62, 43, pp. 1–26, 2025.

[17] Yosuke Todo, Gregor Leander, and Yu Sasaki: "Nonlinear invariant attack -practical attack on full SCREAM, iSCREAM, and Midori64", In Jung HeeCheon and Tsuyoshi Takagi, editors, ASIACRYPT 2016, Part II, volume 10032 of LNCS, pages 3–33. Springer, Heidelberg, 2016.

¿Sobremuestreo o submuestreo? Cómo mejorar las métricas de eficiencia en la detección de amenazas basada en aprendizaje automático

Jordi Doménech, Olga León, Isabel V. Martin-Faus, Enrica Zola, Josep Pegueroles
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya
c. Jordi Girona 1-3, 08034, Barcelona, Spain
{jordi.domenech.fons, olga.leon, isabel.martin-faus, enrica.zola, josep.pegueroles}@upc.edu

***Resumen*—La ciberseguridad en IoMT es especialmente crítica tanto por sus vulnerabilidades específicas como por la sensibilidad de los datos manejados. Entre las soluciones más prometedoras para proteger estos entornos destacan los Sistemas de Detección y Prevención de Intrusiones (IDPS) basados en técnicas de IA. En estos sistemas, la calidad de los datos de entrenamiento (dataset) y su preprocesado influyen decisivamente en el rendimiento del modelo. En particular, la distribución estadística de clases del dataset condiciona la capacidad del algoritmo para aprender, especialmente en escenarios desbalanceados. Este trabajo analiza el impacto del desbalanceo en el dataset CICIoMT2024 y se evalúan diversas técnicas de submuestreo (RUS) y sobremuestreo (SMOTE, Borderline-SMOTE y ADASYN). Los resultados muestran que, si bien las técnicas de balanceo permiten obtener matrices de confusión más equilibradas, ello conlleva una disminución del F1-score. Se requiere un compromiso que evidencia la necesidad de métodos híbridos capaces de alcanzar una matriz de confusión más uniforme, garantizando, al mismo tiempo, un F1-score mínimo aceptable.**

***Index Terms*—Internet de las Cosas Médicas (IoMT), Sistemas de Detección de Intrusiones (IDS), Aprendizaje automático, Desbalanceo de clases, Preprocesamiento, Modelado de amenazas.**

I. Introducción

El aprendizaje automático es un pilar fundamental para el desarrollo de sistemas de detección de intrusiones (IDS) en entornos de IoT e IoT médicas (IoMT) [1]. Numerosos estudios han propuesto IDS basados en técnicas de machine learning (ML) y deep learning (DL) como mecanismos eficaces para detectar amenazas, que las soluciones tradicionales basadas en firmas no abordan adecuadamente [2], [3], [4].

Una de las dimensiones más críticas en la investigación de IDS es el desbalanceo de clases en los conjuntos de datos (datasets) utilizados para entrenar los modelos basados en aprendizaje automático. En escenarios reales, las muestras suelen estar fuertemente sesgadas: ataques frecuentes como Denial of Service (DoS) o Distributed Denial of Service (DDoS) dominan el conjunto de datos, mientras que amenazas más sigilosas, como spoofing o ataques de reconocimiento, se encuentran escasamente representadas [5], [6]. Este desbalanceo afecta negativamente a los modelos de detección, que tienden a optimizar su rendimiento hacia las clases mayoritarias, reduciendo drásticamente la capacidad para identificar ataques minoritarios pero críticos [7], [8].

Este trabajo tiene como objetivo estudiar cómo la distribución estadística de muestras entre clases influye en el rendimiento de los modelos de aprendizaje automático empleados en IDS para entornos IoMT. Para ello, se analiza el desbalanceo presente en el dataset CICIoMT2024 [5] y se evalúa el impacto de diversas técnicas de submuestreo y sobremuestreo sobre la capacidad del modelo para identificar tanto ataques frecuentes como amenazas poco representadas. Asimismo, se investiga el compromiso existente entre preservar la distribución original del conjunto de datos, para favorecer determinadas métricas globales, y corregir el desbalanceo que dificulta la detección de clases minoritarias. La metodología propuesta supone una primera aproximación al diseño de estrategias de balanceo híbridas capaces de optimizar simultáneamente la fidelidad estadística del dataset y la equidad en la detección de todas las categorías de ataques.

El artículo se estructura de la siguiente manera: la sección II revisa el estado del arte en técnicas de balanceo para clasificadores basados en aprendizaje automático. La sección III presenta la metodología seguida en el presente trabajo, incluyendo la descripción del dataset, el preprocesamiento aplicado, las técnicas de balanceo evaluadas, las métricas utilizadas y los modelos empleados. La sección IV expone y discute los resultados obtenidos tanto en validación como en la fase de prueba, comparando el impacto de cada técnica de balanceo. Finalmente, la sección V recoge las conclusiones del estudio y plantea posibles líneas futuras de investigación.

II. Estado del arte

Para mitigar el problema del desbalanceo de datos, diversos trabajos han propuesto técnicas de balanceo sintético o de reajuste de distribución [9], [10]. Por ejemplo, el uso de Wasserstein GAN con penalización del gradiente (WGAN-GP) ha permitido generar muestras sintéticas de ataques minoritarios, aumentando así la diversidad de los datos disponibles para el entrenamiento [11]. Del mismo modo, el algoritmo SMOTE (Synthetic Minority Oversampling Technique) ha sido aplicado a conjuntos de datos IoMT, logrando una mejora en la detección de ataques poco representados [12]. Otros trabajos recientes también aplican técnicas de balanceo, como ADASYN (Oversample using Adaptive Synthetic) o Borderline-SMOTE, para generar o reforzar clases minoritarias en conjuntos de datos desbalanceados, mejorando la

detección de los ataques minoritarios [13], [14].

A pesar de estos avances, una proporción considerable de estudios sigue sin aplicar ninguna técnica de balanceo o apenas la mencionan superficialmente. En investigaciones centradas en la detección de *botnets*, por ejemplo, se han entrenado modelos sin ningún tipo de ajuste de distribución, lo que introduce un sesgo hacia las clases más frecuentes o mayoritarias [15]. De manera similar, varios enfoques híbridos basados en redes neuronales convolucionales (CNN) o *autoencoders* son entrenados directamente sobre datos desbalanceados sin corrección alguna [16], [17]. Esta falta de consistencia metodológica dificulta la generalización de los resultados, ya que los modelos entrenados sobre distribuciones sesgadas tienden a mostrar un rendimiento deficiente cuando se enfrentan a entornos reales con diferentes distribuciones de muestras por clase. Por tanto, tal como se muestra en este trabajo, la correcta gestión del desbalanceo de clases es un desafío crucial para mejorar la fiabilidad y robustez de los sistemas de detección de intrusiones en entornos IoT e IoMT.

III. Metodología

En este trabajo se presenta una aproximación preliminar a la gestión del desbalanceo de datos aplicada a un dataset concreto, descrito a continuación.

III-A. Descripción del dataset

La base de datos utilizada es la CICIoMT2024 [5]. Este conjunto de datos amplía el alcance de otras previas bases de datos tradicionales de detección de intrusiones, poniendo énfasis particular en el dominio del IoMT. Este dataset fue generado mediante una plataforma experimental heterogénea compuesta por aproximadamente 40 dispositivos IoT e IoMT, tanto reales como simulados, que incluyen monitores de pacientes, bombas de infusión, pasarelas IoT y dispositivos médicos inteligentes. Todos ellos se comunican empleando diversos protocolos, tales como Wi-Fi, MQTT y Bluetooth, lo que permite representar con alta fidelidad la complejidad y la heterogeneidad de los entornos médicos conectados. El dataset contiene tanto tráfico benigno (*Benign*) como un subconjunto de 18 tipos de ataques agrupados en 5 categorías: ataques DoS, ataques DDoS, ataques ARP spoofing, ataques basados en el protocolo MQTT, y ataques de reconocimiento (*Recon*). La distribución de muestras por cada clase del conjunto de entrenamiento del dataset se ilustra en la Fig. 1, donde se puede apreciar que los ataques DDoS son 2,5 veces más frecuentes que los ataques DoS, y que estos últimos son casi 7 veces más frecuentes que los ataques MQTT. También cabe destacar que los ataques ARP representan alrededor de un 0.3 % de los DDoS.

III-B. Preparación y manejo del conjunto de datos

Tras elegir el dataset, y antes de la fase de balanceo y entrenamiento del modelo, se aplicó un proceso de limpieza de datos según los siguientes pasos.

1. Estandarización de etiquetas. En primer lugar, se estandarizaron las etiquetas de clase con el fin de crear un esquema de clasificación multiclase coherente y específico para cada conjunto de datos.

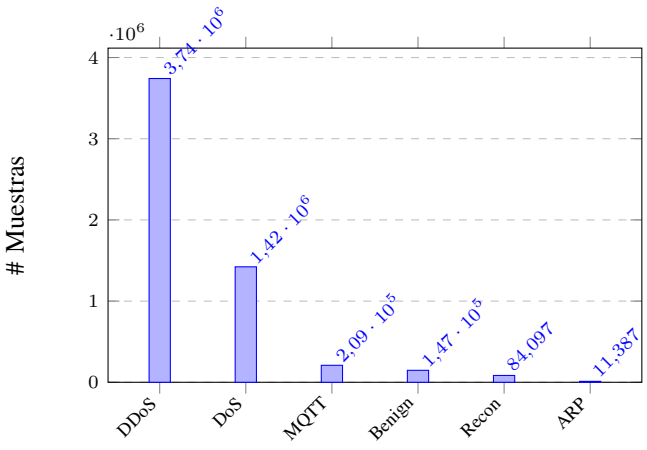


Figura 1. Distribución de muestras (Baseline) en el conjunto de entrenamiento.

2. Codificación One-Hot. Posteriormente, las clases que originalmente correspondían a variables categóricas fueron transformadas a un formato numérico mediante codificación *One-Hot*. Este procedimiento evita que el modelo asuma una relación ordinal entre las categorías.
3. Normalización Min-Max. Finalmente, todas las características numéricas fueron escaladas a un rango uniforme de $[0, 1]$ utilizando la normalización *Min-Max*. Esta transformación se muestra en la Eq. 1:

$$X_{escalado} = \frac{X - X_{min}}{X_{max} - X_{min}}, \tag{1}$$

donde X representa el valor original, y X_{min} y X_{max} son, respectivamente, los valores mínimo y máximo del conjunto de datos para esa característica. Este paso es esencial para garantizar que las variables con rangos de valores más amplios no ejerzan una influencia desproporcionada durante el proceso de aprendizaje del modelo.

Finalmente, adoptamos una metodología estándar de división *train–validation–test* en el conjunto de datos CICIoMT2024 con el objetivo de garantizar experimentos rigurosos y sin sesgo. Se dividió inicialmente en un subconjunto de entrenamiento/validación (80 %) y otro de prueba (20 %). Esta división se realizó mediante muestreo estratificado y aleatorio, asegurando que la distribución original de las clases se mantuviera en cada partición. Asimismo, también se utilizó un proceso de validación cruzada de 5 iteraciones (*5-fold cross-validation*) en la fase de entrenamiento/validación.

III-C. Técnicas de balanceo aplicadas

Para mitigar el desbalanceo de clases en nuestro estudio, hemos aplicado cuatro técnicas de balanceo ampliamente utilizadas en la literatura científica [18]:

1. Random Undersampling (RUS). Consiste en reducir aleatoriamente el tamaño de las clases mayoritarias para igualar un número de muestras concreto. En nuestro estudio, las muestras de las clases han sido reducidas hasta igualarse al número de muestras de ARP spoofing (la clase minoritaria), tal y como se ve en la Fig. 2.

2. SMOTE (Syntethic Minority Oversampling Technique). SMOTE crea nuevas instancias sintéticas de las clases minoritarias a partir de interpolaciones entre muestras existentes. Específicamente, para cada instancia minoritaria se elige uno de sus k-vecinos más cercanos y se genera un nuevo punto en la línea que une ambas instancias.
3. Borderline-SMOTE. Es una variante de SMOTE que genera muestras alrededor de la frontera entre las clases; de esta forma, refuerza las regiones más críticas para la clasificación.
4. ADASYN (Adaptive Synthetic Sampling). Crea muestras sintéticas como SMOTE, pero de forma adaptativa: la cantidad de nuevos puntos generados para cada instancia minoritaria depende de la densidad local de la clase mayoritaria alrededor de ella.

Para los casos 2, 3 y 4, en nuestros experimentos hemos usado la técnica para aumentar las clases minoritarias (ARP spoofing, Recon y Benign) hasta el mismo número de muestras que la tercera clase mayoritaria MQTT, tal y como se observa en la Fig. 3.

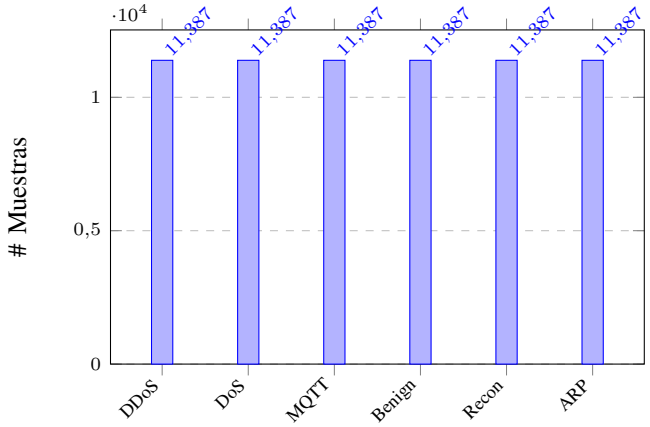


Figura 2. Distribución de muestras usando submuestreo (RUS) en el conjunto de entrenamiento.

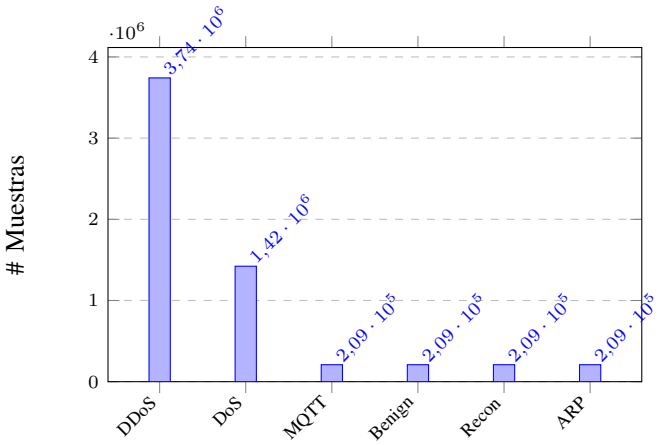


Figura 3. Distribución de muestras usando sobremuestreo (SMOTE, Borderline-SMOTE y ADASYN) en el conjunto de entrenamiento.

III-D. Métricas de evaluación en clasificación multiclase

La evaluación del rendimiento de un modelo de clasificación multiclase requiere el uso de métricas que permitan cuantificar su capacidad para predecir correctamente las distintas categorías. Las métricas de evaluación pueden calcularse de forma global o de forma individual para cada clase y, en este último caso, las métricas se pueden promediar usando la media macro o micro, cada una con interpretaciones y usos particulares. Las métricas globales evalúan el rendimiento total del modelo considerando todas las clases de manera conjunta. La más utilizada es la precisión global (*accuracy*), definida como la proporción de observaciones correctamente clasificadas respecto del total de muestras. Esta medida proporciona una visión general del funcionamiento, pero puede resultar engañosa en contextos con clases desbalanceadas, ya que su valor tiende a estar dominado por el rendimiento sobre las clases más frecuentes. En consecuencia, un modelo que clasifique correctamente la mayoría de las instancias de la clase dominante puede alcanzar una alta precisión global, incluso si falla sistemáticamente en las clases minoritarias. Por esta razón, la precisión global debe complementarse con métricas que consideren la distribución desigual de las clases, como las medias macro y micro.

Por un lado, las métricas de tipo macro evalúan el rendimiento por clase de manera independiente y luego promedian los resultados, otorgando el mismo peso a cada clase, sin importar su tamaño o frecuencia en el conjunto de datos. Este enfoque permite analizar si el modelo se comporta de manera equitativa entre las distintas clases, ya que todas las categorías contribuyen por igual al resultado final. Por tanto, las métricas macro son especialmente útiles para detectar sesgos inducidos por el desbalanceo de clases, ya que una baja puntuación en clases minoritarias afectará de forma proporcional el promedio global. Por otro lado, las métricas micro combinan las predicciones individuales de cada clase ponderando cada clase según su número de muestras, y por tanto reflejan el rendimiento global del modelo pero suelen estar más influenciadas por las clases mayoritarias.

En nuestro estudio, utilizaremos las métricas macro, ya que nos dan información más relevante al tratar con distribuciones estadísticas desbalanceadas, otorgando el mismo peso a todas las clases. Asimismo, sea un problema de clasificación con K clases, para cada clase $i \in \{1, 2, \ldots, K\}$, se definen los siguientes parámetros:

- TP_i: Verdaderos positivos de la clase i.
- FP_i: Falsos positivos de la clase i.
- FN_i: Falsos negativos de la clase i.

Las métricas individuales por clase se calculan de la siguiente forma:

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \tag{2}$$

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i} \tag{3}$$

$$\text{F1}_i = 2 \cdot \frac{\text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{4}$$

El promedio global (macro) se obtiene calculando las métricas individualmente para cada clase y luego promediándolas

de forma uniforme:

$$\text{Precision}_{macro} = \frac{1}{K}\sum_{i=1}^{K}\text{Precision}_i \quad (5)$$

$$\text{Recall}_{macro} = \frac{1}{K}\sum_{i=1}^{K}\text{Recall}_i \quad (6)$$

$$\text{F1}_{macro} = \frac{1}{K}\sum_{i=1}^{K}F1_i \quad (7)$$

III-E. Determinación del número de muestras por clase

El conjunto de datos original presenta un desbalanceo significativo entre clases (Fig. 1), lo que puede afectar directamente al rendimiento del modelo, especialmente en la detección de ataques minoritarios. Por este motivo, nuestro objetivo en esta fase metodológica es identificar qué distribución estadística de muestras entre clases debemos utilizar, atendiendo a al rendimiento del modelo de clasificación deseado. Se pone especial énfasis en la métrica $F1_{macro}$, que evalúa de manera equilibrada todas las clases independientemente de su frecuencia. Esta métrica resulta especialmente útil en escenarios multiclase desbalanceados, ya que penaliza el mal rendimiento en clases minoritarias, incluso cuando la precisión global del modelo es alta.

III-F. Entrenamiento de los modelos de aprendizaje automático

En este estudio se utilizan cuatro tipos distintos de modelos de aprendizaje automático, descritos en la Tabla I. Estos modelos simples y ligeros son ideales para ser utilizados en entornos IoT e IoMT [7].

Tabla I
MODELOS DE APRENDIZAJE AUTOMÁTICO UTILIZADOS EN EL ESTUDIO.

Modelos	Hiperparámetros
Decision Tree (DT)	{criterion='gini', max_depth=10}
Random Forest (RF)	{criterion='gini', n_estimators=100}
XGBoost (XGB)	{objective='multi:softprob', max_depth=6, learning_rate=0.3, subsample=0.8, colsample_bytree=0.8, eval_metric='mlogloss'}
Feedforward Neural Network (FNN)	{hidden_layer_sizes=(64,64,64), optimizer='adam', activation='relu', learning_rate=0.001, beta_1=0.9, beta_2=0.999, epsilon=1e-07, early_stopping=True, monitor='val_loss', patience=5}

IV. RESULTADOS Y DISCUSIÓN

Los resultados obtenidos en la fase de validación se presentan en la Tabla II. En esta tabla se analiza el rendimiento de los cuatro modelos evaluados junto con las cuatro técnicas de balanceo (RUS, SMOTE, Borderline-SMOTE y ADASYN), así como el rendimiento sin aplicar ningún método de submuestreo o sobremuestreo (Baseline). Las métricas de evaluación utilizadas corresponden al promedio global (macro), además se resalta en negrita el mejor resultado alcanzado. Asimismo, dentro de cada técnica de balanceo, se identifica el modelo con el rendimiento más alto, el cual será posteriormente utilizado en la fase de prueba. Los resultados muestran una clara superioridad de los modelos basados en árboles: el Random Forest (RF) ofrece los mejores resultados cuando se aplican técnicas de sobremuetreo, mientras que XGBoost (XGB) destaca cuando se aplica Random Undersampling (RUS).

Tabla II
RESULTADOS DE LOS EXPERIMENTOS EN LA FASE DE VALIDACIÓN USANDO DISTINTOS MODELOS DE APRENDIZAJE AUTOMÁTICO.

Técnica de balanceo	Modelo	Accuracy	Precision (*macro*)	Recall (*macro*)	F1 (*macro*)
Baseline	DT	0.9977	**0.9911**	0.9287	0.9537
	RF	**0.9987**	0.9889	0.9628	**0.9748**
	XGB	0.9986	0.9761	**0.9664**	0.9710
	FNN	0.8053	0.8616	0.7737	0.7954
RUS	DT	0.9942	0.8826	0.9704	0.9007
	RF	0.9949	**0.9288**	0.9757	0.9445
	XGB	**0.9976**	0.9275	**0.9838**	**0.9474**
	FNN	0.7353	0.7503	0.8065	0.7324
SMOTE	DT	0.9966	0.9096	0.9695	0.9256
	RF	**0.9985**	**0.9706**	0.9729	**0.9716**
	XGB	0.9983	0.9433	**0.9806**	0.9585
	FNN	0.8057	0.8156	0.8431	0.7888
Borderline SMOTE	DT	0.9957	0.9372	0.9447	0.9377
	RF	**0.9985**	**0.9768**	0.9683	**0.9725**
	XGB	0.9983	0.9518	**0.9749**	0.9623
	FNN	0.8016	0.8058	0.8159	0.7803
ADASYN	DT	0.9956	0.9305	0.9525	0.9361
	RF	**0.9984**	**0.9762**	0.9689	**0.9725**
	XGB	0.9983	0.9525	**0.9765**	0.9633
	FNN	0.8034	0.8089	0.8269	0.7863

La Tabla III presenta los resultados en la fase de prueba de los modelos seleccionados en la fase de validación. Para un análisis detallado, se emplea la métrica F1-score por clase ($F1_i$), y su versión agregada macro ($F1_{macro}$). Los modelos alcanzan un rendimiento notable (por encima del 98 %) en las clases Benign, DDoS, DoS, MQTT y Recon. Sin embargo, la clase ARP spoofing obtiene consistentemente los valores más bajos, lo cual refleja el fuerte desbalanceo existente en la distribución del conjunto de datos. Esta discrepancia también impacta negativamente en la métrica $F1_{\text{macro}}$, donde la clase minoritaria ARP reduce el rendimiento global del modelo.

Tal como era de esperar, el desbalanceo en la distribución de clases penaliza la capacidad de aprendizaje sobre las clases minoritarias. Este efecto se intensifica en problemas multiclase como el que nos ocupa, donde la heterogeneidad en la frecuencia de aparición de cada categoría complica aún más la adquisición de fronteras de decisión adecuadas para todas ellas. A pesar de ello, los resultados de validación muestran una clara dominancia del modelo RF en condiciones Baseline, es decir, sin aplicar ninguna técnica de balanceo. Esto sugiere que, desde un punto de vista estadístico, la distribución original del conjunto de datos constituye un punto de partida óptimo para maximizar métricas globales de evaluación, especialmente cuando las clases mayoritarias dominan el rendimiento global del modelo.

Sin embargo, un análisis más profundo basado en las matrices de confusión normalizadas revela que, aunque el RF sin utilizar ninguna técnica de balanceo (Baseline) alcanza métricas globales elevadas, su matriz de confusión presenta el comportamiento más desequilibrado: sólo predice exactamente el 88.22 % de los ataques ARP spoofing (Fig. 4). Por lo contrario, el modelo XGB combinado con RUS (Fig. 5) muestra la diagonal más equilibrada entre todas las clases, con

Tabla III
RESULTADOS POR CLASE Y GLOBALES DE LOS EXPERIMENTOS EN LA FASE DE PRUEBA.

Técnica de balanceo	Modelo	$F1_{ARP}$	$F1_{Benign}$	$F1_{DDoS}$	$F1_{DoS}$	$F1_{MQTT}$	$F1_{Recon}$	$F1_{macro}$
Baseline	RF	**0.8698**	**0.9876**	**0.9999**	**0.9997**	**0.9977**	**0.9859**	**0.9873**
RUS	XGB	0.7303	0.9763	0.9996	0.9990	0.9963	0.9804	0.9470
SMOTE	RF	0.8526	0.9853	0.9998	**0.9997**	0.9976	0.9850	0.9700
Borderline-SMOTE	RF	0.8681	0.9848	**0.9999**	**0.9997**	0.9976	0.9831	0.9722
ADASYN	RF	0.8692	0.9843	**0.9999**	**0.9997**	0.9976	0.9836	0.9724

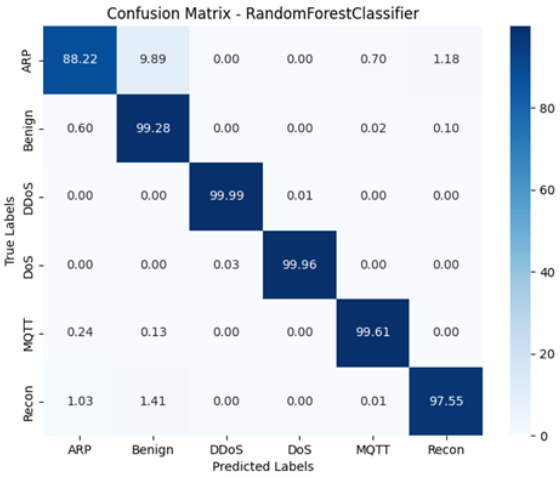


Figura 4. Matriz de confusión en la fase de prueba del modelo RF sin utilizar ninguna técnica de balanceo (Baseline).

mejoras sustanciales en la detección de la clase minoritaria ARP. Resultados similares se observan en las matrices de confusión del modelo RF cuando se aplican las técnicas de sobremuestreo SMOTE, Bordeline-SMOTE o ADASYN (Fig. 6, Fig. 7 y Fig. 8).

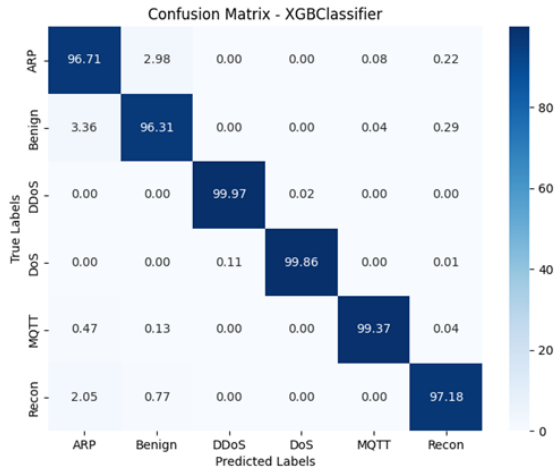


Figura 5. Matriz de confusión en la fase de prueba del modelo XGB utilizando Random Undersampling (RUS).

Estos resultados indican que la mejor estrategia para optimizar simultáneamente las métricas globales del modelo es a partir de la distribución original (Baseline). En cambio, para obtener un equilibrio de la matriz de confusión entre clases debemos aplicar técnicas de sobremuestreo o submuestreo. Asimismo, el sobremuestreo (SMOTE, Borderline-SMOTE, ADASYN) permite aumentar la representatividad de

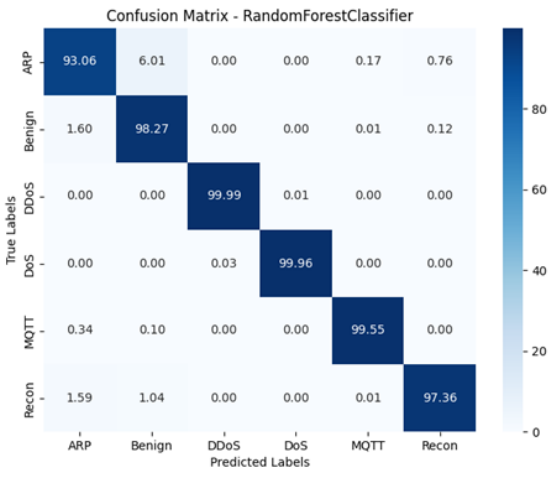


Figura 6. Matriz de confusión en la fase de prueba del modelo RF utilizando SMOTE Oversampling.

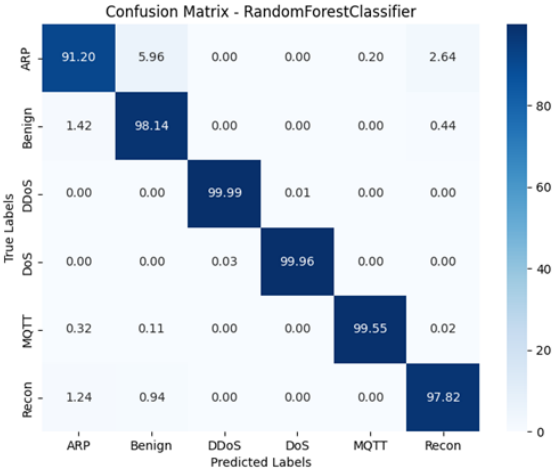


Figura 7. Matriz de confusión en la fase de prueba del modelo RF utilizando Borderline-SMOTE Oversampling.

las clases minoritarias, reforzando la capacidad del modelo para aprender sus características; y el submuestreo (RUS) reduce la influencia de las clases mayoritarias, evitando que el modelo quede sesgado hacia ellas. En conjunto, los resultados sugieren que mantener la distribución base del conjunto de datos y combinarla con técnicas de balanceo podría mejorar la capacidad del modelo para identificar tanto ataques frecuentes como infrecuentes, alcanzando un rendimiento más robusto y equilibrado en escenarios de seguridad IoMT.

V. CONCLUSIONES Y LÍNEAS FUTURAS

En este trabajo se han analizado distintas técnicas de sobremuestreo y submuestreo para mejorar las métricas de

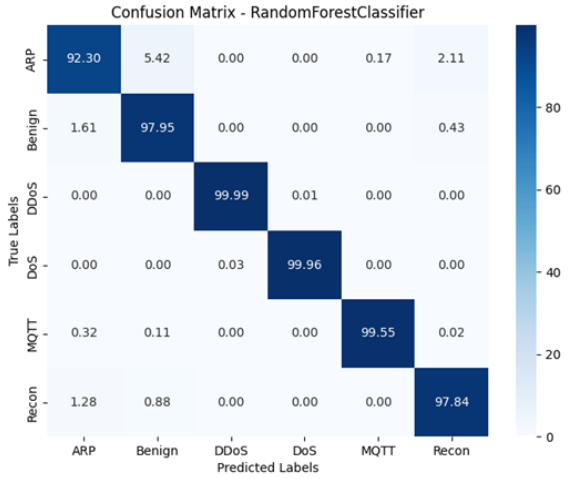


Figura 8. Matriz de confusión en la fase de prueba del modelo RF utilizando ADASYN Oversampling.

eficiencia en la detección de amenazas, en especial en entornos IoT/IoMT. Los resultados obtenidos demuestran que, aunque las técnicas de sobremuestreo (SMOTE, Borderline-SMOTE y ADASYN) y la técnica de submuestreo (RUS) permiten mejorar la detección de clases minoritarias especialmente visibles en las matrices de confusión mediante una diagonal más equilibrada, los modelos entrenados sin aplicar ningún método de balanceo (Baseline) alcanzan el mejor rendimiento global en términos de métricas agregadas, destacando especialmente el modelo Random Forest (RF) con una F1-score del 98.73 %. Esta evidencia sugiere que al modificar la distribución estadística original del dataset de forma agresiva, podemos inducir a representaciones que no siempre reflejan correctamente el comportamiento de los datos. No obstante, las mejoras observadas en las matrices de confusión de los modelos utilizando técnicas de balanceo demuestran que el uso exclusivo de la distribución original también tiene limitaciones, especialmente en ataques con baja frecuencia como el ARP spoofing. En conjunto, los resultados ponen de manifiesto la necesidad de buscar un equilibrio entre el respeto por la distribución original del dataset y la corrección del desbalance extremo entre clases. Esta combinación permitiría preservar la fidelidad estadística del conjunto de datos sin sacrificar la capacidad del modelo para identificar amenazas poco frecuentes pero críticas en entornos médicos.

Como trabajo futuro, proponemos profundizar en estrategias híbridas de balanceo que combinen técnicas de submuestreo y sobremuestreo, manteniendo controlada la distribución original del dataset. Además, investigaciones futuras también podrían explorar métodos de balanceo adaptativos y técnicas de *data augmentation* utilizando modelos generativos como GANs especializadas en IoMT.

Agradecimientos

Esta publicación y sus resultados de investigación cuentan con el apoyo del programa predoctoral AGAUR-FI Ajuts Joan Oró (2024 FI-1 00643) Ajuts Joan Oró (Generalitat de Catalunya) y de la Cátedra de Ciberseguridad CARISMATICA financiada por la UE e INCIBE.

Referencias

[1] J. Doménech, I. V. Martin-Faus, S. Mhiri, and J. Pegueroles, "Ensuring patient safety in iomt: A systematic literature review of behavior-based intrusion detection systems," *Internet of Things*, vol. 28, p. 101420, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660524003615

[2] T. Adeyemi, F. Ngobigha, and A. Ez-Zizi, "Future-proofed intrusion detection for internet of things with machine learning," in *2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC)*, 2025, pp. 1–6.

[3] A. Aljuhani, A. Alamri, P. Kumar, and A. Jolfaei, "An intelligent and explainable saas-based intrusion detection system for resource-constrained iomt," *IEEE Internet of Things Journal*, vol. 11, no. 15, pp. 25 454–25 463, 2024.

[4] A. Alarcón, O. León, J. Pegueroles, and J. Doménech, "Técnicas de machine learning para la detección de ciberataques en redes IoT médicas," *XVIII Reunión Española sobre Criptología y Seguridad de la Información: XVIII RECSI, León 23-25 octubre 2024*, pp. 25–30, 2025.

[5] S. Dadkhah, E. C. P. Neto, R. Ferreira, R. C. Molokwu, S. Sadeghi, and A. A. Ghorbani, "Ciciomt2024: A benchmark dataset for multi-protocol security assessment in iomt," *Internet of Things*, vol. 28, p. 101351, 2024.

[6] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "Ciciot2023: A real-time dataset and benchmark for large-scale attacks in iot environment," *Sensors*, vol. 23, no. 13, 2023.

[7] J. Doménech, O. León, M. S. Siddiqui, and J. Pegueroles, "Evaluating and enhancing intrusion detection systems in iomt: The importance of domain-specific datasets," *Internet of Things*, vol. 32, p. 101631, 2025.

[8] J. Doménech, S. Mhiri, M. Shuaib Siddiqui, and J. Pegueroles, "Preventive and reactive cybersecurity techniques on iot devices in healthcare environments," in *2025 IEEE 11th International Conference on Network Softwarization (NetSoft)*, 2025, pp. 261–264.

[9] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. L. P. Chen, "A survey on imbalanced learning: latest research, applications and future directions," *Artificial Intelligence Review*, vol. 57, no. 6, p. 137, May 2024. [Online]. Available: https://doi.org/10.1007/s10462-024-10759-6

[10] M. B. Musthafa, S. Huda, Y. Kodera, M. A. Ali, S. Araki, J. Mwaura, and Y. Nogami, "Optimizing iot intrusion detection using balanced class distribution, feature selection, and ensemble machine learning techniques," *Sensors*, vol. 24, no. 13, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/13/4293

[11] S.-a. Ayoub, A.-G. Mohammed Ali, and B. Narhimene, "Enhanced intrusion detection system for remote healthcare," in *Advances in Computing Systems and Applications*, M. R. Senouci, S. Y. Boulahia, and M. A. Benatia, Eds. Cham: Springer International Publishing, 2022, pp. 323–333.

[12] M. Gautam, S. Ahuja, and A. Kumar, "Intrusion detection system for internet of thing environment using feature engineering and balanced random forest algorithm," in *2024 International Conference on Expert Clouds and Applications (ICOECA)*. IEEE, 2024, pp. 449–455.

[13] C. Zhang, J. Li, N. Wang, and D. Zhang, "Research on intrusion detection method based on transformer and cnn-bilstm in internet of things," *Sensors*, vol. 25, no. 9, 2025. [Online]. Available: https://www.mdpi.com/1424-8220/25/9/2725

[14] X. Chen, Z. Gong, D. Huang, N. Jiang, and Y. Zhang, "Overcoming class imbalance in network intrusion detection: A gaussian mixture model and adasyn augmented deep learning framework," in *Proceedings of the 2024 4th International Conference on Internet of Things and Machine Learning*, ser. IoTML '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 48–53. [Online]. Available: https://doi.org/10.1145/3697467.3697595

[15] S. Saif, N. Yasmin, and S. Biswas, "Feature engineering based performance analysis of ml and dl algorithms for botnet attack detection in iomt," *International Journal of System Assurance Engineering and Management*, vol. 14, no. 1, pp. 512–522, Mar 2023.

[16] G. Kirubavathi, S. Sivakumar, and S. Manickam, "Optimized hybrid approach for anomaly detection of ddos and network attacks in iomt systems using autoencoders and tabnet," in *2024 4th International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, 2024, pp. 1–7.

[17] A. Mohammadi, H. Ghahramani, S. A. Asghari, and M. Aminian, "Securing healthcare with deep learning: A cnn-based model for medical iot threat detection," in *2024 19th Iranian Conference on Intelligent Systems (ICIS)*, 2024, pp. 168–173.

[18] I. Dey and V. Pratap, "A comparative study of smote, borderline-smote, and adasyn oversampling techniques using different classifiers," in *2023 3rd International Conference on Smart Data Intelligence (ICSMDI)*, 2023, pp. 294–302.

Quantum Entanglement and Communication

Guillermo Morales-Luna
Computer Science Department
Cinvestav, Mexico
gmorales@cinvestav.mx

***Abstract*—Quantum communication allows supercoding: by the transmission of quantum states a greater number of classical bits may be transmitted. For this purpose, entanglement is an essential feature. We outline very basic procedures to decide whether a quregister, a word of qubits, can be factored as the tensor product of shorter quregisters and we recall well known protocols to simulate entanglement. Certainly, the current exposition may be considered mainly as a lecture note. Finally, we explore the use of super-dense encoding in multiparty protocols.**

***Index Terms*—Quantum communication, entanglement, entanglement simulation, superdense coding.**

I. Introduction

Entanglement is an essential component in quantum computing and quantum cryptography. In order to simulate entanglement, two parties, Alice and Bob receive corresponding qubits, without any further communication in between, and they emit respective (classical) bits, say b_A and b_B which should be statistically correlated, namely either always (i.e., with probability 1) $b_A = b_B$ or always $b_A = 1 + b_B$ (in the arithmetic of $\mathbb{Z}_2$) [1], [2]. In 1999, a simulation procedure was proposed [3] and there a motivation for superdense coding was given, namely the possibility to communicate more classical bits than the actually communicated "information particles", the qubits. Superdense coding may be treated either by considering multiparty entanglement [4], [5] or multilevel quantum states.

Here we expose the basic notions of entanglement and communication in the context of formal Hilbert space analysis. As a further development, outside the scope of this paper, is entanglement swapping [6].

II. Decision on entanglement

Let I be a non-empty set, $\mathbb{K}$ be a field and $\mathbb{K}^I$ be the set of sequences $(x_i)_{i\in I}$ such that for each $i \in I : x_i \in \mathbb{K}$ and $x_i = 0$ except for a finite number of indices i, namely $\mathbb{K}^I$ is the *direct sum of card*(I) *copies of* $\mathbb{K}$. Clearly $\mathbb{K}^I$ is a vector space over $\mathbb{K}$ and its canonical (Hamel) basis is $\left(e_j = (\delta_{ij})_{i\in I}\right)_{j\in I}$ where δ_{ij} is Kronecker's delta.

Let $\mathbb{B} = \{0, 1\}$ be the alphabet of two symbols, or *bits*, and $\mathbb{B}^* = \bigcup_{n\geq 0} \mathbb{B}^n$ be the *dictionary* of $\mathbb{B}$. The dictionary has a structure of monoid under concatenation $\mathbb{B}^* \times \mathbb{B}^* \to \mathbb{B}^*$, $(\boldsymbol{\sigma}_0, \boldsymbol{\sigma}_1) \mapsto \boldsymbol{\sigma}_0\boldsymbol{\sigma}_1$, where the unit is the empty word λ.

Let $\mathbb{C}$ be the field of complex numbers, $\forall n \in \mathbb{N}$: $\mathbb{H}_n = \mathbb{C}^{\mathbb{B}^n}$, and $\mathbb{H}_* = \mathbb{C}^{\mathbb{B}^*}$. Any vector $z = (z_{\boldsymbol{\sigma}})_{\boldsymbol{\sigma}\in\mathbb{B}^*} \in \mathbb{H}_*$ will be written as $z = \sum_{\boldsymbol{\sigma}\in\mathbb{B}^*} z_{\boldsymbol{\sigma}} e_{\boldsymbol{\sigma}}$. The canonical basis of $\mathbb{H}_*$ is $E = (e_{\boldsymbol{\sigma}})_{\boldsymbol{\sigma}\in\mathbb{B}^*}$.

Remark 1. *The following relations hold:*

- $\forall n \geq 0$ $\mathbb{H}_n$ *is isomorphic to a subspace of* $\mathbb{H}_*$. *Indeed, the identity map* $e_{\boldsymbol{\sigma}} \mapsto e_{\boldsymbol{\sigma}}$ *extends to a linear monomorphism* $\mathbb{H}_n \to \mathbb{H}_*$.
- $n_0 < n_1 \implies \mathbb{H}_{n_0}$ *is isomorphic to a subspace of* $\mathbb{H}_{n_1}$. *Indeed, the map* $e_{\boldsymbol{\sigma}} \mapsto e_{\boldsymbol{\sigma}0^{(n_1-n_0)}}$ *extends to a linear monomorphism* $\mathbb{H}_{n_0} \to \mathbb{H}_{n_1}$.

Consider the operation $\otimes_E : E \times E \to E$, $(e_{\boldsymbol{\sigma}}, e_{\boldsymbol{\tau}}) \mapsto e_{\boldsymbol{\sigma\tau}}$. Thus E is a non-commutative monoid with operation $\otimes_E$ and unit e_λ and $\mathbb{B}^* \to E$, $\boldsymbol{\sigma} \mapsto e_{\boldsymbol{\sigma}}$ is a natural monoid isomorphism. The operation $\otimes_E$ can be extended by bilinearity to the *vector tensor product*

$$\begin{array}{rcl} \otimes : \mathbb{H}_* \times \mathbb{H}_* & \to & \mathbb{H}_* \ , \\ \left(\sum_{\boldsymbol{\sigma}\in\mathbb{B}^*} x_{\boldsymbol{\sigma}} e_{\boldsymbol{\sigma}}, \sum_{\boldsymbol{\tau}\in\mathbb{B}^*} y_{\boldsymbol{\tau}} e_{\boldsymbol{\tau}}\right) & \mapsto & \sum_{\boldsymbol{\sigma},\boldsymbol{\tau}\in\mathbb{B}^*} x_{\boldsymbol{\sigma}} y_{\boldsymbol{\tau}} e_{\boldsymbol{\sigma\tau}} \end{array}$$

Remark 2. $\otimes$ *is associative, non-commutative with unit* e_λ. *If* $x_1 \in \mathbb{H}_{n_1}$ *and* $x_0 \in \mathbb{H}_{n_0}$ *then* $x_1 \otimes x_0 \in \mathbb{H}_{n_1+n_0}$.

We say that a vector $z \in \mathbb{H}_n$ is (n_1, n_0)-*decomposable* if $n = n_1 + n_0$ and there are $x_1 \in \mathbb{H}_{n_1}$, $x_0 \in \mathbb{H}_{n_0}$ such that $z = x_1 \otimes x_0$.

Remark 3. *The following assertions are equivalent:*

- $\sum_{\boldsymbol{\sigma}\in\mathbb{B}^n} z_{\boldsymbol{\sigma}} e_{\boldsymbol{\sigma}} = \left(\sum_{\boldsymbol{\sigma}_1\in\mathbb{B}^{n_1}} x_{1\boldsymbol{\sigma}_1} e_{\boldsymbol{\sigma}_1}\right) \otimes \left(\sum_{\boldsymbol{\sigma}_0\in\mathbb{B}^{n_0}} x_{0\boldsymbol{\sigma}_0} e_{\boldsymbol{\sigma}_0}\right)$
- $\forall \boldsymbol{\sigma} \in \mathbb{B}^n,\ \boldsymbol{\sigma}_1 \in \mathbb{B}^{n_1},\ \boldsymbol{\sigma}_0 \in \mathbb{B}^{n_0}$: $[\sigma = \sigma_1\sigma_0 \implies x_{1\boldsymbol{\sigma}_1} x_{0\boldsymbol{\sigma}_0} = z_{\boldsymbol{\sigma}}]$.

2^n *coefficients* z *are determined by* $2^{n_1} + 2^{n_0}$ *coefficients* x.

Given $z \in \mathbb{H}_n$ *to find* $x_1x_0 \in \mathbb{H}_{n_1} \times \mathbb{H}_{n_0}$ *such that* $z = x_1 \otimes x_0$, *if they exist, poses a quadratic equation system.*

A vector $z \in \mathbb{H}_n$ is *entangled* if for no $n_0 \in [\![1, n-1]\!]$, z is $(n - n_0, n_0)$-decomposable (where for $i < j$, $[\![i, j]\!] = \{i, i+1, \ldots, j-1, j\}$).

Decision Problem.
Instance: A vector $z \in \mathbb{H}_n$
Solution: A decision whether z is entangled or not.

Factorization Problem.
Instance: A vector $z \in \mathbb{H}_n$ and an integer n_0 such that $0 < n_0 < n$.
Solution: $x_1 \in \mathbb{H}_{n-n_0}$, $x_0 \in \mathbb{H}_{n_0}$ such that $z = x_1 \otimes x_0$, if z is $(n - n_0, n_0)$-decomposable.

From now on, we are going to restrict the Factorization Problem to unitary vectors, since evidently:

$$\|x_1\| = 1 \ \&\ \|x_0\| = 1 \implies \|x_1 \otimes x_0\| = 1.$$

In order to solve the Factorization Problem, given $z \in \mathbb{H}_n$ we must decide whether the quadratic equation system stated at Remark 3 has solutions on $\mathbb{C}$.

Let $n \in \mathbb{N}$, $n_0 \in [\![1, n-1]\!]$ and

$$\begin{array}{rcl} \phi_{nn_0} : [\![0, 2^{n-n_0}-1]\!] \times [\![0, 2^{n_0}-1]\!] & \to & [\![0, 2^n-1]\!] \ , \\ (i,j) & \mapsto & 2^{n_0} i + j. \end{array}$$

ϕ_{nn_0} is a bijection and its inverse is

$$\psi_{nn_0} : k \mapsto (\psi_{0nn_0}(k), \psi_{1nn_0}(k)) = (k \operatorname{div} 2^{n_0}, k \bmod 2^{n_0}).$$

The quadratic equation system stated at Remark 3 can be written in the equivalent form $\forall (i,j) \in [\![0, 2^{n-n_0}-1]\!] \times [\![0, 2^{n_0}-1]\!]$:

$$x_{1i} x_{0j} = z_{\phi_{nn_0}(i,j)},$$

which gives the matrix equation $X = Z$ where

$$\begin{array}{rcl} X & = & (x_{1i}x_{0j})_{ij \in [\![0,2^{n-n_0}-1]\!] \times [\![0,2^{n_0}-1]\!]} \ , \\ Z & = & \left(z_{\phi_{nn_0}(i,j)}\right)_{ij \in [\![0,2^{n-n_0}-1]\!] \times [\![0,2^{n_0}-1]\!]}. \end{array} \quad (1)$$

A. A First Factorization Algorithm

We will use the following notation: If M is a complex matrix, m_{ij} denotes the (i,j)-th entry of M, M_i its i-th row and, M^j its j-th column. A vector shall be considered as well as a column vector.

Suppose that matrix X is a solution of (1) for given Z and $n_0 \in [\![1, n-1]\!]$. The following remarks are evident:

1) For any $(i,j) \in [\![0, 2^{n-n_0}-1]\!] \times [\![0, 2^{n_0}-1]\!]$, if $z_{ij} = 0$ then either $X_i = 0$ or $X^j = 0$.
2) For any $i \in [\![0, 2^{n-n_0}-1]\!]$, the i-th row of X is of the form $X_i = x_{1i} [x_{0j}]_{j \in [\![0,2^{n_0}-1]\!]} = x_{1i} x_0^T$. Consequently, the row-rank of X is 1.
3) For any $j \in [\![0, 2^{n_0}-1]\!]$, the j-th column of X is of the form $X^j = [x_{1i}]_{i \in [\![0,2^{n-n_0}-1]\!]} x_{0j} = x_1 x_{0j}$. Consequently, the column-rank of X is 1.

Then the following solving criteria to semidecide the Decision Problem follow:

1) If there is a pair of indices $(i,j) \in [\![0, 2^{n-n_0}-1]\!] \times [\![0, 2^{n_0}-1]\!]$, such that $z_{ij} = 0$ and $Z_i \neq 0$ and $Z^j \neq 0$ then z is not $(n-n_0, n_0)$-decomposable.
2) If $\operatorname{rank}(Z) > 1$ then z is not $(n-n_0, n_0)$-decomposable.

Let us assume that $\operatorname{rank}(Z) = 1$. Let $z_0 \in \mathbb{C}^{2^{n_0}}$ be an unitary vector spanning $\mathcal{L}(Z_i)_{i \in [\![0,2^{n-n_0}-1]\!]}$ and let $z^0 \in \mathbb{C}^{2^{n-n_0}}$ be an unitary vector spanning $\mathcal{L}\left(Z^j\right)_{j \in [\![0,2^{n_0}-1]\!]}$. Then $x_0 = z_0$ and $x_1 = z^0$.

The algorithm for $(n-n_0, n_0)$-decomposability depicted at Table I results. There, the statement 3. and the test at 4. check whether the spaces spanned by the columns and the rows of Z are 1-dimensional. This checking can be performed equivalently as follows:

- Two columns j_0, j_1 are parallel if the ratios $\frac{z_{ij_0}}{z_{ij_1}}$ have a constant value (independent of i), and
- the space spanned by the columns is 1-dimensional if each column is parallel to the first column.

A similar criterion holds for rows.

The operation `Merge` at statement 4.c is necessary since at statement 2. several entries of the resulting vectors are forced to be 0, and the current matrix Z has non-zero entries after statement 2.

Input. $n \in \mathbb{N}$, $n_0 \in [\![1, n-1]\!]$, $z \in \mathbb{H}_n$
Output. $x_1 \in \mathbb{H}_{n-n_0}$, $x_0 \in \mathbb{H}_{n_0}$: $z = x_1 \otimes x_0$ if z is $(n-n_0, n_0)$-decomposable.
Procedure. PartialDecomp

1) Let $Z = \left(z_{\phi_{nn_0}(i,j)}\right)_{ij \in [\![0,2^{n-n_0}-1]\!] \times [\![0,2^{n_0}-1]\!]}$;
2) for each (i,j) such that $z_{ij} = 0$ do
   if $Z_i = 0$ then
   $x_{1i} := 0$; suppress i-th row in Z ; renumber indices i
   else if $Z^j = 0$ then
   $x_{0j} := 0$; suppress j-th column in Z ; renumber indices j
   else "z is not $(n-n_0, n_0)$-decomposable" ;
3) $k := \operatorname{rank}(Z)$;
4) if $k > 1$ then "z is not $(n-n_0, n_0)$-decomposable"
   else
   a) let $z_0 \in \mathbb{C}^{2^{n_0}}$ be an unitary vector spanning $\mathcal{L}(Z_i)_{i \in [\![0,2^{n-n_0}-1]\!]}$;
   b) let $z^0 \in \mathbb{C}^{2^{n-n_0}}$ be an unitary vector spanning $\mathcal{L}\left(Z^j\right)_{j \in [\![0,2^{n_0}-1]\!]}$;
   c) output $x_0 = \texttt{Merge}(x_0, z_0)$ and $x_1 = \texttt{Merge}(x_1, z^0)$.

TABLE I
ALGORITHM **PartialDecomp** PERFORMING $(n-n_0, n_0)$-DECOMPOSABILITY.

A vector $z \in \mathbb{H}_n$ is entangled whenever the algorithm **PartialDecomp**(n, n_0, z) witnesses that z is not $(n-n_0, n_0)$-decomposable for each $n_0 \in [\![1, n-1]\!]$.

B. A Second Factorization Algorithm

Suppose that for $z \in \mathbb{C}^{2^n}$, the matrix X is a solution of (1), for $n_0 \in [\![1, n-1]\!]$.

Let $J_0 = \{j \in [\![0, 2^n-1]\!] |\ x_{0j} \neq 0\}$ be the set of indices with non-zero value x_{0j}, and, similarly, let $I_1 = \{i \in [\![0, 2^{n-n_0}-1]\!] |\ x_{1i} \neq 0\}$. Define:

$$\begin{array}{rcl} \forall j_0 \in J_0 \ \forall j_1 \in [\![0, 2^n-1]\!] & : & \xi_{0j_0j_1} = \dfrac{x_{0j_1}}{x_{0j_0}} \\ \forall i_0 \in I_1 \ \forall i_1 \in [\![0, 2^{n-n_0}-1]\!] & : & \xi_{1i_0i_1} = \dfrac{x_{1i_1}}{x_{1i_0}} \end{array} \quad (2)$$

First, we observe that, for each $j_0 \in J_0$:

$$\|x_0\|^2 = \sum_{j_1=0}^{2^{n_0}-1} |x_{0j_1}|^2 = |x_{0j_0}|^2 \sum_{j_1 \in J_0} |\xi_{0j_0j_1}|^2$$

thus

$$|x_{0j_0}| = \left[\sum_{j_1 \in J_0} |\xi_{0j_0j_1}|^2\right]^{-1} \|x_0\|^2 = \left[\sum_{j_1 \in J_0} |\xi_{0j_0j_1}|^2\right]^{-1}. \quad (3)$$

Similarly, for each $i_0 \in I_1$:

$$|x_{1i_0}| = \left[\sum_{i_1 \in I_1} |\xi_{1i_0i_1}|^2\right]^{-1} \|x_1\|^2 = \left[\sum_{i_1 \in I_1} |\xi_{1i_0i_1}|^2\right]^{-1}. \quad (4)$$

Now, since X is a solution of (1), from (2) we have $\forall i \in I_1$:

$$\xi_{0j_0j_1} = \frac{x_{0j_1}}{x_{0j_0}} = \frac{\frac{z_{\phi_{nn_0}(i,j_1)}}{x_{1i}}}{\frac{z_{\phi_{nn_0}(i,j_0)}}{x_{1i}}} = \frac{z_{\phi_{nn_0}(i,j_1)}}{z_{\phi_{nn_0}(i,j_0)}}$$

or, in summary:

$$j_0, j_1 \in J_0 \ ,\ i \in I_1 \implies \xi_{0j_0j_1} = \frac{z_{\phi_{nn_0}(i,j_1)}}{z_{\phi_{nn_0}(i,j_0)}} \quad (5)$$

and similarly,

$$i_0, i_1 \in I_1 \ , \ j \in J_0 \implies \xi_{1 i_0 i_1} = \frac{z_{\phi_{nn_0}(i_1,j)}}{z_{\phi_{nn_0}(i_0,j)}}. \quad (6)$$

Consequently, the ratios at the right side of condition (5) have constant value for all $i \in I_1$ and the ratios at the right side of condition (6) have constant value for all $j \in J_0$. If any of these conditions is not fulfilled, then z is not $(n-n_0, n_0)$-decomposable. Otherwise, from (5) and (6) we may calculate the ratios $\xi_{0 j_0 j_1}$ and $\xi_{1 i_0 i_1}$ respectively, and from (3) and (4) the absolute values of the entries of x_0 and x_1, provided that all involved vectors are unitary.

III. Simulation of entanglement

A. Von Neumann measurements

For each angle $x \in [0, 2\pi]$ the linear rotation map

$$M_x : \mathbb{H} \to \mathbb{H} \ , \ M_x = \begin{pmatrix} \cos x & \sin x \\ \sin x & -\cos x \end{pmatrix}$$

is unitary, its eigenvalues are $\lambda_0 = -1$, $\lambda_1 = +1$ and the eigenvectors are

$$u_{x0} = \sin\frac{x}{2}\, e_0 - \cos\frac{x}{2}\, e_1 \ , \ u_{x1} = \cos\frac{x}{2}\, e_0 + \sin\frac{x}{2}\, e_1.$$

M_x is an *observable*, namely a self-adjoint linear map. According to the Spectral Theorem, $M_x = -\pi_{L_{x0}} + \pi_{L_{x1}}$ where $\pi_{L_{xi}} : y \mapsto \langle y|u_{xi}\rangle u_{xi}$ is the orthogonal projection along the i-th eigenvector. For any state $y \in \mathbb{H}$, the observable M_x outputs the eigenvalue λ_i with probability $|\langle y|u_{xi}\rangle|^2$. Let $\mu_x(y)$ be the output of M_x at state y. Then μ_x is a *von Neumann measurement*. Over the canonical vectors, we have

$$\begin{aligned} \Pr\left(\mu_x(e_i) = \lambda_j\right) &= |\langle e_i|u_{xj}\rangle|^2 \\ &= \sin^2\frac{x}{2}\delta_{ij} + \cos^2\frac{x}{2}(1-\delta_{ij}). \end{aligned} \quad (7)$$

B. Bell basis

Consider the maximally entangled vectors

$$\forall ij \in [\![0,1]\!]^2 : \ b_{ij} = \frac{1}{\sqrt{2}}\left(e_{j0} + (-1)^i e_{1-j,1}\right).$$

where $e_{ij} = e_i \otimes e_j$ is the ij-th canonical vector of $\mathbb{H}_2$. The set $(b_{i_0 i_1})_{i_0,i_1}$ is called the *Bell basis* of $\mathbb{H}_2$. The matrices

$$\sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad , \quad \sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \ ,$$

$$\sigma_y = \begin{pmatrix} 0 & -\iota \\ \iota & 0 \end{pmatrix} \quad , \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

($\iota^2 = -1$) are called *Pauli matrices*, and they conform a basis of $\mathbb{C}^{2\times 2}$: $\forall A = \begin{pmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{pmatrix} \in \mathbb{C}^{2\times 2}$ $\exists c_0, c_x, c_y, c_z$:

$$A = c_0\sigma_0 + c_x\sigma_x + c_y\sigma_y + c_z\sigma_z$$

namely

$$\begin{aligned} (c_0, c_x, c_y, c_z) &= \frac{1}{2}\big((a_{00}+a_{11}), (a_{01}+a_{10}), \\ &\qquad i(a_{01}-a_{10}), (a_{00}-a_{11})\big). \end{aligned}$$

We number the Pauli matrices as

$$\sigma_{00} = \sigma_0 \ , \ \sigma_{01} = \sigma_x \ , \ \sigma_{10} = \sigma_z \ , \ \sigma_{11} = \sigma_y. \quad (8)$$

Then by letting $\tau_{ij} = (\mathrm{Id}_2 \otimes \sigma_{ij})$ each vector in the Bell basis can be transformed into the others by constant multiples of matrices τ_{ij}. Namely, in table II we put at each entry the matrix $T_{\boldsymbol{\varepsilon\delta}}$ such that $b_{\boldsymbol{\varepsilon}} = T_{\boldsymbol{\varepsilon\delta}} b_{\boldsymbol{\delta}}$.

Let us assume that two parties, Alice and Bob, are applying von Neumann measurements M_{x_0} and M_{x_1} over the first and the second qubit of a Bell vector $b_{i_0 i_1}$ respectively. Then the corresponding outputs are $\mu_{x_k}(e_{i_k})$, $k = 0, 1$. According to relation (7), for any $j_0, j_1 \in \{0,1\}$, we have

$$\Pr\left((\mu_{x_0}(e_{i_0}), \mu_{x_1}(e_{i_1})) = (\lambda_{j_0}, \lambda_{j_1})\right) = \frac{1}{2}\begin{cases} \cos^2\frac{x_0-(-1)^{i_0}x_1}{2}\,\delta_{j_0 j_1} + \sin^2\frac{x_0-(-1)^{i_0}x_1}{2}\,(1-\delta_{j_0 j_1}) & \text{if } i_0 = i_1 \\ \sin^2\frac{x_0-(-1)^{i_0}x_1}{2}\,\delta_{j_0 j_1} + \cos^2\frac{x_0-(-1)^{i_0}x_1}{2}\,(1-\delta_{j_0 j_1}) & \text{if } i_0 \neq i_1 \end{cases} \quad (9)$$

$b_{\boldsymbol{\varepsilon}} \setminus b_{\boldsymbol{\delta}}$	b_{00}	b_{01}	b_{10}	b_{11}
b_{00}	τ_{00}	τ_{01}	τ_{10}	$-i\,\tau_{11}$
b_{01}	τ_{01}	τ_{00}	$-i\,\tau_{11}$	τ_{10}
b_{10}	τ_{10}	$-i\,\tau_{11}$	τ_{00}	τ_{01}
b_{11}	$-i\,\tau_{11}$	τ_{10}	τ_{01}	τ_{00}

TABLE II
Each entry $T_{\boldsymbol{\varepsilon\delta}}$ is such that $b_{\boldsymbol{\varepsilon}} = T_{\boldsymbol{\varepsilon\delta}} b_{\boldsymbol{\delta}}$.

C. Steiner protocol

Let Alice and Bob share a sequence $(t_n)_n$ of uniformly distributed random numbers in the real interval $[0,1]$.

Suppose that Alice knows a number $y_0 \in [0,1]$ and Bob a number $y_1 \in [0,1]$.

Let Alice generate another sequence $(s_n)_n$ of uniformly distributed random numbers in $[0,1]$. Alice finds $k_0 = \min\left(k | s_k \leq |\cos\left(2\pi(t_k - y_0)\right)|\right)$ and sends it to Bob. Alice outputs the value

$$a_0 = \mathrm{Sgn}\left(\cos\left(2\pi(t_{k_0} - y_0)\right)\right).$$

Bob receives k_0 and outputs the value

$$a_1 = \mathrm{Sgn}\left(\cos\left(2\pi(t_{k_0} - y_1)\right)\right).$$

Then $\Pr\left(a_0 = a_1\right) = \cos^2\left(2\pi(y_0 - y_1)\right)$. Thus whenever $y_0 = y_1$, Alice and Bob would output the same values. Namely, this protocol is simulating the measurements $\mu_{2\pi y_k}$ over the entangled state b_{00}, according to relation (9). The common knowledge of sequence $(t_n)_n$ can be considered as a *local hidden variable*.

In order to avoid hidden variables, using the above protocol, Alice may choose a real random variable $t = \sum_{k\geq 1} a_k 2^{-k}$ with density function $p : t \mapsto \frac{\pi}{2}\left|\cos\left(2\pi(t-x)\right)\right|$. If Alice would communicate the exact value of t, she should sent an infinity of bits to Bob. However just a small bits of

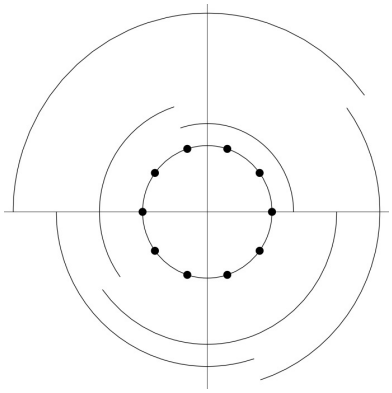


Fig. 1. Sets B_{tj} and C_{tj}, for $t = 0$.

t are sufficient to determine unambiguously the bit $a_1 = \mathrm{Sgn}\left(\cos\left(2\pi(t-y_1)\right)\right)$. Namely the map

$$f_{y_1} : t \mapsto \frac{\pi}{2}\cos\left(2\pi(t-y_1)\right)$$

is integrable (it is continuous on a compact set) and for each $z \in [0,1]$, $w \in [0, \frac{1}{4}]$,

$$\int_{z-w}^{z} f_{y_1} > \frac{1}{4}\int_{z}^{z+w} f_{y_1},$$

thus, having received a few bits of t, say $t' = \sum_{\kappa=1}^{k} a_\kappa 2^{-\kappa}$, Bob may decide whether $\mathrm{Sgn}\left(\cos\left(2\pi(t-y_1)\right)\right)$ has been determined. In general, 5 bits of t will suffice to determine a_1 and this protocol would mimic entanglement without hidden variables by exchanging at most 22 bits.

D. Brassard protocol

Let $[\![0,9]\!]$ consists of the 10 integers from 0 to 9. Through the bijection $x \mapsto e^{2i\pi x}$, the unit interval $[0,1]$ is identified with the unit circle of the complex plane. The collection $V_{10} = \left(e^{i\frac{\pi}{5}j}\right)_{j\in[\![0,9]\!]}$ is the regular decagon in the unit circle in $\mathbb{C}$. For each $j \in [\![0,9]\!]$, let $A_j = \{e^{2i\pi x} \in \mathbb{C} |\ \frac{j}{10} \le x < \frac{j+1}{10}\}$ be the arc joining the j-th and $(j+1)$-th vertices in the regular decagon. Similarly, let for $t \in [0, \frac{3}{10}[$ and for $j \in [\![0,2]\!]$,

$$\begin{array}{lll} \beta_{tj} = \frac{3}{5}j + t & , & B_{tj} = \left\{e^{2i\pi x} \in \mathbb{C} |\ \beta_{tj} \le x < \beta_{t,j+1}\right\}, \\ \gamma_{tj} = \beta_{tj} + \frac{1}{2} & , & C_{tj} = \left\{e^{2i\pi x} \in \mathbb{C} |\ \gamma_{tj} \le x < \gamma_{t,j+1}\right\}. \end{array}$$

In Fig. 1, the inner circle represents the unit circle, the set V_{10} is depicted by bigger points and, going from the most inner radius to outer radii, there appear some sets whose projections over the unit circle are, respectively, B_{t0}, B_{t1}, B_{t2}, C_{t0}, C_{t1}, C_{t2}, with $t = 0$. From Fig. 1 we see that by applying a rotation of $2\pi t$ radians, the following equations hold:

$$\begin{array}{rcl} e^{2i\pi t}A_0 & = & B_{t0} \cap C_{t1} \\ e^{2i\pi t}\left(A_1 \cup A_2\right) & = & B_{t0} \cap C_{t2} \\ e^{2i\pi t}\left(A_3 \cup A_4\right) & = & B_{t1} \cap C_{t2} \\ e^{2i\pi t}A_5 & = & C_{t0} \cap B_{t1} \\ e^{2i\pi t}\left(A_6 \cup A_7\right) & = & C_{t0} \cap B_{t2} \\ e^{2i\pi t}\left(A_8 \cup A_9\right) & = & C_{t1} \cap B_{t2} \end{array}$$

Let $\mathcal{P}_t = \left(A_{j_a} \cap B_{tj_b} \cap C_{tj_c}\right)_{(j_a,j_b,j_c)\in[\![0,9]\!]\times[\![0,2]\!]^2}$ be the induced partition over the unit circle. If t is a multiple of $\frac{1}{10}$ then $\mathcal{P}_t$ consists of the 10 arcs A_j, otherwise it consists of 16 arcs (each of 6 arcs A_j is divided into two arcs). Obviously, for each $x \in [0,1]$ the point $e^{2i\pi x}$ is in the unit circle and there is a unique triple $(j_a, j_b, j_c)(x,t) \in [\![0,9]\!] \times [\![0,2]\!]^2$ such that $e^{2i\pi x} \in A_{j_a(x,t)} \cap B_{tj_b(x,t)} \cap C_{tj_c(x,t)}$. Since $\mathcal{P}_t$ has at most 16 arcs, 4 bits will suffice to determine the triple $(j_a, j_b, j_c)(x,t)$. Let $\boldsymbol{\varepsilon}(x,t) \in \{0,1\}^4$ be the corresponding 4-bit word.

The entanglement simulation protocol is the following. Alice and Bob share a hidden variable $c \in \{0,1\}$, they agree in a parameter $t \in [0, \frac{3}{10}[$. Alice possesses $x \in [0,1]$ and Bob $y \in [0,1]$.

1) Alice calculates $\boldsymbol{\delta} = \boldsymbol{\varepsilon}(x,t)$ and sends it to Bob. She outputs bit $a = c$.
2) Bob calculates the triple (i_a, i_b, i_c) corresponding to $\boldsymbol{\delta}$ and his own triple $(j_a, j_b, j_c) = (j_a, j_b, j_c)(y,t)$.
   a) If $|i_a - j_a| > 2$ then $y = y + \frac{1}{2}$ and $c = 1 - c$;
   b) for each $j \in [\![0,2]\!]$ let $\alpha_j = \beta_{tj}$;
   c) if $j_a \in \{7,8,9,0,1\}$ then for each $j \in [\![0,2]\!]$ let $\alpha_j = \gamma_{tj}$;
   d) if $\exists j \in [\![0,2]\!]$: $\alpha_j \le x, y \le \alpha_{j+1}$ then output $b = c$ else output $b = c$ with probability

$$1 - \frac{3\pi}{5}\sin(2\pi|y - \alpha_j|)$$

   (since there exists $j \in [\![0,2]\!]$ such that α_j lies between x and y).

In order to check the stochastic behaviour, we see that after the transformation made on y it can be assumed that $|x-y| \le \frac{3}{10}$. Thus, $\Pr(i_b \ne j_b) = \frac{10}{3}|x-y|$ and in this case the position of α_j in step (d) of above protocol is uniform between x and y, hence

$$\begin{array}{rcl} \Pr(a=b) & = & \left(1 - \frac{10}{3}|x-y|\right) + \\ & & \frac{10}{3}\int_0^{|x-y|}\left(1 - \frac{3\pi}{5}\sin(2\pi u)\right)du \\ & = & \frac{1}{2}\left(1 + \cos(2\pi|x-y|)\right) \\ & = & \cos^2(\pi|x-y|) \end{array}$$

Then in order to simulate the entanglement, with von Neumann measurements, the local hidden variable shall be augmented in order to allow the communication of $4 = 2^2$ classical bits. In general, for 2^{n+1}-quregisters, let

$$b_{0^{n+1}} = \frac{1}{2^{\frac{1}{2}}}\left(e_{0^{n+1}} + e_{1^{n+1}}\right),$$

then entanglement simulation will require the communication of 2^{n+1} classical bits.

The general Bell basis is

$$\left(b_{\boldsymbol{\varepsilon}} = \frac{1}{2^{\frac{1}{2}}}\left(e_{0\boldsymbol{\varepsilon}'} + (-1)^{\varepsilon_0} e_{1\overline{\boldsymbol{\varepsilon}'}}\right)\right)_{\boldsymbol{\varepsilon}\in\{0,1\}^{n+1}},$$

where $\boldsymbol{\varepsilon} = \varepsilon_0\boldsymbol{\varepsilon}'$.

Let us mention here that another basis consisting of entangled states is

$$\left(\mathbf{c}_{\boldsymbol{\varepsilon}} = \frac{1}{2^{\frac{n}{2}}}\sum_{\boldsymbol{\eta}\in\{0,1\}^n}(-1)^{\varepsilon_0\eta_0} e_{\boldsymbol{\eta}} \otimes e_{(\boldsymbol{\eta}+\boldsymbol{\varepsilon}) \bmod 2}\right)_{\boldsymbol{\varepsilon}\in\{0,1\}^{n+1}}.$$

IV. SUPERDENSE CODING

A. Multidimensional primitive states

Let $k \geq 2$ be a positive integer and $\rho_k = e^{i\frac{2\pi}{k}}$ be the *primitive k-th root of unity*. Let $\mathbb{H}_1^{(k)} = \mathbb{C}^k$ be the k-dimensional complex vector space and $e_0, \ldots, e_{k-1}$ be the vectors in its canonical basis. For any $m, n \in [\![0, k-1]\!]$ let

$$b_{mn} = \frac{1}{\sqrt{k}} \sum_{j=0}^{k-1} \rho_k^{jm} e_j \otimes e_{(j+n)}.$$

Then $B_{k2} = (b_{mn})_{m,n\in[\![0,k-1]\!]}$ is an orthonormal basis of $\mathbb{H}_2^{(k)} = \mathbb{H}_1^{(k)} \otimes \mathbb{H}_1^{(k)}$, called *Bell basis*.

For any $m, n \in [\![0, k-1]\!]$ let $U_{mn} = [u_{mn\mu\nu}]_{\mu,\nu\in[\![0,k-1]\!]}$ be the matrix with entries in the unit circle of $\mathbb{C}$ such that

$$u_{mn\mu\nu} = \rho_k^{m\nu} \delta_{\mu,(\nu+n)}. \tag{10}$$

Observe here that for $k = 2$, $\sigma_{00} = U_{00}$, $\sigma_{01} = U_{01}$, $\sigma_{10} = -\iota U_{10}$ and $\sigma_{11} = U_{11}$ (see eq's (8)). From now on, all indices and superscripts operations are modulo k. Then,

$$U_{mn} e_j = \sum_{\mu=0}^{k-1} \rho_k^{mj} \delta_{\mu,(j+n)} e_\mu = \rho_k^{mj} e_{(j+n)}$$

and, by denoting by $\mathbf{1}_k$ the identity matrix of order k,

$$\begin{aligned}
(\mathbf{1}_k \otimes U_{mn})\, b_{00} &= (\mathbf{1}_k \otimes U_{mn}) \left(\frac{1}{\sqrt{k}} \sum_{j=0}^{k-1} e_j \otimes e_j \right) \\
&= \frac{1}{\sqrt{k}} \sum_{j=0}^{k-1} (\mathbf{1}_k \otimes U_{mn})\, (e_j \otimes e_j) \\
&= \frac{1}{\sqrt{k}} \sum_{j=0}^{k-1} \rho_k^{mj} e_j \otimes e_{(j+n)} \\
&= b_{mn}
\end{aligned}$$

Let $C_k = \left[\delta_{\mu,(\nu+1)}\right]_{\mu,\nu\in[\![0,k-1]\!]}$ be the matrix representing the "rotation" linear map such that $e_\nu \mapsto e_{(\nu+1)}$. Then, from relation (10) we may see that $\forall m, n$:

$$U_{mn} = C_k^n U_{m0} = C_k^n U_{10}^m = C_k^n \left(\operatorname{diag} [\rho_k^\nu]_{\nu\in[\![0,k-1]\!]} \right)^m.$$

Besides,

$$U_{10} C_k = \rho_k C_k U_{10}.$$

Consequently, $U_{10} C_k^p = \rho_k^p C_k^p U_{10}$ and $U_{10}^q C_k^p = \rho_k^q C_k^p U_{10}^q$, which implies $\forall m, n, p, q$: $U_{mn} U_{pq} = \rho_k^{nq} U_{(m+p),(n+q)}$.Thus,

$$\begin{aligned}
(\mathbf{1}_k \otimes U_{mn})\, b_{pq} &= (\mathbf{1}_k \otimes U_{mn}) \circ (\mathbf{1}_k \otimes U_{pq})\, b_{00} \\
&= \rho_k^{nq}\, b_{(m+p),(n+q)}.
\end{aligned} \tag{11}$$

and $\forall m, n, p, q$:

$$\left[\rho_k^{-(m-p)(n-q)} \left(\mathbf{1}_k \otimes U_{(m-p),(n-q)} \right) \right] b_{pq} = b_{mn}. \tag{12}$$

Eq's. (11) and (12) allow *superdense coding*:

1) Alice and Bob agree in a maximally entangled state b_{pq}.
2) Bob applies to his qubit a transformation U_{uv} in order to produce a phase displacement of b_{mn}, according to eq. (11), and sends his qubit to Alice.
3) Knowing both qubits, Alice is in possession of a phase displacement of b_{mn}. She performs a measurement with respect to Bell basis B_{k2}, she recognizes b_{mn} and using eq. (12) she is able to recognize the transformation U_{uv} applied by Bob.

Thus through the transmission of one qubit, Bob may communicate $\log_2 k^2$ classical bits to Alice.

	σ_{00}	σ_{01}	σ_{10}	σ_{11}
σ_{00}	σ_{00}	σ_{01}	σ_{10}	σ_{11}
σ_{01}	σ_{01}	σ_{00}	$-i\,\sigma_{11}$	$i\,\sigma_{10}$
σ_{10}	σ_{10}	$i\,\sigma_{11}$	σ_{00}	$-i\,\sigma_{01}$
σ_{11}	σ_{11}	$-i\,\sigma_{10}$	$i\,\sigma_{01}$	σ_{00}

TABLE III
MULTIPLICATION TABLE OF THE GENERATORS OF PAULI GROUP.

B. A first multiparty protocol

Let $k \geq 2$ be a positive integer, and let $\mathbb{H}_k^{(k)} = \left(\mathbb{H}_1^{(k)}\right)^{\otimes k}$ be the k-fold tensorial power of $\mathbb{H}_1^{(k)} = \mathbb{C}^k$. For any $n_0, n_1, \ldots, n_{k-1} \in [\![0, k-1]\!]$ let

$$b^{(k)}_{n_0 n_1 \cdots n_{k-1}} = \frac{1}{\sqrt{k}} \sum_{j=0}^{k-1} \rho_k^{jn_0} e_j \otimes \bigotimes_{\ell=1}^{k-1} e_{(j+n_\ell)}.$$

Then $B_{kk} = \left(b^{(k)}_{n_0 n_1 \cdots n_{k-1}} \right)_{n_0, n_1, \ldots, n_{k-1} \in [\![0,k-1]\!]}$ is an orthonormal basis of $\mathbb{H}_k^{(k)}$. As in eq. (11), we have

$$\begin{aligned}
&\left(\mathbf{1}_k \otimes \bigotimes_{\ell=1}^{k-1} U_{p_\ell q_\ell} \right) b^{(k)}_{n_0 n_1 \cdots n_{k-1}} \\
&= \rho_k^{\sum_{\ell=1}^{k-1} q_\ell n_\ell}\, b^{(k)}_{(n_0 + \sum_{\ell=1}^{k-1} p_\ell),(q_1+n_1),\cdots,(q_{k-1}+n_{k-1})}
\end{aligned}$$

which gives an equivalent form of eq. (12) in order to recover a map $\left(\mathbf{1}_k \otimes \bigotimes_{\ell=1}^{k-1} U_{p_\ell q_\ell} \right)$ transforming a given register in the Bell basis into another given register in that basis.

However, at present case there are $\left(k^2\right)^{k-1} = k^{2k-2}$ maps of the form $\left(\mathbf{1}_k \otimes \bigotimes_{\ell=1}^{k-1} U_{p_\ell q_\ell} \right)$ and there are k^k registers in Bell basis. One can see that for any two $b^{(k)}_{n_0 n_1 \cdots n_{k-1}}$, $b^{(k)}_{m_0 m_1 \cdots m_{k-1}}$ there are exactly

$$\frac{k^{2k-2}}{k^k} = k^{k-2}$$

maps transforming the first register into the second one. Thus there is no univocity in superdense coding.

C. Use of Pauli maps

Let $R_4 = \langle i \rangle = \{1, i, -1, -i\}$ be the cyclic group generated by i in the multiplicative subgroup S_1, consisting of the unit circle, of $\mathbb{C}$. The Pauli group has 16 elements, $P = R_4 \oplus (\sigma_{\boldsymbol{\varepsilon}})_{\boldsymbol{\varepsilon}\in\{0,1\}^2}$, and its product is determined component-wise by the operation of R_4 and the multiplication depicted at table III.

Let us consider now, the space $\mathbb{H}_3$ of 3-quregisters. The maps of the form $\sigma_{00} \otimes \sigma_{\boldsymbol{\delta}} \otimes \sigma_{\boldsymbol{\varepsilon}}$, $\boldsymbol{\delta}, \boldsymbol{\varepsilon} \in \{0,1\}^2$, applied on the Bell vector $b_{000} \in \mathbb{H}_3$ give R_4-multiples of the other elements in the Bell basis. We may see that

$$(\sigma_{00} \otimes \sigma_{\delta_0 \delta_1} \otimes \sigma_{\varepsilon_0 \varepsilon_1})\, b_{000} \in \mathcal{L}\, (b_{\eta_0 \eta_1 \eta_2})$$

where $\eta_0 = (1-\delta_0)\varepsilon_0 + \delta_0(1-\varepsilon_0)$, $\eta_1 = \delta_1$, $\eta_2 = \varepsilon_1$.

Consider the "controlled-not" map

$$\begin{aligned} CN : \{0,1\}^2 &\to \{0,1\} \ , \\ (\delta,\varepsilon) &\mapsto CN(\delta,\varepsilon) = (1-\delta)\varepsilon + \delta(1-\varepsilon) \end{aligned}$$

and define recursively,

$$\begin{aligned} CN_1(\delta_0,\delta_1) &= CN(\delta_0,\delta_1) \\ CN_{n+1}(\delta_0,\delta_1,\ldots,\delta_{n+1}) &= CN(\delta_0, CN_n(\delta_1,\ldots,\delta_{n+1})) \end{aligned}$$

For each integer $n \geq 1$, CN_n is a balanced map, and consequently card $CN_n^{-1}(0) = 2^n = \text{card}\, CN_n^{-1}(1)$.

One can also see that, for $n = 4$, in $\mathbb{H}_4$:

$$(\sigma_{00} \otimes \sigma_{\delta_0\delta_1} \otimes \sigma_{\varepsilon_0\varepsilon_1} \otimes \sigma_{\zeta_0\zeta_1})\, b_{0000} \in \mathcal{L}\left(b_{\eta_0\eta_1\eta_2\eta_3}\right)$$

where

$$\eta_0 = CN_2(\delta_0,\varepsilon_0,\zeta_0) \ \ , \ \ \eta_1 = \delta_1 \ \ , \ \ \eta_2 = \varepsilon_1 \ \ , \ \ \eta_3 = \zeta_1.$$

In general, for any $n \geq 3$, in $\mathbb{H}_n$:

$$\left(\sigma_{00} \otimes \bigotimes_{j=1}^{n-1} \sigma_{\varepsilon_0^{(j)}\varepsilon_1^{(j)}}\right) b_{0^n} \in \mathcal{L}\left(b_{\eta_0\eta_1\cdots\eta_{n-1}}\right) \tag{13}$$

where

$$\begin{aligned} \eta_0 &= CN_{n-2}(\varepsilon_0^{(1)},\ldots,\varepsilon_0^{(n-1)}) \ \ , \\ \eta_j &= \varepsilon_1^{(j)} \ \ \forall j = 1,\cdots,n-1. \end{aligned} \tag{14}$$

Thus, for $n \geq 4$, for each $\xi \in R_4$ and each $\eta_0,\eta_1,\ldots,\eta_{n-1} \in \{0,1\}$, there are exactly

$$\frac{4^{n-1}}{4\cdot 2^n} = \frac{2^{2n-2}}{2^{n+2}} = 2^{n-4}$$

operators of the form $\left(\sigma_{00} \otimes \bigotimes_{j=1}^{n-1} \sigma_{\varepsilon_0^{(j)}\varepsilon_1^{(j)}}\right)$ mapping b_{0^n} into $\xi b_{\eta_0\eta_1\cdots\eta_{n-1}}$. If this last quregister is measured with respect to the Bell basis, then it is produced $b_{\eta_0\eta_1\cdots\eta_{n-1}}$. Consequently, after measurement on the Bell basis, each $b_{\eta_0\eta_1\cdots\eta_{n-1}}$ is obtained in 2^{n-2} possible ways.

Just as an example, for the case $n = 4$, there are $4^{n-1} = 64$ triplets $\left(\varepsilon_0^{(j)}\varepsilon_1^{(j)}\right)_{j=1}^{3}$ and there are $4\cdot 2^n = 64$ pairs $(\xi, b_{\eta_0\eta_1\eta_2\eta_3})$. Thus each pair is obtained just once ($2^{n-4} = 1$), but each direction $b_{\eta_0\eta_1\eta_2\eta_3}$ appears four times ($2^{n-2} = 4$). A quick glimpse in the indices allows to check that relations (14) do hold indeed.

Given $\eta_0,\eta_1,\ldots,\eta_{n-1} \in \{0,1\}$, from relations (14), it is direct to find the corresponding operator satisfying (13), and for any other $\theta_0,\theta_1,\ldots,\theta_{n-1} \in \{0,1\}$ through the operation in Pauli group it is also direct to find $\left(\varepsilon_0^{(j)}\varepsilon_1^{(j)}\right)_{j=1}^{n-1}$ such that

$$\left(\sigma_{00} \otimes \bigotimes_{j=1}^{n-1} \sigma_{\varepsilon_0^{(j)}\varepsilon_1^{(j)}}\right) b_{\eta_0\eta_1\cdots\eta_{n-1}} \in \mathcal{L}\left(b_{\theta_0\theta_1\cdots\theta_{n-1}}\right) \tag{15}$$

D. Another multiparty protocol

A straight generalization of *superdense coding* is the following:

1) Alice and $n-1$ correspondents B_j, $1 \leq j \leq n-1$, agree in a maximally entangled state $b_{\eta_0\eta_1\cdots\eta_{n-1}} \in \mathbb{H}_n$.
2) Each correspondent B_j applies to his qubit a transformation $\sigma_{\varepsilon_0^{(j)}\varepsilon_1^{(j)}}$. These actions produce a phase displacement of an entangled quregister $b_{\theta_0\theta_1\cdots\theta_{n-1}}$, according to eq. (15). Each B_j sends his qubit to Alice.
3) Knowing all qubits, Alice is in possession of a phase displacement of $b_{\theta_0\theta_1\cdots\theta_{n-1}}$. She performs a measurement with respect to Bell basis, she recognizes $b_{\theta_0\theta_1\cdots\theta_{n-1}}$ and using relations (14) she is able to recognize the transformations applied by her correspondents, provided that a disambiguation criterion for the first equation in (14) has been agreed.

An elementary disambiguation criterion is the following:

$$\begin{aligned} \eta_0 = 0 &\implies (\varepsilon_0^{(1)},\ldots,\varepsilon_0^{(n-2)},\varepsilon_0^{(n-1)}) = (0,\ldots,0,0) \\ \eta_0 = 1 &\implies (\varepsilon_0^{(1)},\ldots,\varepsilon_0^{(n-2)},\varepsilon_0^{(n-1)}) = (0,\ldots,0,1) \end{aligned}$$

Thus through the transmission of one qubit by each correspondent, Alice is able to receive $2(n-1)$ classical bits from her correspondents.

V. Final comments

As a summary of the above exposition, I would like to point out the following remarks:

- Formal Hilbert space analysis may help to introduce mathematicians and computer scientists into quantum computing concepts
- The sketched procedures to decide entangled quantum registers are rather conventional, for a full exposition of entanglement see [7]
- Entanglement simulation aims to implement in a probabilistic way superdense coding within conventional deterministic computing devices
- The matrix sequence $(U_{mn})_{mn}$, see eq. (10), generalizes the Pauli matrices, although they are slightly different for the 2-quantum levels case
- Superdense coding is a characteristic feature of quantum computing and it can be extended to many quantum levels as well as broadcast in a multiparty context.

References

[1] N. J. Cerf, N. Gisin, S. Massar, and S. Popescu, "Simulating maximal quantum entanglement without communication," *Phys. Rev. Lett.*, vol. 94, p. 220403, Jun 2005. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.94.220403

[2] G. Brassard, L. Devroye, and C. Gravel, "Remote sampling with applications to general entanglement simulation," *Entropy*, vol. 21, no. 1, 2019. [Online]. Available: https://www.mdpi.com/1099-4300/21/1/92

[3] G. Brassard, R. Cleve, and A. Tapp, "Cost of exactly simulating quantum entanglement with classical communication," *Physical Review Letters*, vol. 83, no. 9, p. 1874?1877, Aug. 1999. [Online]. Available: http://dx.doi.org/10.1103/PhysRevLett.83.1874

[4] S. Koudia, "The quantum internet: an efficient stabilizer states distribution scheme," *Physica Scripta*, vol. 99, no. 1, p. 015115, dec 2023. [Online]. Available: https://doi.org/10.1088/1402-4896/ad1565

[5] E. Sutcliffe and A. Beghelli, "Fidelity-aware multipath routing for multipartite state distribution in quantum networks," *IEEE Transactions on Quantum Engineering*, vol. 6, pp. 1–18, 2025.

[6] C. Branciard, N. Brunner, H. Buhrman, R. Cleve, N. Gisin, S. Portmann, D. Rosset, and M. Szegedy, "Classical simulation of entanglement swapping with bounded communication," *Physical Review Letters*, vol. 109, no. 10, Sep. 2012. [Online]. Available: http://dx.doi.org/10.1103/PhysRevLett.109.100401

[7] I. Bengtsson and K. Życzkowski, *Geometry of Quantum States: An Introduction to Quantum Entanglement*, 2nd ed. Cambridge University Press, 2017.

Statistical measures in LLM text Watermark detection

Jose Luis Martin-Navarro
VRAIN - Valencian Research Institute for Artificial Intelligence. Universitat Politècnica de València. Spain
Aalto University - Finland
jomarna6@upv.es

***Abstract*—Detecting machine-generated text is a relevant problem due to the widespread adoption of Large Language Models. Between all the detection techniques, watermarking stands out due to the direct verification of the origin and its precision. To compare the different watermark methods, it is crucial to measure its effectiveness in discerning between text with a watermark or without it. However, the diversity of watermarking approaches difficulties its comparison.**

This work analyses the most used statistic measures in watermarking: the z-score and Area under the Receiver Operating Characteristic curve. As a result, the analysis revealed incorrect usage in popular watermarking work, responsible to a mismatch in the method evaluation. Correcting this issue not only clarifies the performance of the method but also highlights the importance of consistent statistical interpretation when evaluating watermarking systems. Our analysis provides guidance for future studies to ensure methodological rigor and comparability across results.

***Index Terms*—Watermark, LLM, z-score, AUROC**

I. The problem of detecting synthetic text

The popularity of generative machine learning technologies such as Large Language Models (LLM) has brought a new set of challenges. One of them is discerning whenever a text has been generated by a human (organic) or by a machine (synthetic), called synthetic text detection. This is a problem with direct security implications in the form of fraud detection and LLM's collapse prevention. An example of fraud detection can be academic fraud, where a researcher submits a fake report, with experiments and results machine generated. An example of LLM's collapse can occur if a LLM model is trained with data that is synthetic, for example when using publicly available data.

To address this issue, researchers have proposed a variety of synthetic text detectors. These detectors can be grouped into four distinct approaches: neural network-based detection, zero-shot detection, information retrieval, and watermarking. Neural network-based detectors [1, 2] leverage deep learning models trained on huge datasets to identify patterns in synthetic text.

Zero-shot detection [3, 4, 5] uses models as well, but unlike the neural network based approach, it relies only on general language understanding, not prior training.

Information retrieval detection [6] compares the data to be examined with extensive databases of known AI outputs to identify matches or similarities. Watermarking [7, 8, 9, 10], as opposed to the previous techniques, embeds identifiable markers when the synthetic text is generated, allowing a direct verification of the origin rather than a post-hoc analysis or pattern recognition, which is influenced by the quality of available data to recognize the patterns.

In this exploratory work, we review recent developments in watermark detection in the context of LLM generated text, with special focus on the statistic measures employed. As a result, we identify a misuse of the z-score for comparing watermark techniques, paving the work for further analysis of other statistic measures.

II. Watermarking properties and processes

Its direct verification approach makes watermarking an interesting technique to solve synthetic text detection, however it comes with its own set of challenges. The principal properties of watermarking are defined by [8] as follows:

- Watermarks should be identified just through an algorithmic process, with no necessity of prior knowledge of the model parameters or access to the language model API. This permits open-source watermarking detection even with close-source models, with the detection being cost and time effective, as the LLM does not require loading or execution.
- The generation of the watermarks should be possible without retraining the model, making the generation of watermarks time and cost effective.
- In addition, the detection should only use a contiguous portion of the generated text to detect the watermark. This ensures that the watermark remains detectable when only a slice of the generation is used to create a larger document.
- Related to the resilience of the watermark, the watermark should persist unless a significant fraction of the generated tokens are modified.
- Finally, the detection should be calculated with a rigorous statistical measure of confidence.

Authors have proposed watermarking techniques with diverse types of embedding based on these properties. An embedding technique that has gained a lot of

attention comes from KGW [8], which proposed the partition of the vocabulary into a ”green list” and a ”red list” based on a hash of the last token processed (KGW-hard) and some weights (hardness parameter) to improve the quality of the chosen tokens. This has been extended by several authors, for example SWEET [11] adapts the embedding of KGW to increase the quality of the watermarked text. The hardness parameter in KGW is used in the embedding to balance the proportion of green tokens without damaging the quality of the output (for example leaving out an important token because it lands in the red list). On SWEET, tokens with low entropy are ignored instead. By carefully choosing the threshold, SWEET is able to improve the quality of the generated text, especially for code-generation models. Another work [12] uses a similar watermarking model that embeds first a message and then its signature, making it publicly verifiable. There are other watermarking proposals who differ from KGW, for example proposing error-correcting codes as part of the watermark encoding to resist edits attempting to remove it [13].

To be able to compare the different proposed watermarks, one needs to analyze their accuracy on detecting watermarked text. A popular method among researchers [8, 9, 14, 15] is to evaluate the precision of the detection by analyzing if a measurement (e.g. amount of green tokens) fits into the distribution known in organic text or not (z-score). To complete the evaluation the rate of false positives and negatives is analyzed using AUROC. Other works differ on their evaluation, for example [12] performs the evaluation on the watermark of characters instead of tokens. Another work in the literature [13] proposes an interesting theoretical approach, however they do not provide an experimental evaluation. This work focuses on the works that use the z-score and AUROC, analyzing its usage and whenever it is possible to compare methodologies that use the same statistics on the detection.

Other evaluations have been proposed to extend the accuracy of the detection, measuring the robustness of the watermark against manipulations on the watermarked text. Kirchenbauer et al. [16] considered the watermark precision on some input with certain level of paraphrasing done by an attacker. In [13], due to the error-correcting code using in the embedding, they calculated the edit-distance resistance of the watermark to non-adversarial modifications. It is left for future work the analysis of how active manipulation on watermarked text affects the detection depending on the statistic measure.

III. Watermark detection statistics

The z-score [17]is an statistics test that allows to decide between two complementary hypothesis (hypothesis and null hypothesis), and it tests the deviation from a population mean distribution that is already known. In the context of watermarking, the z-score is usually applied as a one-sample location test [8, 14, 15, 10, 11, 9]. We now proceed to describe its application using the vocabulary partition technique (Fig. 1). In this watermarking technique, the text under study is classified depending on a partition of the vocabulary, counting the tokens of the text that belong to the ”green” list. In this case, the null hypothesis corresponds to a text that is organic. Organic text is assumed to be generated by a person writing normally, without being aware of the green list. Since the words

Prompt: Who built the first chess playing machine?

Human or non watermarked text

If you meant the first true (non-fraudulent) chess-playing machine, that title is usually given to Leonardo Torres y Quevedo, who built an electromechanical chess-playing device called El Ajedrecista in 1912 . It could automatically play—and win—the endgame of king and rook versus king.

z-score: 0.152
p-value: 0.8788
Result: No watermark

Watermarked text

The first genuine automated chess-playing machine was built in 1912 by Leonardo Torres y Quevedo, a Spanish engineer. He created ’ El Ajedrecista ’ (The Chess Player) , which could actually play an endgame of king and rook versus king automatically using electromagnets beneath the board.

z-score: 2.598
p-value: 0.0094
Result: Watermark

Fig. 1. Example of watermarked text verification using the vocabulary partition technique. If the partitions are created randomly (KGW), the organic text should have a 50% proportion of green and red tokens, as opposed to the watermarked text that will include more tokens from one of the partitions.

on the list are random, it is possible to conclude that the distribution of green tokens is known, and it is roughly 50% with a big enough sample. The complementary hypothesis is that the text is watermarked. With the z-score, and given a certain error threshold, it is possible to conclude if the proportion of green tokens in a sample does not match the proportion of green tokens from the known organic distribution (one-sample location).

The z-score fits the problem of detecting a watermark in a text sample. However, it can only be used to compare watermark methods if both methods fulfill the same requirements. In particular, the most important assumption needed to apply the z-score is that the distribution of the null hypothesis is known (organic text). In practice we should know what is the standard deviation, and measure it if there is available data.

The Receiver Operating Characteristic curve (ROC) is a plot of the true positive rate (y-axis) against the false positive rate (x-axis) at varying classification thresholds [18]. This plot illustrates the performance of a binary classification model. A classifier picking at random is depicted as the diagonal line, where results above the diagonal depict better than random classification results. Thus, the closer to the upper left corner (0,1) the better the model performs.

The Area under the Receiver Operating Characteristic curve (AUROC) also seen as Area under the curve (AUC) represents the probability that the model, given a randomly chosen positive and negative example, will rank the positive higher than the negative. It is also called the degree of separability.

It is a measurement used in the machine learning community for model comparison due to the coherence on the aggregated classification performance. Since the detection of the watermarked can be phrased as a binary classification problem, the AUROC is used in watermark papers [9, 16, 11] to represent the sensitivity of the resulting hypothesis, complementing the detection statistics of the z-score.

However, AUROC is not a perfect indicator. It can be noisy, and leaves out the negative predictive value, since it only compares the true positive rate against the false positive rate [18].

IV. State of the art analysis

Following is the analysis of the statistics of popular watermark works:

- KGW [8] uses the test to compare if the distribution of the "green" tokens on the sample satisfies the null hypothesis (non-watermarked text) or does not satisfy it (watermarked text). The test can be applied here because a natural writer is expected to violate the red list rule with half of their tokens. If the null hypothesis is true, given T number of tokens, the mean of tokens in the list will be $T/2$ with a variance of $T/4$. Although a dataset of text is used to analyze the perplexity and the configuration parameter, **the actual distribution of organic text is not measured**.
- Fu et al. [9] use the distribution of green tokens as a detection metric, like KGW and SWEET. In particular, they propose a watermark embedding that modifies the token partition, prioritizing the inclusion of content related to the input based on semantic analysis. They report a mix of results in their evaluation, where they compare their proposal with KGW. The z-score of their watermarking technique shows a significant improvement, whereas the AUROC comparison shows a better result for KGW.
- Yang et al. [10] uses a z-score with a slightly different null assumption than [8, 9]. In their case the null hypothesis is that the observed encoding occurs randomly (human generated). **They verified that the probability of their encoding with organic data is close to 50%**, which validates the theoretical assumption of the null hypothesis distribution.
- SWEET [11] adapts the detection method of KGW. They provide a z-score calculation that includes the threshold as $z_{\text{threshold}}$, and predict the lower bound comparison between SWEET and KGW. However, **they do not calculate if the null hypothesis mean is the same or has changed with their threshold method**. This does not necessarily mean the z-score is incorrectly used, since on the AUROC evaluation their model seems to perform as predicted.
- Zhang et al. [15] analyze the elimination of the watermark from the watermarked text through quality preserving perturbation methods. To assess their approach, they measure the drop of the z-score in KGW [8] and Unigram [14]. Since both works use the same approach in their z-score measurement, using them to compare the quality drop of the elimination technique is accurate.

V. Detail discussion

Given the disparity of results and the nature of the approach, we took a closer look at the evaluation in [9]. In particular, the reported results are the z-score and the AUC scores of KGW and their watermark model with different settings, depicted in Fig. 2 and Fig. 3 respectively. Although in Fig. 2 the z-score regarding the ratio of correct tokens is higher in their proposed method compared with KGW, in the AUROC evaluation the performance switches. Since AUROC is a more coherent aggregated classification metric, this lead us to think that the problem could originate in the z-score measurement.

The proposed watermark embedding uses semantic knowledge to prioritize the inclusion of content related to the input into the green list partition. This approach can improve the overall quality of the generated text,

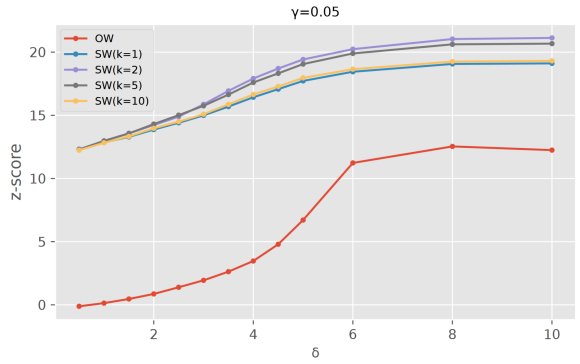


Fig. 2. Watermark detection: Average z-score under different δ settings (x-axis). The hardness parameter δ affects how much the watermarking is allowed to deviate from the green token list. A high δ makes the watermark easier to detect but decreases the quality of the output. The red dataset (OW) is [8] model meanwhile the others (SW) correspond to [9]. Image from [9]

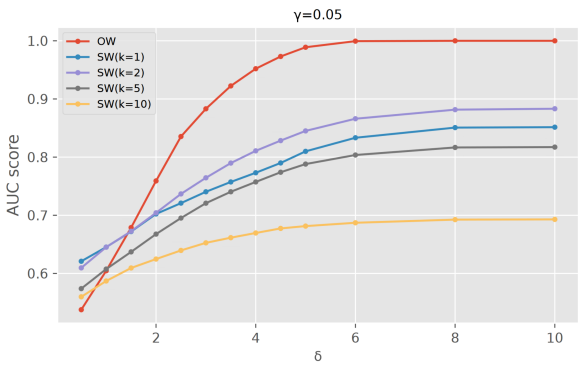


Fig. 3. Watermark detection: AUC scores under different δ settings. Higher AUC scores indicates a better detection performance. Image from [9]

with an increased chance of picking a token that is both green and semantically relevant to the context of the task. However, this approach also has an unintended consequence. By changing how the "green token" list is partitioned, from random to an informed way (with semantic knowledge), they are also affecting the assumption of the null hypothesis of the z-score, which assumes that a human generated text will have an equally probable number of tokens in and out of the list (mean $T/2$ and variance $T/4$).

In their discussion, the authors acknowledge the disparity of results, attributing it to the fact that humans also use semantic knowledge to decide which word they write after another. We believe that this implies a more general finding, where watermarking methods modifying the distribution of the partitions need to assess the distribution of organic text, making sure that the mean and average align with the null hypothesis used when calculating the z-score.

Our finding align with the work by [10], where they assessed that their encoding did not modify significantly the organic data distribution between the partitions. For those cases where the partition is affecting the distribution probability of organic text, the AUROC measurement can still be used to compare with other watermarking techniques.

VI. Conclusions

This work presents the analysis of different watermarking methods that use the z-score statistic measurement for watermarking detection. By analyzing the specifics of the z-score, its assumptions, and how it is used by watermark detectors, we gained a better understanding on its usage for watermarking and when it is fair to use it to compare models. Although most of the work uses the z-score correctly, we identified an incorrect usage of the z-score in [9], where the informed partition of the token list affected also the null hypothesis the z-score is based on. The disparity in watermark detection precision was acknowledged in the paper, attributed to the fact that humans also use semantic knowledge to decide which word they write after another. We expand this notion and recommend that watermarking papers include the null hypothesis mean and variance values, which is especially relevant for methods altering the token partitions. If the partition is close to random sampling, the z-score is a perfect metric to compare with other solutions. Otherwise, the AUROC score can still be used to rank different watermark methods.

It is left for future work the measurement of the null hypothesis mean and variance values for the semantic partition proposed in [9]. Such analysis can help determine the accuracy of AUROC when comparing the work with KGW. Further research is also needed to explore how other statistic analysis can be applied to measure the accuracy of watermark verification, for example for short and repetitive text where the z-score distribution assumption has been proved to fail [19].

Acknolwedgments

Supported by INCIBE's Chair funded by the EU-NextGenerationEU through the Spanish government's Plan de Recuperación, Transformación y Resiliencia.

References

[1] OpenAI, "New AI classifier for indicating AI-written text," 2023. [Online]. Available: https://openai.com/index/new-ai-classifier-for-indicating-ai-written-text/

[2] M. A. Panhwar, K. A. Memon, A. Abro, D. Zhongliang, S. A. Khuhro, and S. Memon, "Signboard Detection and Text Recognition Using Artificial Neural Networks," in *2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, Jul. 2019, pp. 16–19, iSSN: 2377-844X. [Online]. Available: https://ieeexplore.ieee.org/document/8784625/?arnumber=8784625

[3] S. Gehrmann, H. Strobelt, and A. M. Rush, "GLTR: Statistical Detection and Visualization of Generated Text," Jun. 2019, arXiv:1906.04043

[cs]. [Online]. Available: http://arxiv.org/abs/1906.04043

[4] D. Ippolito, D. Duckworth, C. Callison-Burch, and D. Eck, "Automatic Detection of Generated Text is Easiest when Humans are Fooled," May 2020, arXiv:1911.00650 [cs]. [Online]. Available: http://arxiv.org/abs/1911.00650

[5] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, and C. Finn, "DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature," Jul. 2023, arXiv:2301.11305 [cs]. [Online]. Available: http://arxiv.org/abs/2301.11305

[6] K. Krishna, Y. Song, M. Karpinska, J. Wieting, and M. Iyyer, "Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense," Oct. 2023, arXiv:2303.13408 [cs]. [Online]. Available: http://arxiv.org/abs/2303.13408

[7] S. Aaronson and H. Kirchner, "Watermarking gpt outputs," 2022. [Online]. Available: https://www.scottaaronson.com/talks/watermark.ppt

[8] J. Kirchenbauer, J. Geiping, Y. Wen, J. Katz, I. Miers, and T. Goldstein, "A Watermark for Large Language Models," in *Proceedings of the 40th International Conference on Machine Learning*. PMLR, Jul. 2023, pp. 17 061–17 084, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v202/kirchenbauer23a.html

[9] Y. Fu, D. Xiong, and Y. Dong, "Watermarking Conditional Text Generation for AI Detection: Unveiling Challenges and a Semantic-Aware Watermark Remedy," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 16, pp. 18 003–18 011, Mar. 2024, number: 16. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/29756

[10] X. Yang, K. Chen, W. Zhang, C. Liu, Y. Qi, J. Zhang, H. Fang, and N. Yu, "Watermarking Text Generated by Black-Box Language Models," May 2023, arXiv:2305.08883 [cs]. [Online]. Available: http://arxiv.org/abs/2305.08883

[11] T. Lee, S. Hong, J. Ahn, I. Hong, H. Lee, S. Yun, J. Shin, and G. Kim, "Who Wrote this Code? Watermarking for Code Generation," Jul. 2024, arXiv:2305.15060 [cs]. [Online]. Available: http://arxiv.org/abs/2305.15060

[12] J. Fairoze, S. Garg, S. Jha, S. Mahloujifar, M. Mahmoody, and M. Wang, "Publicly-Detectable Watermarking for Language Models," May 2024, arXiv:2310.18491 [cs]. [Online]. Available: http://arxiv.org/abs/2310.18491

[13] N. Golowich and A. Moitra, "Edit Distance Robust Watermarks for Language Models," 2024.

[14] X. Zhao, P. Ananth, L. Li, and Y.-X. Wang, "Provable Robust Watermarking for AI-Generated Text," Oct. 2023, arXiv:2306.17439 [cs]. [Online]. Available: http://arxiv.org/abs/2306.17439

[15] H. Zhang, B. L. Edelman, D. Francati, D. Venturi, G. Ateniese, and B. Barak, "Watermarks in the Sand: Impossibility of Strong Watermarking for Generative Models," Nov. 2023, arXiv:2311.04378 [cs]. [Online]. Available: http://arxiv.org/abs/2311.04378

[16] J. Kirchenbauer, J. Geiping, Y. Wen, M. Shu, K. Saifullah, K. Kong, K. Fernando, A. Saha, M. Goldblum, and T. Goldstein, "On the Reliability of Watermarks for Large Language Models," May 2024, arXiv:2306.04634 [cs]. [Online]. Available: http://arxiv.org/abs/2306.04634

[17] M. D. Ugarte, A. F. Militino, and A. T. Arnholt, *Probability and Statistics with R*. CRC Press, Apr. 2008, google-Books-ID: 0zjcBQAAQBAJ.

[18] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," *BioData Mining*, vol. 16, no. 1, p. 4, Feb. 2023. [Online]. Available: https://doi.org/10.1186/s13040-023-00322-4

[19] P. Fernandez, A. Chaffin, K. Tit, V. Chappelier, and T. Furon, "Three Bricks to Consolidate Watermarks for Large Language Models," in *2023 IEEE International Workshop on Information Forensics and Security (WIFS)*, Dec. 2023, pp. 1–6, iSSN: 2157-4774. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10374576

An extended approach towards root-based attacks against PLWE instances via trace operators

Beatriz Barbero Lucas[1], Iván Blanco Chacón[2], Raúl Durán Díaz[2],
Rodrigo Martín Sánchez-Ledesma[3,4], and Rahinatou Yuh Njah Nchiwo[5]

[1]University College Dublin, Dublin, Ireland, beatriz.barberolucas@ucdconnect.ie
[2]Universidad de Alcalá, Alcalá de Henares, Spain, {ivan.blancoc,raul.duran}@uah.es
[3]Universidad Complutense de Madrid, Madrid, Spain, rodrma01@ucm.es
[4]Indra Sistemas de Comunicaciones Seguras, Madrid, Spain, rmsanchezledesma@indra.es
[5]Aalto University, Espoo, Finland, rahinatou.njahepousenchiwo@aalto.fi

***Abstract*—The present work is an extended abstract of [1], which deals with the robustness of the Polynomial Learning With Errors problem extending previous results in [2]. We explain here the attacks presented therein focusing on the case where the defining polynomial of degree N has a root of degree $n < N$, thus allowing us to determine further previously-not-detected dangerous instances. Finally, we provide computational evidences to back up these results.**

***Index Terms*—PLWE, PQC, Trace map, Number Theory**

I. Introduction

Lattice-based cryptography has achieved an extraordinary success in the last years, especially since the National Institute for Standards and Technology launched a public contest in 2017 to standardize different sets of quantum-resistant primitives. With five candidates selected for standardization, three of them belong to the lattice-based category (ML-KEM for Key Encapsulation, ML-DSA and FN-DSA for Digital Signatures). These suites are based on several *learning problems* over certain large-dimensional vector spaces over finite fields, some of them with an extra ring structure.

In the present work, we are concerned in particular with the Polynomial Learning With Errors problem, which was introduced in [3]. Let us denote by R the quotient ring $R := \mathbb{Z}[x]/(f(x))$ with $f(x) \in \mathbb{Z}[x]$ of degree N, monic and irreducible over $\mathbb{Z}[x]$, and let $q \geq 2$ be a prime. Set $R_q := \mathbb{F}_q[x]/(f(x))$ and consider an R_q-valued uniform random distribution U and an R_q-valued discrete Gaussian distribution χ_{R_q}. The PLWE problem in decision form, the version we investigate in this work, consists in guessing, with non-negligible advantage, if a set of pairs $\{(a_i(x), b_i(x))\}_{i\geq 1} \subseteq R_q \times R_q$ of arbitrary size has been sampled either from $U \times U$ or from the PLWE distribution, where each sample is defined as $(a_i(x), b_i(x) = a_i(x)s(x) + e_i(x))$, with $e_i(x)$ sampled from χ_{R_q} and $s(x) \in R_q$ a secret value.

The main contribution of the present work is to offer an extension of the attacks developed by [2] to the case in which $f(x)$ has a zero-trace root $\alpha \in \mathbb{F}_{q^n} \setminus \mathbb{F}_q$. This condition allows to transfer the original attacks to higher degree extensions of $\mathbb{F}_q$, thus generalizing the results presented in [2].

The key tool of these extensions is to combine the evaluation-at-α map with the trace map [4]. This composition of maps allows one to project R_q to a subring $R_{q,0}$ of large enough dimension over which the PLWE problem turns solvable in expected polynomial time with overwhelming probability of success. Then, the composition is suitably applied to each *distinguishability condition* in [2], depending upon the attack to be extended.

II. Original root-based attacks against PLWE

In [2] and [5], the authors describe an attack against the PLWE problem attached to a quotient ring R_q and a prime q, if there exists a simple root $\alpha \in \mathbb{F}_q$ (i.e. $n = 1$) such that either

a) $\alpha = 1$, or
b) α has *small set of possibilities* modulo q, or
c) α has *small residue* modulo q.

For any of the cases above, the idea of the attacks is the same: the samples to be distinguished are transferred from R_q into $\mathbb{F}_q$ via a certain morphism Φ, in such a way that *distinguishability features* are transferred as well. Each particular feature to be tried is represented by one of the cases above. In this work, we focus on the first two ones. A complete analysis of all the attacks can be found in [1].

Following the steps of [2], the present work also assumes centered Gaussian variables $\mathcal{G}_\sigma$ of standard deviation σ, and truncated at 2σ. The reader willing to know how to lift this restriction is referred to [1].

A. *Via* $\alpha = 1$

Assume that $\alpha = 1$ is a root of $f(x)$. Given a PLWE sample $(a(x), b(x) = a(x)s(x)+e(x))$, the error term, $e(x) = \sum_{i=0}^{N-1} e_i x^i$, has its coefficients $e_i \in \mathbb{F}_q$ sampled from χ_{R_q} of small enough standard deviation σ. Applying the evaluation map, we have

$$b(1) - a(1)s(1) = e(1) = \sum_{i=0}^{N-1} e_i,$$

and the sum $\sum_{i=0}^{N-1} e_i$ is hence sampled from a discrete Gaussian variable of standard deviation $\sqrt{N}\sigma$, which according to practical specifications, is of order $\mathcal{O}(q^{1/4})$.

Hence, if one loops over $\mathbb{F}_q$ for a guess for $s(1)$, a right guess g is one such that the value $b(1) - a(1)g$ will belong to the set of integers $\Sigma := [-2\sqrt{N}\sigma, 2\sqrt{N}\sigma] \cap \mathbb{Z}$. Since the probability that $\sum_{i=0}^{N-1} e_i$ falls in this set is 1, we will refer to the set Σ as the *set of possibilities* for this attack.

B. Via the small set of possibilities

The authors of [2] give an attack on the decisional version of PLWE for R_q whenever a root $\alpha \in \mathbb{F}_q$ of $f(x)$ with (small) order r in $\mathbb{F}_q^*$ exists such that the cardinality of the set of all error polynomials evaluated at α, which we denote by Σ, is small enough for the set to be swept through.

Suppose that r is the order of a root $\alpha \in \mathbb{F}_q, \alpha \neq 1$. Again given a PLWE sample $(a(x), b(x) = a(x)s(x) + e(x))$, we have

$$b(\alpha) - a(\alpha)s(\alpha) = e(\alpha) = \sum_{j=0}^{r-1} \sum_{i=0}^{\lfloor \frac{N}{r} \rfloor - 1} e_{ir+j}\alpha^j,$$

where $\lfloor\ \rfloor$ stands for the floor function.

Now each coefficient $e_j^* = \sum_{i=0}^{\lfloor \frac{N}{r} \rfloor - 1} e_{ir+j}$ comes from a discrete Gaussian distribution of 0 mean and standard deviation $\overline{\sigma} := \sqrt{N/r}\sigma$, hence it belongs to the set of integers $T := [-2\overline{\sigma}, 2\overline{\sigma}] \cap \mathbb{Z}$, assuming that Gaussian distributions are truncated to $2\overline{\sigma}$. Then Algorithm 1 gives rise to the *small set attack*.

Input:	A collection S of M samples
	A set Σ of all possible values for $e(\alpha)$
Output:	A guess $g \in \mathbb{F}_q$ for $s(\alpha)$,
	or **NOT PLWE**,
	or **NES** (not enough samples)

- $G := \emptyset$
- `for` $g \in \mathbb{F}_q$ `do`
  - `for` $(a_i(x), b_i(x)) \in S$ `do`
    - `if` $b_i(\alpha) - a_i(\alpha)g \notin \Sigma$ `then next` g
  - $G := G \cup \{g\}$
- `if` $G = \emptyset$ `then return` **NOT PLWE**
- `if` $G = \{g\}$ `then return` g
- `if` $|G| > 1$ `then return` **NES**

Algorithm 1. Small Set Attack over $\mathbb{F}_q$

The reason for the algorithm's applicability is that if the order r of the root α is low and the discrete Gaussian distribution is truncated, then the cardinality of the set Σ of plausible values for $e(\alpha)$ as $e(x)$ runs over the error distribution is small enough with respect to $|\mathbb{F}_q|$. Remember that

$$e(\alpha) = \sum_{j=0}^{r-1} e_j^* \alpha^j.$$

Notice that each term e_j^* follows a Gaussian distribution of mean 0 and variance $\overline{\sigma}^2 = \lfloor \frac{N}{r} \rfloor \cdot \sigma^2$. Therefore, we can ensure that for each $j \in \{0, \ldots, r-1\}$, we have that $|e_j^*| \leq 2\overline{\sigma}$ with probability 1.

Let us define the look-up table

$$\Sigma := \left\{ \sum_{j=0}^{r-1} x_j \alpha^j : \ x_j \in T \right\},$$

namely, Σ becomes the *set of possibilities* for $e(\alpha)$, such that the absolute value of each block e_j^* is bounded by $2\overline{\sigma}$. Hence, if one loops over $\mathbb{F}_q$ for a guess for $s(\alpha)$, a right guess g is one such that the value $b(\alpha) - a(\alpha)g$ will belong to T, for every sample given.

We are now ready to state the following

Proposition 1: Let Σ, M and r be defined as above, and assume $|\Sigma| < q$. Then

1) If Algorithm 1 returns a guess g or **NES**, then the samples are **PLWE** with probability at least $1 - q\left(\frac{|\Sigma|}{q}\right)^M$.
2) If Algorithm 1 returns **NOT PLWE**, then the samples are uniform with probability 1.

III. Extensions via the Trace operator

The applicability of the preceding attacks depends both on the conditions of q and on the order of the roots of the polynomial $f(x)$ in $\mathbb{F}_q$ but, in either case, it is restricted to roots belonging strictly to $\mathbb{F}_q$.

Most PLWE schemes want their polynomials to be fully split in order to optimize polynomial multiplications, typically via NTT. However, this is not always the case and the best-known example is ML-KEM, whose defining polynomial does not fully split in $\mathbb{F}_q$ for the standardized q.

Such scenarios boost the interest of finding ways to generalize the attacks described above to polynomials that do not factor directly over $\mathbb{F}_q$ but rather over a finite field extension. To introduce this new setting, suppose now that the following conditions are met:

- There exists $a \in \mathbb{F}_q$ such that the polynomial $x^n - a$ divides $f(x)$ in $\mathbb{F}_q[x]$ and, therefore, $n < N$.
- The polynomial $x^n - a$ is irreducible over $\mathbb{F}_q[x]$, i.e., $f(x)$ has an irreducible factor of degree n.
- The element a has a small multiplicative order in $\mathbb{F}_q^*$.

We choose $\alpha \in \mathbb{F}_{q^n} \setminus \mathbb{F}_q$ such that its minimal polynomial over $\mathbb{F}_q$ is precisely $x^n - a$. We define now

$$R_{q,0} = \{p(x) \in R_q : p(\alpha) \in \mathbb{F}_q\}.$$

Then we have the following

Proposition 2: The set $R_{q,0}$ is a ring and, as an $\mathbb{F}_q$-vector subspace of R_q, has dimension $N - n + 1$.

To generalize the attacks to roots in arbitrary degree extensions of $\mathbb{F}_q$, we need to *upgrade* the morphism Φ used in the previous section, so that its image still remains in $\mathbb{F}_q$.

In order to achieve this goal, the homomorphism Φ will be the composition of the evaluation homomorphism with the trace map, $\Phi \colon R_q \to \mathbb{F}_q$, defined as

$$\Phi := \mathrm{Tr} \circ \mathrm{ev}_\alpha.$$

In addition to allowing the construction of such a morphism, the trace map also provides other relevant properties. Let us begin with the following

Proposition 3: Suppose that $x^n - a$ divides $f(x)$ over $\mathbb{F}_q[x]$ for some $n < N$. Assume that $x^n - a$ is irreducible over $\mathbb{F}_q[x]$ and let $\alpha \in \mathbb{F}_{q^n} \setminus \mathbb{F}_q$ be a root of $f(x)$. Then, for each $j \geq 1$ such that j is not a multiple of n we have that $\mathrm{Tr}(\alpha^j) = 0$.

This will greatly help with the success probability of the generalized attacks, as it will reduce the noise of the evaluated samples.

Another key feature of the trace operator is that it respects the distribution of the candidate values that will be considered under these attacks:

Proposition 4: The trace operator respects the distribution of the tentative error values $b(\alpha) - g \cdot a(\alpha)$, for any fixed $g \in \mathbb{F}_{q^n}$.

We are ready now to present the generalized attack divided into two stages, each one of them giving rise to a specific algorithm.

A. Attack over the subring $R_{q,0}$

The first stage is carried out by Algorithm 2, whose input is a set of samples S coming from $R_{q,0} \times R_q$ and allowing one to distinguish whether such set is drawn either from the uniform or from the PLWE distribution.

Input:	A set S of M samples $\in R_{q,0} \times R_q$ A set Σ defined in Equation (4)
Output:	**PLWE**, or **NOT PLWE**, or **NES** (not enough samples)

- $G := \emptyset$
- `for` $g \in \mathbb{F}_q$ `do`
  - `for` $(a_i(x), b_i(x)) \in S$ `do`
    - `if` $\frac{1}{n}\left(\mathrm{Tr}(b_i(\alpha)) - a_i(\alpha)g\right) \notin \Sigma$ `then next` g
  - $G := G \cup \{g\}$
- `if` $G = \emptyset$ `then return` **NOT PLWE**
- `if` $|G| = 1$ `then return` **PLWE**
- `if` $|G| > 1$ `then return` **NES**

Algorithm 2. Small Set Attack over finite extensions of $\mathbb{F}_q$

Observe that if $(a(x), b(x) = a(x)s(x) + e(x)) \in R_{q,0} \times R_q$ is a PLWE sample, evaluating it at α yields $\mathrm{Tr}(b(\alpha) - a(\alpha)s(\alpha)) = \mathrm{Tr}(e(\alpha))$. Computing the latter

$$\mathrm{Tr}(e(\alpha)) = \sum_{j=0}^{N'-1} e_{nj} \cdot \mathrm{Tr}(\alpha^{nj}) = n \cdot \sum_{j=0}^{N'-1} e_{nj} \cdot a^j, \quad (1)$$

with $N' = \frac{N}{n}$ and, for simplicity, $n \mid N$. Assuming that r is the multiplicative order of a in $\mathbb{F}_q^*$, let us define $N'' = \frac{N'}{r}$ and, again for simplicity, suppose that $r \mid N'$. Thus we eventually obtain

$$\mathrm{Tr}(e(\alpha)) = n \sum_{i=0}^{r-1} a^i \cdot \sum_{j=0}^{N''-1} e_{n(jr+i)}. \quad (2)$$

Now, two remarks are in order: First, since $a(x) \in R_{q,0}$, this means that $a(\alpha) \in \mathbb{F}_q$, by definition. Then

$$\mathrm{Tr}(b(\alpha) - a(\alpha) \cdot s(\alpha)) = \mathrm{Tr}(b(\alpha)) - a(\alpha)\,\mathrm{Tr}(s(\alpha)).$$

where the only unknown is precisely $\mathrm{Tr}(s(\alpha)) \in \mathbb{F}_q$. This fact allows our algorithm to loop over $\mathbb{F}_q$ rather than over the full ring R_q.

Second, for fixed values n and r, let us define $e_i^* = \sum_{j=0}^{N''-1} e_{n(jr+i)}$, $0 \le i \le r-1$, so that Equation (2) can be written

$$\mathrm{Tr}(e(\alpha)) = n \sum_{i=0}^{r-1} a^i e_i^*. \quad (3)$$

Viewing each sum e_i^* of coefficients as a sample from a centered discrete Gaussian of standard deviation less than or equal to $\sqrt{N''}\sigma$, we can henceforth list all possible elements in truncation, which are at most $4\sqrt{N''}\sigma + 1$.

Equipped with these tools, we can construct a look-up table Σ for all the possible values in Equation (2) as follows:

$$\Sigma := \left\{ \sum_{i=0}^{r-1} a^i x_i : \; x_i \in \left[-2\sqrt{N''}\sigma, 2\sqrt{N''}\sigma\right] \cap \mathbb{Z} \right\}. \quad (4)$$

Observe that

$$|\Sigma| \le \left(4\sqrt{N''}\sigma + 1\right)^r. \quad (5)$$

All together, we have that

Proposition 5: Let Σ, M and r be defined as above, and assume $|\Sigma| < q$. Then:

1) If Algorithm 2 returns **PLWE** or **NES**, then the samples are **PLWE** with probability at least $1 - q\left(\frac{|\Sigma|}{q}\right)^M$.
2) If Algorithm 2 returns **NOT PLWE**, then the samples are uniform with probability 1.

B. Transferring to the ring R_q

The second stage of the attack consists essentially in sampling from the oracle presented to the adversary until they obtain a suitable number of samples from $R_{q,0} \times R_q$. We provide Algorithm 3 in order to achieve this task.

Input:	A distribution X over R_q^2
Output:	A sample $(a(x), b(x)) \in R_{q,0} \times R_q$

- $count := 0$
- `do`
  - $(a(x), b(x)) \xleftarrow{X} R_q^2$
  - $count := count + 1$
- `until` $a(x) \in R_{q,0}$
- `return` $(a(x), b(x)), count$

Algorithm 3. Random variable X_0

Algorithm 3 returns a pair $(a(x), b(x)) \in R_{q,0} \times R_q$ by repeatedly sampling from the original distribution X keeping also track of the number of invocations to X.

Identifying this algorithm with a random variable X_0 with values on $R_{q,0} \times R_q$, we state the following

Proposition 6: Notations as above,

a) If X is uniform over R_q^2 then X_0 is uniform over $R_{q,0} \times R_q$.
b) If X is a PLWE distribution over R_q^2, then X_0 is an $R_{q,0} \times R_q$-valued PLWE distribution.

Hence, the attack can be transferred from $R_{q,0}$ to R_q, provided that a sufficient number of samples can be found in $R_{q,0} \times R_q$.

Notice that the variable *count* stores the expected number of times one has to run X_0 before succeeding. It is important to be able to quantify how costly it is to generate such samples. Fortunately, we have that:

Proposition 7: Notations as before, the variable *count* is distributed as a geometric random variable of first kind with success probability $q^{-(n-1)}$. In particular the expected number of times is just its mean $E[count] = q^{n-1}$

Note that this indeed represents a computational advantage over a "naive" attempt to just go through every guess $g \in \mathbb{F}_q^n$.

C. General decision attack against R_q-PLWE

We are now almost ready to present a general attack, where an R_q-valued random distribution X (an oracle), which can be either uniform or PLWE, is provided to the attacker. The goal of this attacker is to distinguish which one out of the two it actually is. Before proceeding, we need the help of a pair of parameters.

First, the attacker is restricted to query a polynomial number of samples, l, from the oracle; namely, $l = \mathcal{O}(p(N))$, where $p(x)$ is a polynomial of suitable (small) degree and we keep only those $(a_i(x), b_i(x)) \in R_{q,0} \times R_q$ from the full set of samples generated.

Notice that for a given sample in R_q^2, the probability that it actually belongs to $R_{q,0} \times R_q$ is $\tau = 1/q^{n-1}$, hence, the probability that for l samples in R_q^2, at least $k \leq l$ of them belong to $R_{q,0} \times R_q$ can be approximated (if $l\tau > 5$) as

$$P[B(l,\tau) \geq k] \cong P\left[\mathcal{N}\left(l\tau, \sqrt{l\tau(1-\tau)}\right) \geq k\right], \quad (6)$$

where $B(l,\tau)$ stands for a binomial distribution of parameters l and τ and $\mathcal{N}\left(l\tau, \sqrt{l\tau(1-\tau)}\right)$ is a Gaussian distribution of mean $l\tau$ and standard deviation $\sqrt{l\tau(1-\tau)}$. But the right hand side of Equation (6) can be written as

$$P\left[Z \geq \frac{k - l\tau}{\sqrt{l\tau(1-\tau)}}\right],$$

where Z is a standard Gaussian distribution. Hence, the probability that k out of l samples in R_q^2 actually belong to $R_{q,0} \times R_q$ is about $1/2$ if $k \approx l\tau > 5$.

Next, we must choose k so that $1 - (|\Sigma|/q)^k \geq \theta$, where θ is our desired success probability threshold.

We present next Algorithm 4, which is our main result: a general decision attack over an unknown R_q-valued random distribution X which, as stated above, can be either uniform or PLWE. Remark that Algorithm 4 uses Algorithm 3 and Algorithm 2 as subroutines.

Input:	A distribution X over R_q^2, either uniform or PLWE A pair of values $(k,l) \in \mathbb{N}$
Output:	**NOT PLWE**, or **PLWE**, or **NES**, or **FAIL**

- *set* $C := \emptyset$, *count* $:= 0$, $c := 0$
- *do*
  - *set* $(a(x), b(x)), c :=$ Algorithm.3(X, R_q)
  - $C \leftarrow C \cup \{(a(x), b(x))\}$
  - *count* $\leftarrow$ *count* $+ c$
- *until* $|C| = k$ *or count* $> l$
- *if* *count* $\leq l$ *then*
  - *set* $\omega :=$ Algorithm.2(C)
  - *return* ω
- *else*
  - *return* **FAIL**

Algorithm 4. Decision attack against R_q-PLWE

D. Feasibility of the attack

Regarding the feasibility of our attack for explicit parameters, consider first the case when the modulus polynomial is $\Phi_n(x)$, the n-th cyclotomic polynomial. It is well known that for a prime q, letting f be the smallest natural number such that $q^f \equiv 1 \pmod{n}$, we have a factorization of $\Phi_n(x)$ into $\phi(n)/f$ irreducible factors over $\mathbb{F}_q$ of degree f. In particular, if $f = 1$, then $\Phi_n(x)$ is totally split over $\mathbb{F}_q$ and its roots have maximal order n.

More in general (see [6] pag. 202 for details) if $n = p^k$, and $q = 1 + p^A u$ with u coprime to p and $2 \leq A < k$, denoting by $\Omega(p^A)$ the group of p^A-th primitive roots of unity in $\mathbb{F}_q$, we have

$$\Phi_{p^k}(x) = \prod_{\rho \in \Omega(p^A)} (x^{p^{k-A}} - \rho). \quad (7)$$

In this case, for minimal possible order p^2, the attack against $R_{q,0} \times R_q$ is extremely efficient and a small number of samples (about 6) are enough to solve the decisional version of the problem with overwhelming probability. However, the subspace $R_{q,0}$ has a dimension about $N/2$ and the probability that a sample $(a(x), b(x)) \in R_q^2$ actually lies in $R_{q,0} \times R_q$ is of the order of $q^{-N/2}$ making the samples for Algorithm 2 extremely unlikely to occur.

However, the sufficient condition for Algorithm 2 to work is that $|\Sigma| < q$. For some parameter instances, building the table Σ is feasible, so our attack will still succeed. As an example, for $f(x) = x^{316} + 1672x^{314} + 2x^2 + 103347$ and $q = 100003$, the irreducible factor $x^2 + 1672$ divides $f(x)$ and $r := \mathrm{ord}_q(-1672) = 42$. For $\sigma = 0.045$ we have

$$|\Sigma| \leq \left(4\sqrt{\left\lfloor \frac{N^*}{r} \right\rfloor}\sigma + 1\right)^r \approx 89127 < q.$$

Hence, $\frac{|\Sigma|}{q} < 0.8912$ and Proposition 5 predicts that the attack on $R_{q,0} \times R_q$ will succeed at least 75% of times with just 12 samples and 90% with 20 samples.

References

[1] I. Blanco-Chacón, R. Durán-Díaz, and R. M. Sánchez-Ledesma, "A Generalized Approach to Root-based Attacks against PLWE," *Cryptography and Communications: Special Issue QuRCry*, 2025.

[2] Y. Elias, K. E. Lauter, E. Ozman, and K. E. Stange, "Provably Weak Instances of Ring-LWE," in *Advances in Cryptology – CRYPTO 2015*, ser. Lecture Notes in Computer Science, R. Gennaro and M. Robshaw, Eds., no. 9215. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 63–92.

[3] D. Stehlé, R. Steinfeld, K. Tanaka, and K. Xagawa, "Efficient Public Key Encryption Based on Ideal Lattices," in *Advances in Cryptology – ASIACRYPT 2009*, M. Matsui, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 617–635.

[4] B. Barbero-Lucas, I. Blanco-Chacón, R. Durán-Díaz, R. Martín Sánchez-Ledesma, and R. Y. Njah Nchiwo, "Cryptanalysis of PLWE based on zero-trace quadratic roots," arXiv, 2025.

[5] Y. Elias, K. E. Lauter, E. Ozman, and K. E. Stange, "Ring-LWE Cryptography for the Number Theorist," in *Directions in Number Theory*, ser. Association for Women in Mathematics Series, E. E. Eischen, L. Long, R. Pries, and K. E. Stange, Eds., vol. 3. Cham: Springer International Publishing, 2016, pp. 271–290.

[6] H. Wu, L. Zhu, R. Feng, and S. Yang, "Explicit factorizations of cyclotomic polynomials over finite fields," *Designs, Codes and Cryptography*, vol. 83, no. 1, pp. 197–217, 2017.

Anticipando las amenazas cibernéticas: un marco de preparación en cuatro pasos

Jeimy J. Cano M.
Asociación Colombiana de Ingenieros de Sistemas
Colombia
jjcano@yahoo.com

***Resumen*—En un escenario de inestabilidad y transformación acelerada de las amenazas cibernéticas, las organizaciones deben desarrollar capacidades que les permitan no solo anticipar lo que se avecina y afecte sus planes estratégicos, sino también establecer un marco que las guíe para asegurar les permita actualizarse y navegar en un entorno caracterizado por la turbulencia digital y la incertidumbre. En este contexto, se presenta esta propuesta de preparación en cuatro pasos: revelar, ver, darse cuenta y actuar, como un enfoque estructurado que permite a una compañía responder, anticiparse y transformar el ≪no saber≫ en una acción proactiva, para fortalecer su ciberresiliencia y capacidad para prosperar en el futuro. Basándose en un análisis de una señal débil identificada, se detalla la aplicación de la propuesta, para desarrollar una postura vigilante que permita a las empresas explorar y evaluar el contexto actual y los escenarios relevantes, para finalmente cerrar con algunas conclusiones sobre el reto de anticipar.**

***Index Terms*—Amenaza, Anticipar, No saber, Señal débil**

I. Introducción

La dinámica acelerada de cambios que experimentan las organizaciones y el mundo en general, crea necesariamente un "superávit de futuro", una necesidad de distinguir y anticipar aquello que puede aparecer y terminar afectando positiva o negativamente, directa o indirectamente los objetivos de las empresas. En este sentido, más que "predecir" el futuro, las corporaciones deben comenzar a avanzar en estrategias que le permitan contar con la información necesaria para transformar su propia realidad y entorno de manera previa, para limitar las sorpresas indeseadas y capitalizar aquellas que son relevantes para su supervivencia [1].

En este sentido las teorías base de los pensadores de sistemas permiten fundamentar aquellas acciones que las empresas deben asumir para permanecer y navegar por los entornos turbulentos que las inestabilidades globales le plantean ahora y en el futuro. Por tanto, la pregunta es ¿qué deben hacer las organizaciones para "revelar", "ver", "darse cuenta" y "actuar" frente a los cambios relevantes que la pueden impactar ahora y en el futuro? La respuesta está fuera de la zona cómoda de los estándares, buenas prácticas, la ilusión del control y la conceptualización del mundo como un sistema lineal [2].

Esto implica, llevar a la organización a mantenerse incómoda con lo que sabe, abierta a lo que no conoce, dispuesta a abrazar los inciertos, particularmente "aquellos que no sabe, que no sabe" y a tomar decisiones con información limitada.

Anticipar por lo tanto, requiere un esfuerzo adicional, una inversión en energía e incomodidad para generar situaciones no convencionales de forma previa, para identificar nuevos estados que pueden surgir por interacciones entre los componentes de la organización y su ecosistema, de tal forma, que se genere nueva información que le permita a la empresa, revelar aspectos antes inexplorados, los cuales la habiliten para ver situaciones atípicas frente a lo que sabe y de esta forma, darse cuenta (despertar) que algo ha cambiado, que es relevante o no para su objetivos y tomar las acciones, que sean pertinentes, aun sin todo el panorama claro de sus impactos [3].

Por tanto, el ejercicio de anticipar exige un proceso que desinstala a la organización de su saber previo, para generar inestabilidad e incierto en su forma de entender su entorno, para desde allí identificar "diferencias que hacen diferencia" [4], que generen sorpresas o distinciones inexistentes para identificar patrones emergentes y así, conectar nuevos puntos de una nueva realidad que se construye a partir de aquello que se surge en medio de la entropía que produce el "no saber". De esta forma, sólo es posible "mirar hacia adelante", si la compañía está dispuesta a "sorprenderse" de forma periódica y sistemática con el fin de trazar una carta de navegación que la proyecte como un "visitante del futuro" [5].

Con estas ideas en mente, este artículo inicia con una revisión de investigaciones previas sobre el anticipar basadas en modelos de prospectiva, luego la lectura del pensamiento estratégico en ciberseguridad como fundamento de la propuesta, para luego presentar y aplicar los cuatro pasos propuestos para anticipar amenazas cibernéticas, finalizando con algunas conclusiones sobre el reto de mirar hacia adelante sin tener todas las respuestas.

II. Prospectiva para anticipar

Anticipar es un ejercicio que implica no solamente contar con datos precisos, sino establecer patrones y explorar posibilidades para ir decantando aquello que puede ser relevante para una organización de cara a sus objetivos estratégicos de mediano y largo plazo.

En la actualidad, las organizaciones operan en entornos caracterizados por la volatilidad, la incertidumbre, la complejidad y la ambigüedad (VUCA, por sus siglas en inglés). Ante este panorama, emerge la prospectiva como una capacidad dinámica clave que permite a las empresas detectar, interpretar y responder a cambios discontinuos en su entorno. La prospectiva no busca predecir el futuro de manera determinista, sino imaginar múltiples escenarios plausibles para informar la toma de decisiones presente y construir una ventaja competitiva resiliente [20].

La prospectiva como un proceso sistemático y participativo para recopilar inteligencia confiable y construir visiones de mediano y largo plazo se estructura en cuatro fases esenciales: percepción (detección de señales), prospección (creación de escenarios), sondeo (pruebas de estrategias) y transformación (reconfiguración organizacional) [19].

En este contexto, la mayoría de las organizaciones se encuentran en un estado de vulnerabilidad debido a su incapacidad para analizar señales débiles. Estas señales son indicadores tempranos, a menudo ambiguos y fragmentados, de cambios profundos que aún no son evidentes en el escenario actual, lo que conlleva a la configuración de puntos ciegos, que pone a las empresas en desventaja frente a eventos inesperados [21].

Es por esto, que el pensamiento sistémico, en respuesta a la perspectiva tradicional lineal y predecible, permite observar un conjunto de elementos interrelacionados que producen patrones de comportamiento emergentes a lo largo del tiempo para visualizar relaciones entre aspectos aparentemente inconexos, y de esta forma, habilitar a la organización para "ver antes" que sus competidores y prepararse para los puntos de inflexión estratégica [22].

El reto frente a la gestión del riesgo cibernético es incorporar las prácticas de las fases de la prospectiva de forma articulada y metodológica para generar la anticipación requerida frente a las amenazas cibernéticas que por definición son sistémicas, emergentes y disruptivas. Estas fases no están presentes en los marcos de trabajo vigentes en ciberserguridad como el marco de ciberseguridad del NIST 2.0, pues particularmente este marco está diseñado para preparar a las organizaciones frente a riesgos cibernéticos conocidos.

En este sentido se hace necesario desarrollar pensamiento estratégico en ciberseguridad, que le permita a la organización pensar fuera los marcos tradicionales, para retar el conocimiento adquirido a la fecha y desde allí considerar nuevas formas de anticipar las amenazas cibernéticas emergentes.

III. Pensamiento estratégico en ciberseguridad

Darle forma al futuro y retar lo que se tiene en el presente es una competencia básica de los pensadores estratégicos. Este ejercicio requiere un proceso cognitivo complejo, pues pasa inicialmente por reconocer el saber previo, luego desinstalarse de su marcos de referencia conceptuales (desconectar lo que sabe), seguidamente reconocer los nuevos puntos y retos del contexto, nuevamente conectar las novedades y repensar su referentes conceptuales, para generar una visión novedosa e enriquecida que responda a las asimetrías que el entorno le genera [6].

Aprender a mirar hacia adelante implica reconocer en el entorno las señales débiles que introducen ruido y cambian las conversaciones sobre el riesgo cibernético en las empresas, no para distraer la atención de lo importante, sino para afinar e iluminar el camino de los objetivos estratégicos empresariales, que se verán afectados y muchas veces modificados por los cambios propios de un mundo NAVI (No lineal, Acelerado, Volátil e Interconectado) [7], los cuales de acuerdo con la postura empresarial, podrán ser interpretados como oportunidades o como amenazas [8].

En este sentido, el riesgo cibernético requiere un pensamiento estratégico situado tanto en ejecutivos como en directivos, con el fin de lidiar con las contradicciones, las paradojas y las ambigüedades de esta temática, que de manera permanente los ubiquen fuera de su zona cómoda, desinstalados de los referentes conceptuales, para deconstruir lo que saben a la fecha, y que los lleve a pensar más allá del contenido ("el qué") o el proceso ("el cómo"), y preguntar de forma abierta y sin limitaciones sobre la premisa subyacente ("el por qué") que cuestione a los líderes más sobre las posibilidades que sobre las probabilidades [6].

De acuerdo con [9] desarrollar el pensamiento estratégico implica dominar seis disciplinas básicas: anticipar, desafiar, interpretar, decidir, alinear y aprender, las cuales enmarcan el proceso de reflexión y análisis de un estratega, que no sólo conduce a la organización hacia su permanencia, sino que la mantiene vigilante y sensible a los cambios del entorno para anticipar y adaptarse con el fin de repensar su propia lectura estratégica, más allá de las utilidades y posicionamiento en el mercado. A continuación, en la tabla I se hace un resumen de las seis disciplinas y sus definiciones.

Pensar de manera estratégica en ciberseguridad implica abrazar la incertidumbre y el error como fundamento para defender y anticipar. Esto es, reconocer las amenazas emergentes como fuentes esenciales de aprendizaje, agilidad y crecimiento, que habilitan la adaptación, la flexibilidad y el amortiguamiento organizacional frente a un evento cibernético adverso, que no es otra cosa que la resiliencia cibernética [10].

Lo anterior demanda que tanto los ejecutivos como los directivos generen un cambio de mentalidad hacia la inevitabilidad de la falla, reconozcan el error como capacidad de aprendizaje y establezcan mecanismos concretos que operacionalicen esta nueva perspectiva en la práctica de la defensa y la anticipación.

A continuación se hace un resumen de la aplicación de estas seis disciplinas en la ciberseguridad en la tabla II.

Basado en lo anterior, y con el fin de responder la pregunta planteada inicialmente, y dando respuesta a las limitaciones de los marcos de trabajo en ciberseguridad como el del NIST, aplicado al reto de la gestión del riesgo cibernético en las organizaciones modernas, se introducen y detallan los cuatro pasos para adelantar la anticipación de las amenazas cibernéticas, como una forma estructurada y metodológica para orientar un ejercicio prospectivo frente al reto de anticipar amenazas emergentes en ciberseguridad empresarial.

IV. Cuatro pasos para anticipar amenazas cibernéticas

IV-A. Paso 1: "Revelar"

Revelar, es correr un velo, "abrir los ojos" frente a una realidad que posiblemente no se ha visto hasta el momento. Para ello, es necesario contar con sensores especializados y calibrados que recojan información relevante para la organización en términos de las amenazas cibernéticas de su interés. En términos prácticos, esto se traduce en desarrollar de forma sistemática, sistémica y periódica el ciclo de inteligencia de amenazas, que le permita a la empresa identificar patrones emergentes de la realidad cambiante de las técnicas, tácticas, procedimientos y herramientas (TTPH) que utilizan los adversarios, de tal forma que pueda indicar aquellos atacantes nuevos y las variaciones de los conocidos.

Lo anterior implica organizar múltiples datos para darle forma nuevos patrones, que se traduzcan en información. Esto es, "in-forma"r, "darle nueva forma" al escenario donde opera la organización, cambiando la manera como se percibe el entorno, para luego tomar las acciones del caso alrededor de las distinciones que se pueden generar con la información

Tabla I
DISCIPLINAS DE PENSAMIENTO ESTRATÉGICO.

Disciplina	Detalle	Foco
Anticipar	Ver amenazas futuras antes que el mercado.	Identificar señales débiles.
Desafiar	Desafiar activamente el statu quo actual.	Retar resultados vigentes.
Interpretar	Sintetizar datos complejos y ambigüos.	Conectar puntos inconexos.
Decidir	Tomar decisiones audaces y flexibles.	Balancear intuición y análisis.
Alinear	Movilizar el compromiso bajo una visión retadora.	Comunicar una visión.
Aprender	Extraer lecciones y experimentar continuamente.	Hacer nuevas distinciones.

Nota: Basado en [9]

Tabla II
PENSAMIENTO ESTRATÉGICO EN CIBERSEGURIDAD.

Disciplina	Pensamiento estratégico en ciberseguridad
Anticipar	Ver antes y explorar ampliamente los riesgos: identificar y priorizar las rutas de ataque más probables, anticip la respuesta de los adversarios y los fallos en cascada del sistema.
Desafiar	Desafiar los supuestos del modelo de seguridad y control actual: romper el statu quo de la seguridad tradici que tiende a culpar al "error humano" o a las políticas de seguridad.
Interpretar	Interpretar un amplio conjunto de datos y puntos de vista: identificar puntos ciegos en el modelo de seguridad y cor en "patrones no convencionales" que revelan los datos para refinar y ajustar los modelos mentales vigente
Decidir	Tomar decisiones estratégicas: invertir recursos en soluciones basadas en seguridad y resiliencia por diseño basándose de los experimentos y fallas identificadas.
Alinear	Promover una visión retadora y anticipativa: alinear equipos de seguridad y los diferentes grupos de interés c la promesa de valor, el apetito de riesgo cibernético y las inversiones en soluciones resilientes que rompan la cultura de silos en la gestión del riesgo cibernético.
Aprender	Asegurar el ciclo continuo de aprendizaje: preservar la capacidad de sorprenderse frente al comportamiento inesperad (aprender), descubrir y retar los alcances y efectividad de los estándares y buenas prácticas vigentes frente al entorno NA e incorporar nuevas experiencias y riesgos emergentes que cambian el statu quo de la ciberseguridad empresarial (r

Nota: Elaboración propia basado en [9]

disponible. Si la información recolectada no hace diferencia, no exhibe algo distinto que rete lo que la organización sabe, se habla de algo que es conocido, y por lo tanto, en el sentido estricto no sería "información" [3].

En este contexto, contar con mayor "información" es encontrar lugares novedosos que advierten a la organización de posibles sorpresas y por lo tanto, la movilizan para entender y actuar de forma previa. De esta forma, al contar con un sistema de correlación de la "información" de la inteligencia de amenazas, es posible crear una base de datos de amenazas emergentes o nuevas mutaciones disponibles, para generar conocimiento de forma anticipada, que lleva a la organización a crear nuevas estructuras y propuestas de defensa que mejoran sus capacidades cibernéticas de respuesta al entorno en constante evolución.

IV-B. Paso 2: "Ver"

El ejercicio de ver, conlleva no sólo acción biológica de visualizar lo que se tiene en el panorama, sino el reto de trazar nuevos límites en la realidad y desde allí, descubrir formas que no se percibían previamente. La manera como hacen los nuevos trazos de la realidad, tiene como base el ejercicio previo de revelar. En la medida que la organización, pueda crear nuevas forma de organizar la "información", podrá ver nuevos patrones de análisis que le permitan visualizar contextos distintos que la prepararen para situaciones inesperadas.

El ejercicio de "ver" implica diseñar y analizar escenarios, un marco de trabajo que lleva a la organización a ordenar y situar la "información" disponible en un contexto concreto para establecer aquellos futuros posibles y probables que le digan qué situaciones pueden ocurrir y los posibles impactos que se pueden presentar. Tomar información actual, por ejemplo alrededor de archivos PDF "inteligentes malignos" [11], que pueden ser utilizados para desplegar malware asistido por inteligencia artificial, pone de manifiesto eventos que no se han visto previamente, y por lo tanto, generan incomodidad en el marco de seguridad y control de la organización. Esto es, una oportunidad para generar nuevo conocimiento y actualización de las capacidades cibernéticas disponibles.

Ver es trazar nuevas líneas en el horizonte de las amenazas que se tienen "reveladas" para darles forma en la dinámica de la organización, que más allá de un ejercicio estéril de probabilidades (que conceptualmente no viable realizar por la característica sistémica del riesgo cibernético), le permita a la empresa plantear las posibilidades relevantes que la preparen para enfrentar contextos hasta ahora no documentados. Esto es, crear de forma anticipada estados no previstos en los flujos de los procesos empresariales, para establecer cómo se propagan los impactos, cómo establecer estrategias para limitarlos y sobremanera, que capacidades se requieren para contener sus efectos y así proteger la compañía, sus grupos de interés, y por tanto su reputación.

Diseñar y analizar escenarios debe ser la práctica natural de las organizaciones que quieren permanecer, a pesar de tener incidentes de seguridad exitosos. No es una receta mágica para prepararse para cualquier evento, sino un espacio de conversación y creación de conocimiento que la organización tiene para pensar distinto y recrear, al menos en un momento del tiempo, la misma apuesta del adversario cómo crear distracción y engaño.

IV-C. Paso 3: "Darse cuenta"

Este momento es el más complejo de la cadena que se viene desarrollando. Es aquel momento "eureka", el momento en que se "interrumpe la dinámica de la realidad" que se conoce. Es la manera en que se encuentra "algo que no encaja" en lo

que se sabe hasta el momento. Es no poder nombrar algo que se vislumbra desde el "no saber", que permite interrogar las creencias y conocimiento propios, más allá de "ver" el nuevo escenario que se propone, para ahora concentrarse en los impactos.

De acuerdo con [12]: "El riesgo es una estimación. Es la incertidumbre sobre lo que podría suceder, determinada por la exposición, la probabilidad y las consecuencias. El resultado es lo que ocurre cuando la estimación se hace realidad". Esto es, tenga en cuenta qué es lo que está en riesgo, quién o quiénes dependen de él, cómo se tiene acceso a ese elemento y el tamaño del radio del impacto si algo adverso ocurre.

"Darse cuenta" implica identificar los impactos o situaciones no deseadas que se generan luego de los ejercicios y simulaciones de los escenarios, con el fin de encontrar puntos ciegos y zonas indefinidas de acción que llevan a aumentar la entropía propia de un incidente a niveles que pueden desbordar la capacidad de la organización, superar la tolerancia de riesgo previamente definida y violar la declaración de apetito de riesgo cibernético empresarial.

Cuando la organización no se "da cuenta" de lo que puede ocurrir, de lo que puede pasar y se queda sólo con el escenario, no lograr concretar su proceso para anticipar, pues deja sin conclusión la exploración y validación del nuevo escenario que ha trazado para ver la realidad de forma distinta. Pasar del mapa, a la validación en el terreno práctico de la organización y sus procesos es lo que le permita a la empresa, crear las capacidades necesarias para transformar lo que sabe a la fecha y enriquecer la forma de entender la realidad.

Por lo general, cuando el escenario se simula, las preguntas sobre la probabilidad de que esto ocurra, pasan a un segundo plano, pues los eventos y acciones que se desarrollan terminan cambiando la forma como la compañía entiende lo que ocurre en su dinámica empresarial. Las simulaciones y ejercicios no sólo revelan "información", o validan los nuevos trazados en el territorio de amenazas identificado, sino que ubica y confirma en la realidad de la empresa, aquello que hace la diferencia en su hacer y por tanto afecta la forma como hace sus negocios.

IV-D. Paso 4: "Actuar"

Luego de "revelar", "ver" y "darse cuenta", el reto es tener la disposición y determinación para actuar en consecuencia. Muchas organizaciones a pesar de tener la "información" para avanzar en la actualización de sus capacidades, no lo hacen creando lo que la literatura denomina "sorpresas predecibles" [13], esos eventos que se materializan de forma concreta, a pesar de contar con los elementos claves para contenerlos o enfrentarlos. Una organización preparada para anticipar, no sólo requiere los tres pasos previos, exige una firme convicción para invertir tiempo, dinero y personas calificadas, para movilizar sus esfuerzos y cambiarse a sí misma, y a su entorno.

El cambio implica movilizarse desde el punto donde se encuentra actualmente una organización. Es crear nuevas redes y límites extendidos dentro de la postura de ciberseguridad de la empresa que se ajusten con las "diferencias que hacen la diferencia" en sus capacidades cibernética actuales, para crear "memorias del futuro" [14] en la dinámica del presente de la estrategia de ciberseguridad empresarial.

Actuar en el ejercicio de anticipar es coordinar, colaborar, cooperar, comunicar y confiar con los diferentes participantes del ecosistema digital de negocios, para construir nuevas distinciones de una defensa cibernética conjunta, que habilite compartir "información" de forma fluida y estratégica para mantener una red de sensores debidamente calibrados frente al entorno, con el fin de contar con una inteligencia de amenazas enriquecida y eficiente, que haga la diferencia para todos los participantes del ecosistema.

La acción como elemento activo del anticipar termina habilitando a la organización para "aprender haciendo" esto es: [15].

- Conocer para actuar: El conocimiento solo tiene importancia si sirve para la resolución de problemas reales.
- Interactuar en contexto: La organización aprende en interacción con su ambiente, a partir de su capacidad de adaptación funcional, mediante ejercicios y simulaciones.
- Aprender de forma situada: Se considera que el aprendizaje es el resultado de una actividad situada en un contexto que le otorga significado a quien la ejecuta y es pertinente para el logro de sus objetivos.
- Interrogar el saber previo: Implica un esfuerzo por retar lo conocido, con una dimensión prospectiva que supera el presente inmediato.
- Invertir la relación "teoría – práctica": Pasar de la "práctica a la teoría" ejecutando las simulaciones para ganar confianza y contar con "memoria muscular" para responder.
- Reconstruir la experiencia: Fomentar actitudes y valores para pensar distinto y "fallar bien" para incrementar la capacidad de actuar con independencia cognoscitiva.

Parafraseando a Dewey (citado por [13]) en términos del anticipar: "anticipar no tiene valor por sí mismo, no es un agregado de suposiciones o ideas simples, sino que es un ejercicio que adquiere valor (diferencial) para las organizaciones, cuando se conforman y definen acciones concretas por parte de los individuos". Esto es, darle sentido a la "experiencia del no saber" y sacarle provecho para operar en medio de las nuevas amenazas que enfrentará la organización en el futuro.

En síntesis, las organizaciones deben desinstalarse de su saber previo y estar dispuestas a sorprenderse de forma periódica y sistemática para actualizar su carta de navegación en el entorno digital de forma proactiva. Esto implica una inversión deliberada en vigilancia para "revelar" señales débiles, "ver" escenarios no documentados, "darse cuenta" de los impactos no previstos, y "actuar" con convicción para transformar su postura defensiva.

V. Caso de uso de los cuatro pasos para anticipar amenazas cibernéticas

V-A. Contexto del ejercicio

Considerando los elementos de la planeación estratégica aplicados para la ciberseguridad y la aplicación de los elementos de la prospectiva mencionados previamente, se enmarca el ejercicio de aplicación de los cuatro pasos para anticipar amenazas cibernéticas. Este caso experimental que se desarrolla se sitúa en el proceso de identificación y revelación de tendencias

en ciberseguridad empresarial para el 2026 elaborado en el contexto académico para la región latinoamericana.

V-B. Paso 1: "Revelar"- El reto de la señal débil

Luego de desarrollar una revisión de al menos 50 documentos relacionados con tendencias, reportes y pronósticos de ciberseguridad para el año 2026, con el apoyo de una inteligencia artificial tipo RAG (*Retrieval Augmented Generation*), se identificaron algunas señales débiles, que son pequeños fragmentos de información o datos que, en un principio, pueden parecer insignificantes o no relacionados, pero que, una vez interpretados, pueden revelar la posibilidad de un cambio futuro significativo, una ruptura o una oportunidad/amenaza con consecuencias considerables.

El proceso de identificación y clasificación de señales débiles en ciberseguridad es un componente esencial de la inteligencia estratégica anticipativa, buscando ver rupturas en el entorno con suficiente antelación.

Para 2026, las fuentes indican que la Inteligencia Artificial (IA) no solo amplificará las amenazas existentes, sino que también permitirá la emergencia del "crimen sin rostro humano" y "plataformas criminales autónomas (Crime-as-a-Service)". Esto se manifiesta en la automatización de ataques sofisticados y la disolución de la identidad operativa de los ciberdelincuentes.

V-C. Paso 2: "Ver" - El reto de valorar la señal débil

Basado en el resultado anterior, se plantea a continuación un escenario donde se le da forma a una señal débil asociada con el "crimen como servicio" ahora potenciado con IA. La narrativa del escenario se describe a continuación.

Un analista de ciberseguridad monitorea un foro clandestino de la dark web especializado en la venta de exploits. Encuentra un "post" anónimo que, aparentemente, solo anuncia la fase beta de una "plataforma de IA-as-a-Service" llamada "GhostPhish-AI". El anuncio es escueto y se centra en su capacidad para generar campañas de phishing sofisticadas, incluyendo mensajes de voz deepfake personalizados que imitan la voz de un CEO o directivo para la verificación de transacciones financieras. La plataforma afirma que el usuario solo necesita introducir el objetivo y la IA se encarga del resto, con "mínima intervención humana" y "trazabilidad nula". El post también incluye un testimonio (presumiblemente falso) de un "beta tester"" que describe el éxito de un ataque de ingeniería social usando una voz deepfake para inducir una transferencia bancaria en una organización financiera pequeña, logrando un "robo de fondos indetectable".

Seguidamente se inicia la evaluación de la señal débil identificada basado en los siguientes criterios detallados en la tabla III.

La valoración que se adelanta por cada señal débil identificada se detalla a continuación.

- Las señales con una puntuación total alta (sumando todos los criterios – 13-15) requieren atención inmediata de la alta dirección y la implementación de estrategias de anticipación. Actuar en menos de 3 meses.
- Las señales con puntuación media (9-11) deben ser monitoreadas de cerca y pueden requerir análisis adicionales para determinar su evolución. Actuar entre 3 a 6 meses.
- Las señales con baja (5-8) puntuación pueden ser "ruido" o no ser relevantes en el contexto actual, pero no deben descartarse sin un mínimo escrutinio inicial. Actuar entre 7 a 12 meses.

La valoración de la señal débil denominada "GhostPhish-AI" se presenta a continuación en la tabla 4.

La puntuación de 13 para la señal débil "GhostPhish-AI" implica que la organización debe tomar esta señal con seriedad y activar una respuesta estratégica en menos de tres meses.

V-D. Paso 3: "Darse cuenta"- El reto de simular la señal débil

Con el resultado de la evaluación de la señal débil, la organización seguidamente debe simular el escenario situado en la dinámica de su negocio con el fin de transformar la incertidumbre en conocimiento estratégico para la empresa.

Estos ejercicios se materializan en "tabletop" o ejercicios de mesa que retan a la organización sobre lo que han aprendido hasta el momento y la llevan fuera de la zona cómoda, entendiendo que, si la señal débil se hace fuerte y se materializa, la organización tendrá que ajustar diferentes procesos y procedimientos, así como ajustar sus "libros de jugadas" para enfrentar una realidad totalmente distinta a lo que se conocía hasta el momento.

La práctica periódica de este tipo de ejercicios demuestra compromiso y madurez de la organización para mantenerse en el ciclo de aprender, desaprender y reaprender, que le permite estar sintonizada con los retos del entorno, calibrando y ajustando sus controles, y sobremanera, afinando su estrategia de defensa para anticipar a sus adversarios en su propio terreno para ganar tiempo, antes de que pueda tener éxito.

V-E. Paso 4: "Actuar"- El reto de atender la señal débil

Considerando la información y el marco desarrollado hasta el momento, la organización deber tomar acciones concretas que den cuenta de su compromiso de anticipación y que le permita capitalizar los inciertos que le plantea la señal débil. Esto implica, mejorar y ajustar cuatro capacidades cibernéticas claves de una organización, a nivel de infraestructura, aplicaciones, servicios y comportamientos humanos, que se enuncian a continuación [23].

- Defensa
- Radar
- Crisis
- Monitorización

Para el caso analizado las posibles acciones se detallan a continuación:

1. Monitoreo continuo y avanzado: establecer un equipo dedicado para rastrear foros de la dark web, plataformas de CaaS y repositorios de IA para detectar desarrollos similares a "GhostPhish-AI".
2. Investigación y desarrollo de controles: invertir en soluciones de detección de deepfakes de voz e imagen, y sistemas avanzados de detección de phishing asistidos por IA.
3. Capacitación y concienciación del personal: Implementar entrenamientos de concienciación sobre ciberseguridad que enfaticen la identificación de deepfakes y ataques de ingeniería social basados en IA, priorizando la "calidad sobre la cantidad".

Tabla III
CRITERIOS DE EVALUACIÓN DE UNA SEÑAL DÉBIL

Criterio	Descripción	Escala de valoración (1 a 3)
Anticipación	¿Cuán temprano es este indicio? (Capacidad de alerta temprana)	1 (Baja): La tendencia ya está bien establecida. 2 (Media): Se está volviendo visible; hay tiempo para reaccionar. 3 (Alta): Es un indicio muy incipiente, fragmentario y novedoso. .
Relevancia	¿Afecta a la misión principal, el modelo de negocio o la estrategia corporativa?.	1 (Baja): Impacta solo en funciones auxiliares. 2 (Media): Impacta en un área clave, pero el negocio es adaptable. 3 (Alta): Impacta la misión crítica o la supervivencia a largo plazo.
Ambigüedad	¿Cuán claro es el significado del evento o tendencia?	1 (Baja): La causa y el potencial impacto son claros. 2 (Media): Hay varias interpretaciones o posibles habilitadores. 3 (Alta): Es muy fragmentario, difícil de interpretar sin más datos.
Credibilidad	¿Existe ya evidencia técnica, PoC (Prueba de Concepto) o base teórica sólida?	1 (Baja): Basado en especulación o teoría no probada. 2 (Media): Existen PoC académicos o experimentos limitados. 3 (Alta): Basado en legislación, tendencias tecnológicas probadas o incidentes incipientes.
Potencial Impacto	¿Cuál sería la gravedad si la señal se materializa a gran escala?	1 (Baja): Pérdidas financieras menores o reputación local. 2 (Media): Interrupción significativa de operaciones o multas regulatorias severas. 3 (Alta): Colapso sistémico o pérdida de Propiedad Intelectual (PI) irremplazable.

Nota: Elaboración propia

Tabla IV
EJEMPLO DE EVALUACIÓN DE SEÑAL DÉBIL.

Criterio	Valoración	Justificación
Anticipación	3	La plataforma está en “fase beta” y se anuncia en un foro de nicho, indicando un desarrollo muy temprano de una capacidad potencialmente disruptiva de “crimen autónomo” y “disolución de la identidad operativa”.
Relevancia	3	Un sistema que automatiza el phishing sofisticado con voces deepfake y que, además, se enfoca en “robo de fondos indetectable” es directamente relevante para la organización.
Ambigüedad	3	El anuncio es de un “post anónimo” en la dark web, y la plataforma “GhostPhish-AI” es desconocida. La descripción de “mínima intervención humana” y “trazabilidad nula” es una pretensión que requiere una verificación técnica profunda.
Credibilidad	1	La fuente es un “post anónimo” en la dark web, lo que reduce significativamente su credibilidad inicial. Aunque las capacidades descritas (IA generativa, deepfakes) son técnicamente posibles, se requiere una “triangulación de datos” y consulta a expertos para validar la información.
Potencial Impacto	3	Si una plataforma de este tipo se consolida y democratiza el phishing con deepfakes y la “trazabilidad nula”, el impacto sería ”sistémico”. Podría provocar “pérdidas significativas” para las organizaciones y erosionar la “confianza pública”.

Nota: Elaboración propia

4. Actualización de protocolos de incidentes: revisar los planes de respuesta a incidentes para incluir escenarios de ataques sofisticados con IA, especialmente aquellos que involucran suplantación de identidad de ejecutivos y la disolución de la identidad operativa.
5. Colaboración y compartir inteligencia: Compartir esta información con pares del sector, autoridades reguladoras y agencias nacionales de ciberseguridad.
6. Gobierno de IA: es clave desarrollar un “gobierno de IA” dentro de la organización, que integre los riesgos de

IA en los modelos de amenazas y asegure la supervisión de las operaciones de IA.

7. Preparación para la "deshumanización de la ejecución del delito": reconocer que "el crimen ya no se trata solo de digitalizar, sino de deshumanizar su ejecución" y prepararse para escenarios donde "el perpetrador puede ser una interfaz, la amenaza un algoritmo, y el trofeo una línea de código".

VI. Conclusiones

Responder la pregunta: ¿qué deben hacer las organizaciones para "revelar", "ver", "darse cuenta" y "actuar" frente a los cambios relevantes que la pueden impactar ahora y en el futuro? en el contexto del riesgo cibernético no es un ejercicio pasivo, es una acción comprometida con el futuro de la empresa y de responsabilidad con sus grupos de interés. Es una acción proactiva que da sentido y afina la declaración de apetito de riesgo cibernético de la empresa de forma situada y específica, para atender las inestabilidades que la organización va a enfrentar más adelante.

Anticipar frente al riesgo cibernético es retirar la venda de "qué probabilidad se tiene que esto puede pasar", "esto no ha pasado antes" y "esto no es viable que pase" para avanzar en el debido cuidado empresarial que no sólo prepare a la organización para eventos adversos en sus operaciones, sino que la habilite para "aprender haciendo", un proceso activo, experiencial y reflexivo, profundamente arraigado en la resolución de retos auténticos y en la interacción con el entorno dinámico y asimétrico donde operan las organizaciones.

Anticipar implica desarrollar la epistemología del "no saber" [16], de pactar con el incierto para navegar en medio de las tensiones y tendencias que se generan en el ecosistema digital de negocios, no sólo desde la perspectiva de los eventos cibernéticos adversos, sino desde la perspectiva de un atacante que evoluciona y reta todo el tiempo el conocimiento de las organizaciones en el contexto digital. Esto implica que la organización, no sólo construye y adquiere conocimiento sino que es un diseñador y gestor del incierto en su entorno, con la capacidad de identificar patrones, diagnosticar situaciones, establecer prioridades y adaptar su estrategia de ciberseguridad a las necesidades específicas de su negocio y contexto.

Anticipar no es sólo conocer las posibles amenazas cibernéticas emergentes en un momento particular del tiempo, sino entender los estados de transición dentro del ecosistema digital que desencadenan efectos dominó con afectaciones e impactos no previsibles. No es un ejercicio de controlar variables o situaciones, sino modelarlas, situarlas y simularlas para entender, en la dinámica de los procesos de negocio, cómo se van a comportar y los efectos que se pueden presentar, esto es, revelar esos puntos críticos donde las consecuencias en el negocio pueden o no ser reversibles [18].

Cuando una organización anticipa frente al riesgo cibernético estructura y desarrolla nuevo conocimiento para responder a los cambios inesperados del entorno. Por tanto, "Si consideramos que el futuro ya no es proyectivo (lineal) sino prospectivo (escenarios) entonces se vuelve crítica la curiosidad, la capacidad de observar, detectar patrones, analizar tendencias, considerar múltiples perspectivas, estudiar lo qué hacen otras organizaciones e industrias (inteligencia competitiva), imaginar posibilidades" [17]. Por tanto, anticipar no es una competencia opcional, sino imperativo estratégico que prepara a las organizaciones no sólo para sobrevivir, sino para permanecer, aun cuando en la actualidad abundan los inciertos que proponen los riesgos cibernéticos y escasean las certezas que antes brindaban los estándares y las buenas prácticas. Como decía Carl Sagan: "*la ausencia de evidencia, no es evidencia de ausencia*".

Referencias

[1] Day, G. and Schoemaker, P. *See soon, act faster. How vigilant leader thrive in an era of digital turbulence*. Cambridge, MA. USA: MIT Press, 2019.

[2] Cano, J. "The Illusion of Control and the Challenge of an Adaptable Digital Enterprise". *ISACA Journal*. 3. https://www.isaca.org/resources/isaca-journal/issues/2023/volume-3/the-illusion-of-control-and-the-challenge-of-an-adaptable-digital-enterprise, 2023

[3] Rasmussen, L. *Seeing. A field guide to the patterns and processes of nature, culture and consciousness*. Makawao, Hawaii. USA: The Maui Institute, 2024

[4] Bateson, G. *Mind and Nature: A Necessary Unity*. E.P. Dutton: New York, NY, USA, 1979.

[5] Bruner, J. Prologue to the English edition. En Rieber, R. W. and Carton, A. S. (editors). *The collected works of L. S. Vygotsky. Vol. 1. Problems of general psychology*. 1-16. New York: Plenum Press, 1987

[6] Sloan, J. *Learning to think strategically*. Fourth Edition. Oxxon, OX. UK: Routledge, 2024

[7] McCowan, S., Krumbmüller, F. and Jaggi, G. "How can reimagining risk prepare you for an unpredictable world?" *EY Insights*. https://www.ey.com/en_/insights/consulting/how-can-reimagining-risk-prepare-you-for-an-unpredictable-world, 2025

[8] James, E. H., and Wooten, L. P. *The prepared leader: Emerge from any crisis more resilient than before*. Wharton School Press, 2022

[9] Krupp, S. and Schoemaker, P. J. H. *Winning the long game: How strategic leaders shape the future*. PublicAffairs, 2014

[10] Shortridge, K. *Security chaos engineering: Developing resilience and safety at speed and scale*. O'Reilly Media, 2023

[11] Elattar, M., Younes, A., Gad, I., and Elkabani, I. "Explainable AI model for PDFMal detection based on gradient boosting model". *Neural Computing and Applications*, 36(34), 21607–21622. https://doi.org/10.1007/s00521-024-10314-y, 2024

[12] Ferguson, R. "Defensive driving for cybersecurity". *Linkedin*. https://www.linkedin.com/pulse/defensive-driving-cybersecurity-rik-ferguson-vpizf, 2025

[13] Bazerman, M. and M. Watkins. *Predictable Surprises: The Disasters You Should Have Seen Coming and How to Prevent Them*. Boston, MA. USA: Harvard Business School Press, 2004

[14] De Geus, A. *The living company: Habits for survival in a turbulent business environment*. Harvard Business School Press, 2002

[15] Ruiz, G. "La teoría de la experiencia de John Dewey: significación histórica y vigencia en el debate teórico contemporáneo". *Foro de Educación*, 11(15), pp. 103-124. http://dx.doi.org/10.14516/fde.2013.011.015.005, 2013

[16] Correa Gallardo, J. I. "Hacia una epistemología del desconocimiento". *Revista Colombiana De Filosofía De La Ciencia*, 17(35), 119–144. https://doi.org/10.18270/rcfc.v17i35.2325, 2017

[17] Martínez, J. "Aprender del futuro". *Newsletter*. https://www.javiermartinezaldanondo.com/n179-aprender-del-futuro, 2021

[18] Geer, D. E., Jr. "Indeterminism". *IEEE Security and Privacy*. Doi: 10.1109/MSEC.2025.3612537, 2025

[19] J. O. Schwarz, *Strategic Foresight: An Introductory Guide to Practice*. Abingdon, Oxon: Routledge, 2023.

[20] M. Conway, *Foresight Infused Strategy Development: A How-To Guide for Using Foresight in Practice*. Melbourne, Australia: Thinking Futures, 2015.

[21] G. S. Day and P. J. H., *See Sooner, Act Faster: How Vigilant Leaders Thrive in an Era of Digital Turbulence*. Cambridge, MA: MIT Press, 2019.

[22] D. Loveridge, *Foresight: The Art and Science of Anticipating the Future*. New York, NY: Routledge, 2008.

[23] J. Cano, *Ciberseguridad empresarial. Reflexiones y retos para los ejecutivos del siglo XXI*. Bogotá, Colombia: Ediciones de la U.

EIP 712-based Off-Chain Oracles: Benefits and Caveats

Daniel Morales*
Applied Cryptography Team
Input Output Group, R&D
Madrid, Spain
daniel.morales@iohk.io

Isaac Agudo
NICS Lab
Universidad de Málaga
Málaga, Spain
isaac@uma.es

***Abstract*—Blockchain oracles solve the problem of interacting with off-chain data from smart contracts, but introduce several security problems and increase the cost of data consumption. There exist different types of oracles, but they mostly try to enhance security and accuracy by incorporating most of the process on-chain, e.g., by performing requests and fulfillments of data entirely through smart contracts. In order to minimize the cost associated to requests, we propose an alternative design where users can request data to the oracles off-chain, which is fulfilled in a verifiable way by using EIP 712 signed structures, and where verification is done at the time of uploading the data provided. We also discuss the security implications of this approach and propose mitigation techniques, together with a concrete implementation of the solution to Chainlink's VRF oracle, where we prove that our solution achieves a $\times 3.17$ speed up in the execution cost.**

***Index Terms*—blockchain, oracle, eip712, efficiency**

I. Introduction

One of the main problems for blockchain usability, and most exacerbated when using smart contracts, is that they can only interact with data that exist on-chain, i.e., that have been generated through the execution of an on-chain process or that have been provided by users through transactions. Since there are many situations where data exist outside of the blockchain world and held by third parties different than the users interacting with the dApp, blockchain oracles [1] have been proposed as a solution. Roughly speaking, there are intermediate parties that connect the on-chain and the off-chain world, either by performing inputs of off-chain data to the on-chain state or by gathering on-chain data and exposing them to off-chain applications in an easy-to-use format. Oracles have been considered as interesting solutions, but also as problematic ones [1, Section 3.2], mainly because they limit the amount of decentralization. A malicious or biased oracle becomes a single point of failure that can manipulate the data added to the blockchain; therefore new trust assumptions are incorporated to the blockchain ecosystem and must be handled accordingly.

Due to their negative impact on decentralization, researchers have proposed decentralized oracles that add an off-chain decentralized layer (i.e., multiple oracle servers must collaborate between them) prior to incorporating a data inclusion in the on-chain state. This avoids the need to deposit all trust in a single server that provides all data. An example of such a decentralized oracle is Chainlink's solution [2]. In fact, the general approach of two layers of decentralization (one on-chain and one off-chain) has become more popular in recent years, not only for oracles but also for other applications such as decentralized committees for key management[1][2].

There exist different types of oracle, which will be covered in more detail in Section II-A. Some of them try to facilitate the incorporation of more off-chain data to the on-chain state, arguing that they achieve simpler web3 applications and more accurate on-chain data. This happens, for example, in DeFi applications [3], where periodic reports of prices are uploaded to smart contracts. However, while these oracles truly benefit accuracy, they also introduce a major caveat: an increase in the overall on-chain storage. Notice that the more high the on-chain volume of data is, the less nodes will be available to keep a full copy of the blockchain (as archive nodes do in Ethereum), and therefore there will be a increment in centralization and decrease in data availability, which are not desirable.

Since blockchain oracles are somewhat costly to use, we introduce in this work a exploratory solution to reduce the impact of on-chain data produced by them, and also the cost of the interactions when requesting off-chain data. We target an off-chain oracle solution, i.e., one that can provide data directly to the application through an off-chain interface, but in a way that is easily verifiable on-chain when the application actually feeds that data to the smart contract. To facilitate the verification of potentially structured data, we make use of the EIP 712. A remark is that this solution for the interface between the off-chain and the on-chain worlds is agnostic to the actual level of decentralization in the oracle network. We now present the structure of the rest of the paper. In Section II we introduce specific aspects about blockchain oracles and the EIP 712. Next, Section III briefly presents the design for EIP 712-based off-chain oracles, and Section IV performs a concrete comparison between Chainlink's Verifiable Random

* Work performed while first author was at Universidad de Málaga.

[1]https://www.coinbase.com/es-es/learn/wallet/what-is-a-multi-party-computation-mpc-wallet

[2]https://zengo.com/mpc-wallet/

Function (VRF) oracle and our off-chain variation, making emphasis in key considerations for security. Finally, Section V provides a brief discussion on potential paths to follow for solving the remaining security issues, while Section VI summarizes and concludes the paper.

II. Preliminaries

A. Blockchain oracles

Blockchain oracles can be classified according to different aspects, where the main ones are covered in this section.

The first classification is based on the domains they connect. For this, there are mainly two types of oracles: (1) those that connect the on-chain world with the off-chain world (e.g., to feed real-world measures or to generate random numbers), and (2) those that connect different blockchains between them, achieving cross-chain interoperability (these are normally called *bridges*).

Another aspect for classification is the mechanism the oracles use to gather data and make it secure. According to [4], there are two main types of oracles: (1) *voting-based oracles*, where enough votes must be obtained for a piece of data input in order to be considered valid (and typically rely on incentive mechanisms to ensure correct votes), and (2) *reputation-based oracles*, where there exist a mechanism to track the actions of the oracles and classify them according to the trust that can be deposited on them (and also include authenticity-proof mechanisms). Authors of [4] also remark that there is still a need in improving oracles performance and reducing their operating cost.

On the other hand, regarding on how data is provisioned and according to [5], oracles can be classified based on two independent dimensions: (1) the *flow direction* of the data with respect to the blockchain and (2) based on *who triggers* the data provisioning. In a first classification dimension, oracles can be *inbound*, if data is fed from the outside to the blockchain state, or *outbound*, if on-chain data is provided to an off-chain consumer through the oracle. W.r.t. a second classification dimension, in a *pull case*, it is the data receiver the one that delivers a request in order to obtain it. With respect to the *push case*, it is the data provider the one that pushes the data to the receiver each time it is required, e.g., whenever data is updated or based on a certain periodicity. Note that both dimensions can be combined, leading to four scenarios, despite push and pull patterns are typically used in the context of inbound oracles. Even when [5] reports that there is no significant cost or latency in a single invocation of any of the oracle patterns, they also state that latency can be dominated by transaction inclusion time. This clearly makes a difference between inbound pull and push oracles, where the first ones usually require two on-chain transactions to be executed: the first for the request triggered by the dApp and the second for the on-chain data fulfillment action. Therefore, push oracles minimize latency by requiring a single transaction to be included on-chain.

Finally, it is also interesting to emphasize off-chain decentralization, e.g., as it happens with Chainlink's Off-Chain Reporting (OCR) mechanism [3]. This allows the oracle nodes in an oracle network to communicate between them and aggregate data in a secure way before uploading them to the blockchain, therefore saving resources. We remark that such a solution works at the oracle level, but not at the interaction phase of the oracle with the on-chain world, where the latter is where our main contribution is focused.

B. EIP 712

The EIP 712 [6] proposes a standard to hash and sign structured data on Ethereum so it is human-readable and can be easily integrated within smart contracts, in a need to gain additional structure beyond pure bytestrings. EIP 712 is based on three main concepts (refer to Figure 3 for an example in our VRF use case), but before introducing them, we specify the function *hashStruct(s) = keccak256(typeHash ‖ encodeData(s))*, where *typeHash = keccak256(encodeType(typeOf(s)))* is a constant value for each type and *encodeData* is the concatenation of all encoded data items inside the structure *s*.

1) Domain Separator: It helps in distinguishing the application or service using the typed data. It is computed as *hashStruct(eip712Domain)*, where *eip712Domain* is also a typed structure with the following fields:

- string name
- string version
- uint256 chainId
- address verifyingContract
- bytes32 salt

2) Typed Data: It is the structured data that includes the type information. One can construct new struct types by using base types (from the EVM) and then use the new ones inside other types defined. This way, on-chain applications can be easily adapted to verify use case specific data in a standard way.

3) Structured Signing: It specifies the standard procedure to perform signatures on typed data. More specifically, a signature on typed data would be computed as *sign(keccak256("\x19\x01" ‖ DomainSeparator ‖ hashStruct(data)))*.

One of the most extended use cases of EIP 712 are meta-transactions. Basically, users can generate transaction data and sign it with their wallet private keys by using this EIP, and then send the signed meta-transactions to an off-chain bundler. The bundler can wait until a certain number of meta-transaction have been received, and then aggregate them in a smart contract with a single transaction. This allows, for example, to reduce gas costs, to pay for transaction through other non-native mechanisms or to implement layer 2 aggregators.

III. Off-chain oracle requests using EIP 712

In this paper we target a new approach to consume data from inbound oracles which is more efficient in gas consumption and on-chain storage. This design's objective is to mitigate

[3] https://blog.chain.link/off-chain-reporting-live-on-mainnet/

the impact of performing queries to oracles from dApps, and we do so by moving the queries to the off-chain world and providing a verifiable mechanism thanks to EIP 712, which allows the oracle to provide signed responses which are linked to the query and application domain. We remark that this approach does not target the off-chain mechanisms to gather off-chain data in a secure way [7], but how to improve the communication interface between the off-chain and on-chain worlds. For the sake of simplicity, we'll assume a centralized oracle, but this could be instead a relay node from some decentralized solution such as OCR or other type of Secure Multi-Party Computation protocol where a single aggregated report has been produced.

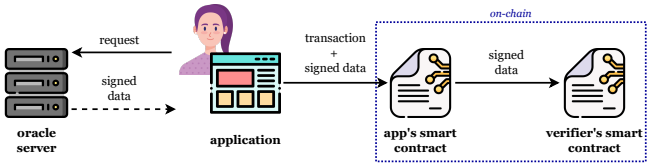


Fig. 1. EIP 712-based Off-chain Oracle.

This solution is conceptually simple, as shown in Figure 1. First, we assume that the oracle exposes an off-chain interface that can be used by the dApp to query off-chain data (or just receive it if the oracle follows a push design). Therefore, it is the user device (running the application) the one that interacts with the oracle. The data is provided within an EIP 712 structure, such that it can be uploaded on-chain by the user itself when the application requires it and the oracle's signature is easily verified by a specific smart contract (a wrong verification would imply the application not accepting that data).

Despite this solution achieves a simpler and cheaper interface to perform requests, it also presents some challenges that must be addressed. For example, the EIP 712 does not provide protection against replay attacks by itself, so this must be incorporated by the application designers, e.g., by incorporating time-stamps to the responses and linking them to the user's identity.

Therefore we must remark that this design requires, more than the others, that the oracle providing the service is deployed in a secure, accountable and transparent way, and that its identity must be known on-chain in order to perform verification of data. Since the oracle takes part in the off-chain process of time-stamping data (or any other application-specific control), reputation is key and decentralization mechanisms such as OCR can play a central role for security, otherwise the user could try to benefit from collusion with the oracle.

We now present a brief discussion on the different mechanisms to enhance security regarding on the type of application and service provided by the oracle. Note that we only consider inbound oracles because this is the scenario where on-chain costs really matter due to performing on-chain requests and fulfilling these with the requested data.

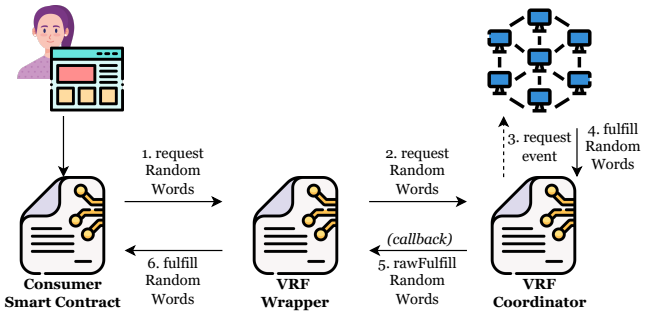


Fig. 2. Chainlink's VRF Oracle architecture.

Periodic reporting oracle. In this case the periodic reports are kept off-chain and the user only uploads to the blockchain those to be consumed on-chain. What is mandatory is to add time-stamps such that the application accepts only those reports generated within a validity period, otherwise they are rejected. If reports are public, there is no need to link data to the identity of users requesting them, but this can vary if there is a subscription mechanism involved. In addition, the accuracy of the data can be ensured by on-chain certifiers or by keeping some succinct mechanism on-chain to verify accuracy, e.g., by using Merkle Trees, where only the root hash is stored on-chain.

One-time request oracle. In this case, the user requests data a single time using a pull-based query. This can be the case, e.g., for randomness sampling. The main problem with this design is that the user may perform several off-chain queries in order to bias the result to the most favorable. Therefore, additional mechanisms for *query tracking* may be needed, where the application and/or the oracle keeps track of the data already provided to the user, thus avoiding the user to request an additional value until it is consumed in the application. This can also be solved by locking the oracle's request interface during a certain period or by advanced encryption mechanisms, as it will be covered in Section V.

IV. Implementing an Off-chain VRF Oracle

From the set of applications that exist in the blockchain oracle paradigm, one of the most interesting (and difficult to implement securely) is randomness generators. Its importance comes from the fact that randomness generated from on-chain data can always be anticipated or biased due to the public nature of the seed. In this section, we briefly introduce Chainlink's solution for a VRF oracle and show how it can be implemented with our off-chain strategy in order to save resources. We also discuss the security considerations (that are not trivial) that arise because malicious agents may try to bias randomness in order to manipulate the normal and fair execution of the application.

Chainlink's oracle uses an implementation of Goldberg's VRF [8], which roughly speaking allows to generate verifiable randomness linked to the identity of the oracle producing it by using public key cryptography. To enhance security of the VRF generation, they make use of threshold cryptography by

```
const domain = [
    { "name": "name", "type": "string" },
    { "name": "version", "type": "string" },
    { "name": "chainId", "type": "uint256" },
    { "name": "verifyingContract", "type": "address" },
    { "name": "salt", "type": "bytes32" }];
const vrf = [
    { "name": "proof", "type": "Proof" },
    { "name": "userWallet", "type": "address" },
    { "name": "timeStamp", "type": "string" }];
const proof = [
    { "name": "pk", "type": "uint256[2]" },
    { "name": "gamma", "type": "uint256[2]" },
    { "name": "c", "type": "uint256" },
    { "name": "s", "type": "uint256" },
    { "name": "seed", "type": "uint256" },
    { "name": "uWitness", "type": "address" },
    { "name": "cGammaWitness", "type": "uint256[2]" },
    { "name": "sHashWitness", "type": "uint256[2]" },
    { "name": "zInv", "type": "uint256" }];
```

Fig. 3. EIP 712 interface of the VRF Proof.

```
function getRandomness(
        bytes32 blockHash,
        ProofVerifier.VerifiableRandomness calldata vrf,
        ProofVerifier.VerifiableSignature calldata sig
    ) external returns (bool) {
        (bool success, bytes memory data) = verifier.
             staticcall(
            abi.encodeWithSignature(
                "getRandomnessFromVRF(bytes32,((uint256
                    [2],uint256[2],uint256,uint256,
                    uint256,address,uint256[2],uint256
                    [2],uint256),address),(address,
                    bytes32,bytes32,uint8))",
                blockHash,
                vrf,
                sig
            )
        );
        randomness = abi.decode(data, (uint256));
        return success;
    }
```

Fig. 4. Solidity function to verify an off-chain signed VRF in the application smart contract.

decentralizing the secret key involved. Regarding the interaction design, we provide a high-level description in Figure 2. We note that this design follows an on-chain pull-based approach, where the request of the user is registered and emits an event, which is later fulfilled by the oracle through the coordinator smart contract and by using a callback function. The problem of this oracle is not only that both the request and the fulfillment are done on-chain (and must therefore be paid), but also that the fulfillment must be paid in advance by the user in the request phase by using an estimation. Therefore, the payment of the fulfillment will normally be slightly greater than its real cost when executed by the oracle.

To reduce the cost of the requests we have implemented the VRF oracle by following the off-chain approach from Section III. Notice that the benefit is not only that the cost is reduced because of the off-chain query, but also because the cost of verifying the EIP 712 structure is computed at the exact time of data submission and therefore it avoids approximations. Figure 3 shows EIP 712 (version 4) interface to handle the typed data needed by the VRF.

We also provide a rough comparison between the execution cost of both solutions in terms of the amount of gas used. For that, we have executed in Remix IDE both a mock of the VRF oracle provided by Chainlink[4] and our EIP 712 solution (only requesting and storing one random value). A key remark is that Chainlink's mock uses a simplified version of the VRF and therefore the real cost of interacting with their oracle is expected to be slightly bigger. According to Remix, the execution cost when calling *requestRandomWords* is 145,996 gas, while the cost when calling *fulfillRandomWords* is 154,617 gas. With respect to our off-chain oracle call, the cost of submitting the VRF (from Figure 3) to the application smart contract and verifying it on-chain by calling a verifier contract takes 94,953 gas. Therefore, our solution achieves a $\times 3.17$ speed up in the overall gas. Figure 4 shows the dApp Solidity function being called by the off-chain side of the application and where the off-chain provided VRF is submitted to be verified and decoded. This is done by calling the *getRandomnessVRF* method of the verifier smart contract that verifies the EIP 712-based signature and returns the decoded randomness value such that the application can use it.

A. Security considerations

According to Chainlink, there are several security considerations from the dApp side when interacting with their VRF oracle. The following list gathers them, together with the implication for Chainlink's oracle (**CHAIN**) and our off-chain solution (**OFF**).

1) Randomness fulfillment order: It may happen that the same smart contract asks for several randomness samples in different queries and that they are fulfilled by the oracle in an order different from how they were requested. **CHAIN**: it is recommended to verify the requestID to match the request with the response, because the requestID is unique for each request. **OFF**: a requestID can also exist, but in this case, the dApp does not know in advance if the requestID is the correct one, because there is no on-chain request phase that binds the request to its unique ID.

2) Safe block confirmation: Miners/verifiers can do a *re-roll* attack by modifying the block in which the request is added. **CHAIN**: they recommend to leave a larger gap of blocks between the request and the fulfillment. **OFF**: miners cannot re-roll because the request is produced off-chain.

3) Do not allow re-requesting or cancellation of randomness: This must be ensured by the dApp to avoid attacks or failures that bias the result. **CHAIN**: the dApp keeps track of the requests that exist on-chain. **OFF**: the dApp does not know how many requests there are because they are off-chain.

4) Don't accept new application inputs after the randomness request has been made: The dApp should not accept new inputs even if the fulfillment has not been completed yet, otherwise rewriting attacks could be made to bias the result. **CHAIN**: the dApp has a mechanism to close the input phase once the VRF request has been added on-chain. **OFF**: in this

[4]https://docs.chain.link/vrf/v2-5/subscription/test-locally

case, the request is not on-chain and the VRF is not directly related to the blockchain state.

5) The method "fulfillRandomWords" must not revert: This is because the method is paid in advance and if it reverts is because of an implementation failure in the consumer's contract. **CHAIN**: if it reverts, the VRF service will not attempt to call it a second time. **OFF**: in this case, the fulfillment is done by the consumer and therefore he is in charge of deciding how reverts are handled.

V. Discussion and alternative designs

The section above shows that our solution, as it is somewhat simple, requires additional mechanisms to guarantee security regarding on the use case. For VRF fulfillments, the application smart contract does not know the order of the requests or if they have been requested more than once.

One solution would require the smart contract to be coordinated with the oracle service such that they can keep track of requests, but this would imply additional costs. What the oracle can do, at the expense of increasing the trust in it, is to independently keep track of the user requests associated to a specific Domain Separator, such that it will not provide new randomness to the user unless the previous one has been uploaded on-chain. An easier alternative would be for the oracle to add a counter to the randomness provided such that it can be verified by the smart contract. Therefore, the user cannot provide a VRF with counter i if the previous one was different from $i-1$. Note that this could lead to collusion if the oracle is not decentralized enough, otherwise a malicious oracle could provide multiple VRF values with counter i until one of them satisfies the user.

A different approach to solve this for the specific case of providing randomness would be to let the oracle provide it encrypted such that it can only be decrypted once uploaded on-chain. This implies that performing multiple off-chain requests is not a problem because the user does not know if the result benefits him or not until it has been included on-chain. However, providing the mechanism for decryption is not trivial, and normally implies advanced cryptographic techniques such as Trusted Execution Environments (TEEs) or Threshold Cryptography. A straightforward way of achieving this could be by using an fhEVM instance [9], where data encrypted with the blockchain's public key of a Fully Homomorphic Encryption (FHE) scheme can be decrypted by a special blockchain committee (different from the oracle network, otherwise there might be collusion) by using a threshold decryption mechanism. However, the main drawback is that fhEVM requires a two-step procedure for decryption that would imply an increment in the cost: (1) the user uploads the encrypted data to the smart contract through a method that calls *requestDecryption* and (2) the gateway sends the ciphertext to the decryption network, receives the result and then uploads it to the smart contract through a callback. A beneficial tweak would be if this was natively supported by Ethereum, in a transparent way for the user, therefore enabling a one-shot unbiased uploading.

VI. Conclusion

In this paper, we introduced an alternative exploratory approach to the problem of verifying data from oracles on-chain. We also provided an efficient prototype solution to make off-chain requests to a blockchain oracle such that the structured responses can be fulfilled by the user himself in a verifiable and standardized way, thanks to EIP 712. To demonstrate usability of this solution, we also provided an implementation for Chainlink's VRF oracle service, where we have proven a $\times 3.17$ speed up in the execution cost. However, some security implications remain to be solved, for which we have provided a few paths to explore.

References

[1] G. Caldarelli, "Understanding the Blockchain Oracle Problem: A Call for Action," *Information*, vol. 11, p. 509, Nov. 2020. Number: 11 Publisher: Multidisciplinary Digital Publishing Institute.

[2] L. Breidenbach, C. Cachin, A. Coventry, S. Ellis, A. Juels, A. Miller, B. Magauran, S. Nazarov, A. Topliceanu, F. Zhang, B. Chan, F. Koushanfar, D. Moroz, and F. Tramer, "Chainlink 2.0: Next Steps in the Evolution of Decentralized Oracle Networks," Apr. 2021.

[3] B. Liu, P. Szalachowski, and J. Zhou, "A First Look into DeFi Oracles," in *2021 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*, pp. 39–48, Aug. 2021.

[4] A. Pasdar, Y. C. Lee, and Z. Dong, "Connect API with Blockchain: A Survey on Blockchain Oracle Implementation," *ACM Comput. Surv.*, vol. 55, pp. 208:1–208:39, Feb. 2023.

[5] R. Mühlberger, S. Bachhofner, E. Castelló Ferrer, C. Di Ciccio, I. Weber, M. Wöhrer, and U. Zdun, "Foundational Oracle Patterns: Connecting Blockchain to the Off-Chain World," in *Business Process Management: Blockchain and Robotic Process Automation Forum* (A. Asatiani, J. M. García, N. Helander, A. Jiménez-Ramírez, A. Koschmider, J. Mendling, G. Meroni, and H. A. Reijers, eds.), (Cham), pp. 35–51, Springer International Publishing, 2020.

[6] R. Bloemen, L. Logvinov, and J. Evans, "EIP-712: Typed structured data hashing and signing," Sept. 2017.

[7] C. Hendrix and R. Lewis, "Survey on Blockchain Privacy Challenges," 2021.

[8] D. Papadopoulos, D. Wessels, S. Huque, M. Naor, J. Včelák, L. Reyzin, and S. Goldberg, "Making NSEC5 Practical for DNSSEC," 2017. Publication info: Preprint.

[9] M. Dahl, C. Danjou, D. Demmler, T. Frederiksen, P. Ivanov, M. Joye, D. Rotaru, N. Smart, and L. Tremlay Thibault, "fhevm: Confidential evm smart contracts using fully homomorphic encryption," 2023.

DiReCT Voting: Dispute Resolution with Cast Timeliness

Jose Luis Martin-Navarro† Antonio M. Larriba Damián López
Valencian Research Institute for Artificial Intelligence (VRAIN),
Universitat Politècnica de València, Spain
† Aalto University, Finland
dlopez@dsic.upv.es

***Abstract*—Este es un resumen extendido de la publicación [1]. Verification in modern e-voting protocols allows voters and the general public to independently confirm the elections results. However, verification alone is insufficient to hold entities accountable for misconduct, or to protect honest participants from false accusations. This limitation is especially critical in voting protocols with multiple authorities, where the ability to identify the specific misbehaving entity is essential. We present DiReCT, the first multiparty protocol that integrates dispute resolution with individual accountability. Our protocol introduces two new disputes: a voting authority blocking access to the election, and a tally authority denying the casting of a ballot. In addition, DiReCT improves timeliness, allowing misconducts to be proactively detected during the elections. As a result, voters can identify and recover from attacks that prevent their ballots from being recorded. Notably, DiReCT achieves these capabilities with low trust assumptions on the authorities.**

***Index Terms*—Electronic Vote; Dispute Resolution; Verifiability; Accountability; Secret sharing**

I. Introduction

Electronic voting systems (e-voting) are modernizing electoral processes, improving voter access, and reducing expenses associated with traditional paper-based elections. However, ensuring the security, transparency, and privacy of the election remains a significant challenge.

Verifiability plays a crucial role in managing trust within electronic voting systems, as it allows stakeholders to independently confirm the integrity of election results [2, 3]. However, verifiability does not provide evidence of the misbehavior, which can lead to disputes where voters claim a wrongdoing from an authority, which defends its honesty. Researchers have proposed dispute resolution [4] to solve the issue, adapting notions of accountability to e-voting. Dispute resolution defines how the validity of a claim can be decided without the need for a trusted judge. The protocol needs to generate unambiguous evidence, verifiable by any third party with only access to the public data. However, the different types of disputes that can arise in e-voting remain largely unexplored. Furthermore, the timeliness of the verification and accountability happens, at best, during the tally, which serves as a post-mortem evaluation, rather than a proactive process.

This paper proposes DiReCT, a new protocol with low trust requirements for voting authorities, which improves the state of the art in dispute resolution. The protocol combines secret sharing and blind signatures to provide elegibility verifiability, privacy and a decentralized tally. DiReCT improves previous work in dispute resolution [4, 5] by including disputes during casting, during the tally, and a new dispute during the registration. It also extends the original definition to a multiparty protocol, emphasizing individual accountability to prevent any of the authorities from acting surreptitiously.

The threat model of this paper considers *covert adversaries* [6], where entities can misbehave as long as they are not detected. This improves previous results that consider semi-trusted authorities [4], assume Honest-but-curious entities [7], or that require trusting the voting authority [2, 8].

In summary, we make the following main contributions:

- The definition of a new type of dispute: DCert, a dispute related to vote suppression, where the voting authority blocks the voter during the authentication and ballot certification.
- The design of DiReCT, a multi-party electronic voting protocol with improved dispute resolution, individual accountability and cast timeliness, which guarantees that voters possess the evidence to resolve disputes before the election's end.
- The security analysis of DiReCT according to the Universal Composability framework [9], proving that the protocol has recorded-as-cast, elegibility verifiability, and the resolution of both new disputes and tally related disputes such as a tally authority removing a ballot from the tally.

II. Related work in Dispute Resolution

The concept of dispute is stated in [4] as *"when a voter claims that the voting authority is dishonest and did not correctly process his ballot while the authority claims to have followed the protocol"*.

Basin et al. also identify three problems that can lead to disputes:

1) The voter is blocked from casting the ballot. The reasons can be technical or social. This is connected with voter suppression and disenfranchisement in traditional elections. Despite defined, this dispute was left out of the scope by the authors.
2) The voter is prevented from verifying its ballot after casting.
3) The voter cannot check if the ballot was recorded or not. The authors describe two types of dispute: either the cast ballot was not recorded (D1); or, an uncast ballot was recorded for the voter (D2).

In this regard, the protocol Themis [5], considers a casting dispute where the voter can verify the consideration of its physical ballot as long as the server or a set of observers are trustworthy.

Although related to accountability, dispute resolution is a stronger requirement, since it requires unambiguity. However, different protocols consider different notions of accountability. In [4], the definition of dispute resolution only includes partial accountability: for a protocol with multiple parties, it is enough to detect that a party misbehaved, but not which one. In the case of Themis, they improve the accountability guarantees by trying to identify which exact party misbehaved. However, for some cases Themis is only able to narrow down the blame to a set of parties, achieving partial accountability. Our proposed protocol DiReCT, is designed to provide individual accountability as shown in the security analysis in the full version of the paper [1]. A comparison between the voting methods that implement dispute resolution is summarized in Table I.

III. Voting Model

DiReCT's voting model is structured around a secret sharing scheme and blind signatures, following the design proposed in SUVS [7]. In this voting model, the secret (ballot) is divided into shares and a commitment is created and blinded with a mask. The voting authority authenticates voters and signs the blinded commitments. The voters send the unblinded signed commitments to the tallying authorities. These authorities pool the shares during the tally to obtain the original votes.

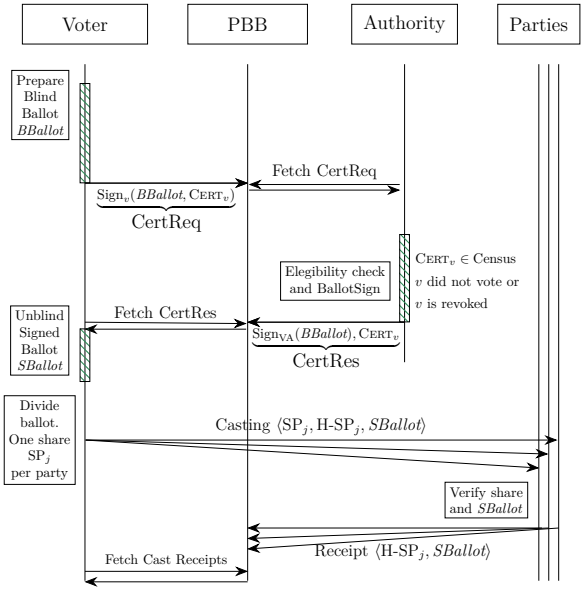


Fig. 1. Voting phase simplified diagram

A. Entities and their roles

The entities involved in the proposed voting protocol are: the *voters*; the *voting authority* (VA); the k *parties* in charge of the decentralized tally procedure; and, a *Public Bulletin Board* (PBB). The PBB behaves as an append-only bulletin board ([10]), providing a consistent view of the information posted, without the possibility of deleting any post. Every entry in the PBB includes the entity and the timestamp.

IV. Protocol

This section describes DiReCT and the steps involved in the the voting and the tally.

A. Voting Phase

In the voting phase voters participate in the election. It starts with the voter crafting the ballot, encoding their vote direction and producing the ballot shares. It follows the ballot certification, where the VA blindly signs the ballots and shares from valid voters in the census. The phase finishes with the casting, where the voters send their shares to the parties to be counted during the tally. Fig. 1 depicts a simplified version of the process.

1) Ballot crafting

The voter starts by encoding the vote direction as an integer C, used as the independent term of a d-degree polynomial $q(x)$:

$$\begin{aligned} q(x) &= a_d x^d + \cdots + a_1 x + C \mod p \\ P &= \langle (x_1, q(x_1)), \ldots, (x_{d+1}, q(x_{d+1})) \rangle \end{aligned} \tag{1}$$

Following the secret sharing scheme [11], the voter can sample a set of points P from $q(x)$, which are used to form the ballot. Any single point from the set does not provide any information about the vote direction, only $d+1$ points allow to interpolate

TABLE I

DISPUTE RESOLUTION COMPARISON IN MULTI-PARTY PROTOCOLS. FOR ACCOUNTABILITY, [4] AND SUVS HAVE PARTIAL ACCOUNTABILITY ◑, THEMIS [5] HAS A MIX OF INDIVIDUAL AND PARTIAL ACCOUNTABILITY ◕, AND THIS WORK HAS COMPLETE INDIVIDUAL ACCOUNTABILITY ●.

	Properties	**[4]**	**Themis**	**SUVS**	**DiReCT**
Disputes	Vote suppression	◑	○	○	●
	Casting	○	◑	○	●
	Recorded	●	●	●	●
	Not Recorded	●	◑	◑	◑
	Accountability	◑	◕	◑	●

$q(x)$ and obtain C. Each one of the k parties in the election receives a subset (SP) of P.

$$P = \bigcup_{i=1}^{k} \text{SP}_i \tag{2}$$

To craft the ballot commitment, the ballot is digested using the election *fresh* value:

$$\begin{aligned} \text{H-P} &:= \text{Hash}(P \mid \textit{fresh}) \\ \text{H-SP}_i &:= \text{Hash}(\text{SP}_i \mid \text{H-P}) \end{aligned} \tag{3}$$

The commitment of the share H-SP_i allows to link each share with the ballot, which will play a role in the casting verification.

2) Ballot certification

The voter certifies its commitments with the VA during the ballot certification, guaranteeing that each voter only votes once. The identity of the voter is used to authenticate, but blind signatures [12] are used to unlink the voter and the ballot.

First, the voter blinds the commitments using the VA public component *va*:

$$\begin{aligned} \text{Blind}\left(\text{H-P}\right) &:= \text{Blind}\left(\text{H-P}, \textit{mask}^{va}\right) \\ \text{Blind}\left(\text{H-SP}\right)_i &:= \text{Blind}\left(\text{H-SP}_i, \textit{mask}_i^{va}\right) \\ \textit{BBallot} &:= \left(\text{Blind}\left(\text{H-P}\right), \left[\text{Blind}\left(\text{H-SP}_i\right)\right]_{i=1}^{k}\right) \end{aligned} \tag{4}$$

Each *mask* needs to be invertible modulo n to unblind the signature, and different from each other. The voter signs *BBallot* using her credentials. This step ensures that every voter can only obtain one ballot (unicity). Only voters with a valid certificate from the census can vote (democracy), and prevents the VA from creating extra ballots (ballot stuffing), since the VA does not possess a voter certificate.

Then, the voter v creates a secure communication with the PBB to pose a request for certification:

$$\text{CerRq}(v \rightarrow \text{PBB}) := v, \text{Sign}_v(\textit{BBallot}, \text{CERT}_v), T_0 \tag{5}$$

The VA retrieves the requests from PBB:certify and processes them: verifies that the certificate CERT_v is valid and that the voter v has not requested a ballot before. If the checks are satisfied, the VA signs the concealed ballot $\text{Sign}_{\text{VA}}(\textit{BBallot})$. The response is sent to PBB:certified:

$$\text{CertRes} := \text{VA}, \langle \text{Sign}_{\text{VA}}(\textit{BBallot}), \text{CerRq}_v \rangle, T_1 \tag{6}$$

The reason for publishing this information is threefold: the voter can confirm that the ballot has been received as intended; it prevents the VA from covertly blocking a voter by not sending the response to the certification; and, every party can check that only voters with valid credentials are certified by the VA.

The voter can easily retrieve the signed data from PBB:certified by unblinding it (Eq. (7)), using the original masks. The process ends with the voter obtaining the signed ballot *SBallot* (Eq. (8)). Hence, the certification of the commitments is performed without revealing the vote.

$$\text{Unblind}\left(\text{Sign}_{\text{VA}}(\textit{BBallot})\right) = \textit{SBallot} \tag{7}$$

$$\textit{SBallot} := \langle \text{Sign}_{\text{VA}}\left(\text{H-P}\right), \left[\text{Sign}_{\text{VA}}\left(\text{H-SP}_i\right)\right]_{i=1}^{k} \rangle \tag{8}$$

3) Ballot Casting

To cast her vote, the voter v sends to each party the signed ballot, a share SP and the share commitment at time T_2, as described in Eq. (9):

$$\text{Casting}(v \rightarrow \text{party}_j) := \langle \text{SP}_j, \text{H-SP}_j, \textit{SBallot} \rangle \tag{9}$$

The party j receiving the share verifies both the signed commitment of the share and the signed ballot:

$$\begin{aligned} \text{Verify}_{\text{VA}}\left(\text{Sign}_{\text{VA}}\left(\text{H-P}\right)\right) &\stackrel{?}{=} \text{H-P} \\ \left[\text{Verify}_{\text{VA}}(\text{Sign}_{\text{VA}}\left(\text{H-SP}_i\right)\right) &\stackrel{?}{=} \text{H-SP}_i\Big]_{i=1}^{k} \\ \text{Hash}(\text{SP}_j \mid \text{H-P}) &\stackrel{?}{=} \text{H-SP}_j \end{aligned} \tag{10}$$

If the checks are successful, the party publishes a receipt in the PBB:cast. The receipt is the content of the cast message without the share. With the receipt the voter can verify that the party received the correct information.

$$\textit{Receipt} := \text{party}_j, \langle \text{H-SP}_j, \textit{SBallot} \rangle, T_3 \tag{11}$$

During the tally, any entity can detect if a party sends a different SP_i than the one signed by checking the information in the PBB:cast.

Only during the tally, when all the shares are published, can a party interpolate the original polynomial and calculate the vote direction

B. Tally Phase

Once the voting finishes, the parties start the tally by sending their shares to the PBB:

$$\text{Tally} := \text{party}_i, \langle \text{SP}_i, \text{H-SP}_i, \mathit{SBallot} \rangle, T_4 \quad (12)$$

The parties collect the shares with an identical *SBallot*, sort them out and reconstruct the original P as seen in Eq. (2). With P it is straightforward to obtain the original polynomial $q(x)$, for example using Lagrange's polynomials. The direction of vote can then be obtained from $q(x)$ by obtaining the independent term $C = q(0) \mod p$.

V. Conclusions

This work presented DiReCT, an electronic voting protocol that improves the current state of the art in terms of accountability. In previous definitions of dispute resolution, for elections conducted by multiple parties the requirement was to detect when *any* of the parties misbehaves [4], or to partially account them [5]. We extended this requirement by providing individual accountability, which allows the identification of the exact misbehaving party.

In our protocol, the timeliness guarantee described in [4] is effective during the casting, not at the end of the election. This ensures that the attack is detected before the tally, allowing the voter to recover and continue voting. Furthermore, DiReCT is the first protocol that addresses dispute resolution in the case of vote suppression, an attack during the registration phase that can block voters from participating in the election.

With this work we highlight the benefits of including dispute resolution as part of voting protocols and showcase that is possible to achieve it in a multi party protocol. The threat model includes covert adversaries, which we believe is a realistic assumption for voting protocols that combines nicely with the verification and accountability focus of DiReCT.

The full version of this work includes the related work and background, covering in detail the secret sharing and blind signatures used as a foundation of the protocol. It also analyses the security of the protocol, proving the resolution of disputes and the effectivity against attacks with the Universal Composability framework. We also included a complexity and scalability analysis of DiReCT, which shows an estimated needed storage in the PBB of 25.5 GB for an election with one million voters. As a future work we will address the formal verification and implementation of our results.

Acknolwedgments

Supported by INCIBE's Chair funded by the EU-NextGenerationEU through the Spanish government's Plan de Recuperación, Transformación y Resiliencia.

References

[1] J. L. Martin-Navarro, A. M. Larriba, and D. López, "DiReCT Voting: Dispute Resolution with Cast Timeliness," *Informatica*, pp. 1–34, 2025.

[2] B. Adida, "Helios: Web-based open-audit voting." in *USENIX security symposium*, vol. 17, 2008, pp. 335–348.

[3] R. Küsters, J. Liedtke, J. Müller, D. Rausch, and A. Vogt, "Ordinos: A Verifiable Tally-Hiding E-Voting System," in *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*, Sep. 2020, pp. 216–235.

[4] D. Basin, S. Radomirović, and L. Schmid, "Dispute Resolution in Voting," in *2020 IEEE 33rd Computer Security Foundations Symposium (CSF)*, Jun. 2020, pp. 1–16.

[5] M. Bougon, H. Chabanne, V. Cortier, A. Debant, E. Dottax, J. Dreier, P. Gaudry, and M. Turuani, "Themis: An On-Site Voting System with Systematic Cast-as-intended Verification and Partial Accountability," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Nov. 2022, pp. 397–410.

[6] Y. Aumann and Y. Lindell, "Security Against Covert Adversaries: Efficient Protocols for Realistic Adversaries," *Journal of Cryptology*, vol. 23, no. 2, pp. 281–343, Apr. 2010.

[7] A. M. Larriba and D. López, "SUVS: Secure Unencrypted Voting Scheme," *Informatica*, vol. 33, no. 4, pp. 749–769, Dec. 2022.

[8] D. Chaum, A. Essex, R. Carback, J. Clark, S. Popoveniuc, A. Sherman, and P. Vora, "Scantegrity: End-to-End Voter-Verifiable Optical- Scan Voting," *IEEE Security & Privacy*, vol. 6, no. 3, pp. 40–46, May 2008.

[9] R. Canetti, "Universally composable security: a new paradigm for cryptographic protocols," in *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, Oct. 2001, pp. 136–145, iSSN: 1552-5244.

[10] J. Heather and D. Lundin, "The Append-Only Web Bulletin Board," in *Formal Aspects in Security and Trust*. Springer, 2009, pp. 242–256.

[11] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, Nov. 1979.

[12] D. Chaum, "Blind Signatures for Untraceable Payments," in *Advances in Cryptology*. Boston, MA: Springer US, 1983, pp. 199–203.

TRITON: Hacia un Marco de Ciberseguridad para Redes de Agua Inteligentes

Isaac TORRES PAREDES
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
IsaacDavid.Torres1@alu.uclm.es

Antonio SANTOS-OLMO
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
Antonio.Santosolmo@uclm.es

David G. ROSADO
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
David.GRosado@uclm.es

Luis Enrique SÁNCHEZ
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
Luise.Sanchez@uclm.es

Eduardo FERNÁNDEZ-MEDINA
Grupo de Seguridad y Auditoría–GSyA
UCLM, Ciudad Real
Eduardo.Fdezmedina@uclm.es

***Resumen*—La digitalización de redes de agua inteligentes las convierte en infraestructuras críticas urbanas, pero las expone a ciberriesgos significativos capaces de comprometer la seguridad del suministro, desafío especialmente agudo en regiones en desarrollo. Este trabajo presenta una revisión sistemática PRISMA de ciberseguridad en estos sistemas. De 467 artículos identificados, se analizaron 14 estudios primarios contra diez características fundamentales de seguridad: autenticación, cifrado, monitorización continua, respuesta a incidentes, integridad, disponibilidad, escalabilidad, defensa contra amenazas avanzadas, análisis de malware y resiliencia. Los hallazgos revelan una notable falta de marcos holísticos y aplicados, predominando soluciones teóricas que ignoran desafíos operativos específicos como escasez de recursos y personal especializado. Como contribución principal, se propone TRITON: un marco modular de ciberseguridad que integra protección, detección y respuesta, diseñado para implementación incremental por fases. El presente artículo establece las bases teóricas y la arquitectura conceptual del marco, quedando la validación empírica y estudios de caso como trabajo futuro.**

***Index Terms*—Agua inteligente, ciberseguridad, infraestructuras críticas, resiliencia, SCADA.**

I. Introducción

La modernización de los recursos hídricos mediante redes de agua inteligentes (*smart water networks*) es fundamental para la gestión de infraestructuras críticas urbanas [1], [2]. La integración de tecnologías como el Internet de las Cosas (IoT) y los sistemas de control industrial (SCADA) optimiza la monitorización y operación de los sistemas hídricos. Sin embargo, esta profunda digitalización introduce una superficie de ataque sin precedentes, convirtiendo la ciberseguridad en un desafío primordial [3], [4].

La creciente interconexión de estos sistemas ciberfísicos aumenta su exposición a ciberataques que pueden comprometer la seguridad del suministro, la calidad del agua y la integridad de los datos de los ciudadanos [5], [6]. A pesar de la criticidad del sector, la literatura revela que la adopción de medidas de ciberseguridad robustas a menudo va a la zaga del rápido avance tecnológico, dejando vulnerabilidades significativas sin abordar [7], [8].

En este contexto, el presente trabajo realiza una revisión sistemática de la literatura centrada exclusivamente en la ciberseguridad de las redes de agua inteligentes. El objetivo es identificar las soluciones existentes, las tendencias y, fundamentalmente, las brechas de investigación que impiden el desarrollo de infraestructuras hídricas ciber-resilientes. A partir de este análisis riguroso, se presenta la arquitectura conceptual de TRITON, un marco de ciberseguridad especializado cuya implementación y validación empírica constituyen el trabajo futuro de esta línea de investigación.

La propuesta conceptual, denominada marco TRITON (*Technologies for Intelligent, Transparent, Optimised and Neutral-impact Networks*), se concibe en su origen como un sistema holístico que integra tres pilares: Ciberseguridad (CI), Sostenibilidad (SO) e Inteligencia Artificial (IA). No obstante, dada la profundidad y criticidad del eje de seguridad, el presente artículo se centrará exclusivamente en el desarrollo conceptual del pilar de Ciberseguridad, dejando el análisis completo de los otros dos ejes para futuras publicaciones. Este artículo constituye así la fase fundacional del proyecto TRITON: establece las bases teóricas mediante revisión sistemática, identifica brechas específicas mediante análisis cualitativo y cuantitativo, y propone la arquitectura conceptual del marco como respuesta fundamentada a dichas deficiencias.

El resto del artículo se organiza de la siguiente manera: la Sección II detalla la metodología de revisión sistemática utilizada para la selección de estudios. La Sección III recopila y describe los 14 trabajos primarios de ciberseguridad. A continuación, la Sección IV presenta el análisis sintético de estos hallazgos, identificando las brechas de la literatura mediante una comparativa sistemática de características de ciberseguridad. Basado en este análisis, la Sección V introduce el diseño conceptual del marco de ciberseguridad TRITON como solución a las deficiencias identificadas. Finalmente, la Sección VI expone las conclusiones del estudio y las futuras líneas de investigación para la implementación y validación del marco.

II. Metodología de Revisión

Para alcanzar los objetivos propuestos, se llevó a cabo una revisión sistemática de la literatura (SLR) siguiendo las directrices PRISMA. El proceso se diseñó para identificar, evaluar y sintetizar la investigación existente en el dominio de la ciberseguridad aplicada a las redes de agua inteligentes, con

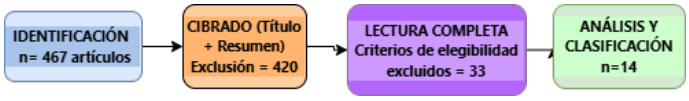


Figura 1. Diagrama PRISMA: flujo de selección de estudios primarios

el fin de identificar las brechas de conocimiento que justifican la creación de un nuevo marco de trabajo.

II-A. Estrategia de Búsqueda

La búsqueda de estudios primarios se centró exclusivamente en la ciberseguridad aplicada a redes de agua inteligentes, realizando consultas en Scopus, Web of Science, IEEE Xplore, Springer, Elsevier y Google Scholar. Se diseñaron cadenas de búsqueda combinando palabras clave del dominio con operadores booleanos.Por ejemplo: *(smart water OR intelligent water) AND (cybersecurity OR cyber security) AND (SCADA OR IoT OR cyber-physical).*

II-B. Criterios de Selección y Proceso de Cribado

El proceso de selección se limitó a publicaciones de los últimos diez años (2015–2025) y se ejecutó en dos fases. Los criterios de inclusión fueron: enfoque en ciberseguridad para redes de agua inteligentes, propuesta de marcos/procesos/modelos de seguridad, publicaciones revisadas por pares en inglés o español. Se excluyeron: estudios fuera del período temporal, enfoques genéricos de IoT sin aplicación específica al agua, revisiones secundarias y literatura gris. Primero, se aplicó el cribado por título, palabras clave y resumen. En la segunda fase, los trabajos preseleccionados fueron sometidos a lectura completa, conservando únicamente aquellos con contribución directa al campo.

II-C. Extracción y Clasificación de Datos

La estrategia de búsqueda arrojó 467 artículos iniciales. El proceso de cribado, ejecutado en dos fases según directrices PRISMA (Figura 1), redujo significativamente el conjunto: primero, mediante evaluación de título y resumen se preseleccionaron 47 artículos (exclusión inicial: 420); posteriormente, la lectura completa, aplicando criterios de elegibilidad directa al campo de ciberseguridad en agua, resultó en 14 estudios primarios (exclusión final: 33). Estos estudios se clasificaron según su contribución principal en: (i) *Proceso* (3 estudios, 21 %), si describe actividades planificadas; (ii) *Marco de Trabajo* (7 estudios, 50 %), si propone una estructura conceptual; y (iii) *Modelo* (4 estudios, 29 %), si presenta un artefacto específico.

III. Recopilación de datos

Esta sección presenta la síntesis detallada de los 14 estudios primarios del eje de Ciberseguridad. Cada trabajo se analiza considerando su enfoque metodológico, contribuciones principales y limitaciones identificadas, proporcionando así la base empírica para el análisis comparativo posterior. La estructura organizativa por tipo de contribución (Procesos, Marcos de Trabajo y Modelos) facilita la identificación de patrones comunes y brechas específicas en cada categoría.

A continuación se sintetizan los hallazgos agrupados por naturaleza de contribución, destacando tanto los logros como las deficiencias que justifican la necesidad de enfoques más integrales.

III-1. Contribuciones basadas en Procesos: Los estudios aquí incluidos detallan secuencias operativas, procedimientos técnicos y flujos de trabajo sistemáticos implementados para la protección de activos digitales y la gestión de incidentes de seguridad.

La creación del conjunto de datos BATADAL 2.0 es presentada por Murillo et al. [17], quienes lo desarrollaron con el simulador DHALSIM con el objetivo de mejorar los sistemas de detección de intrusiones en redes de agua inteligentes. Un logro significativo del estudio es la integración de los datos de proceso con el tráfico de red, lo que facilita el desarrollo de técnicas de detección de ataques más avanzadas al permitir la replicación de condiciones anómalas. Sin embargo, una limitación clave del artículo es el alcance restringido de los tipos de ataque representados, lo que puede limitar la aplicabilidad del dataset en escenarios más complejos y realistas.

La importancia de la ciberseguridad en los sistemas de monitoreo y control del sector energético, especialmente en infraestructuras críticas como las redes inteligentes (smart grids) o las plantas nucleares, es un desafío relevante. En respuesta, Alghassab [9] presenta un enfoque innovador que utiliza una combinación de procesos de Toma de Decisiones Multicriterio (MCDM) basados en el Proceso de Jerarquía Analítica (AHP) y la técnica TOPSIS para evaluar las vulnerabilidades y riesgos cibernéticos.

Este método proporciona un marco para priorizar las características de seguridad y apoya la identificación de soluciones efectivas. Sin embargo, el propio estudio presenta debilidades, como la falta de una validación extensa de los resultados mediante casos de estudio reales, lo que limita su aplicación práctica y podría mejorarse incluyendo una orientación más clara sobre cómo implementar los hallazgos. Aunque se categoriza como un proceso sólido, este enfoque necesita ampliar su aplicabilidad para un uso más efectivo en el sector.

Se han propuesto métodos para optimizar la infraestructura crítica de las redes de suministro de agua mediante la integración de tecnologías avanzadas (TIC). En esta línea, Bosco et al. [4] presentan un estudio cuyo logro significativo es el desarrollo de un modelo hidráulico dinámico que permite simulaciones en tiempo real de los parámetros operativos. El artículo también destaca el cumplimiento de estándares de ciberseguridad, como la Ley de Ciberseguridad, que guía el diseño de un sistema seguro.

El sistema propuesto mejora la fiabilidad y eficiencia del suministro de agua al permitir una respuesta rápida en situaciones de crisis. Sin embargo, la propuesta muestra limitaciones en su énfasis en la implementación de soluciones tecnológicas sin considerar ampliamente los costos y el mantenimiento a largo plazo de los sistemas inteligentes. Además, aunque Bosco et al. subrayan la necesidad de un sistema de comunicación eficaz, no abordan en profundidad los desafíos prácticos que enfrentan las redes de suministro de agua en áreas rurales o menos desarrolladas. En conclusión, este artículo se clasifica dentro de Procesos por su enfoque en la mejora de la gestión y operación de las redes.

III-2. Contribuciones basadas en Marcos de Trabajo: Estas contribuciones proponen arquitecturas conceptuales, conjuntos de directrices y estándares, o guías metodológicas diseñadas para establecer, gestionar y mejorar la postura de seguridad de una organización.

Una revisión sistemática (PRISMA) realizada por Bello et al. [10] analiza los desafíos de ciberseguridad en la infraestructura hídrica de Australia. El estudio identifica un aumento en la investigación, pero también una notable falta de colaboración entre la academia, el gobierno y el sector privado.

Una revisión de la literatura sobre ciberseguridad en sistemas de agua, presentada por Tuptuk et al. [3], concluye que la investigación se centra en el agua potable mientras ignora áreas críticas como las aguas residuales. El estudio, además, evidencia la ausencia de un marco de seguridad integral.

La protección de infraestructuras críticas de agua, basándose en el Marco de Ciberseguridad del NIST, es abordada por Clark y Panguluri [6]. Su trabajo propone medidas prácticas, pero también señala debilidades clave, como la falta de políticas unificadas y la desigualdad de recursos entre las empresas del sector.

En una línea similar, Kumar et al. [11] presentan un análisis de plataformas de inteligencia sobre ciberamenazas (CTI) y estándares de intercambio como TAXII y CybOX. El estudio, sin embargo, tiene como principal limitación la falta de análisis sobre los desafíos de implementación en infraestructuras críticas, así como los problemas de confianza dentro de las comunidades de intercambio.

Ali [12] propone un enfoque cíclico para la gestión de riesgos en Sistemas de Control Distribuido (DCS), alineado con las guías de FISMA y NIST. Sin embargo, su principal debilidad es la falta de un análisis detallado sobre la implementación práctica de dicho enfoque en entornos reales.

Un marco de seguridad para la industria del agua en Indonesia es diseñado por Muttaqin y Ramli [13], quienes integran estándares internacionales (C2M2, ISO 27019) y regulaciones locales. La propuesta, sin embargo, se ve limitada por la falta de investigación previa específica para dicho sector y por la cuestionable aplicabilidad de estándares adaptados de otros ámbitos, como el financiero.

El trabajo de Bosco et al. [4], derivado del proyecto STOP-IT, desarrolla un marco de gestión de riesgos integrado (basado en ISO 31000) para amenazas ciberfísicas. Sus principales barreras para la adopción, sin embargo, son la notable brecha en alfabetización digital en materia de ciberseguridad y la complejidad de su plataforma tecnológica.

III-3. Contribuciones basadas en Modelos: Finalmente, los estudios clasificados en esta categoría presentan representaciones abstractas ya sean formales, matemáticas o conceptuales utilizadas para analizar amenazas, evaluar riesgos o simular comportamientos y defensas en el campo de la seguridad digital.

La creciente preocupación por la ciberseguridad en los sistemas inteligentes de gestión de aguas pluviales es abordada por Lin et al. [14]. Su estudio introduce un método innovador para cuantificar los riesgos de inundación causados por ataques de inyección de datos falsos (FDI), desarrollando un marco matemático que vincula el ciberataque con la respuesta física del sistema. Este modelo presenta, sin embargo, debilidades significativas: se basa en supuestos poco realistas (como una tasa de ataque del 100 % y patrones de lluvia hipotéticos) y su metodología se limita a un solo detector, restringiendo su adaptabilidad.

El estudio de Kovács [15] adopta un enfoque multidisciplinario, explorando la intersección entre las vulnerabilidades de ciberseguridad en el sector del agua y sus implicaciones para la migración humana. Basándose en estudios de caso (Noruega, Israel, EE. UU.), destaca los riesgos para la salud pública y la estabilidad económica derivados de ciberataques. Aunque el modelo propone integrar tecnologías emergentes como IA y blockchain para la resiliencia, presenta limitaciones notables: carece de estrategias de implementación específicas y no profundiza lo suficiente en la interrelación con las políticas migratorias.

La creciente relevancia de la gestión inteligente del agua y sus riesgos de ciberseguridad en Hungría y Europa Central es examinada por Szádeczky [16]. El autor realiza una extensa revisión de la literatura para identificar amenazas y vulnerabilidades en infraestructuras hídricas críticas, enfatizando la necesidad de medidas de seguridad robustas. Sin embargo, el estudio carece de una base empírica sólida que respalde sus hallazgos teóricos, lo que limita la aplicabilidad práctica de sus recomendaciones. Aunque se mencionan posibles soluciones, el modelo no proporciona un marco detallado para su implementación efectiva.

Tras haber detallado las contribuciones individuales de la literatura revisada en ciberseguridad, agrupadas por procesos, marcos de trabajo y modelos, la siguiente sección se dedica a realizar un análisis sintético de estos resultados. El objetivo es interpretar estas observaciones de forma conjunta para identificar los patrones recurrentes, las debilidades comunes y el vacío de investigación principal que existe actualmente en el campo de la ciberseguridad para redes de agua inteligentes.

IV. Análisis y Resultados

El análisis revela un campo de investigación activo, pero con enfoques marcadamente diferenciados. Las contribuciones pueden agruparse en tres categorías principales según su naturaleza: aquellas que proponen procesos operativos, las que establecen marcos de trabajo (frameworks) conceptuales, y las que desarrollan modelos analíticos para amenazas específicas.

En general, la literatura muestra una tendencia positiva hacia la estructuración de la ciberseguridad en el sector. Se observa un esfuerzo por adoptar estándares consolidados como los del NIST e ISO y la implementación de soluciones como Sistemas de Detección de Intrusos (IDS) y cortafuegos. Asimismo, se desarrollan procesos y modelos para evaluar vulnerabilidades y simular ataques concretos, como los de inyección de datos falsos (FDI).

Sin embargo, las vulnerabilidades más significativas identificadas no son puramente tecnológicas, sino también organizacionales: destacan la falta de actualizaciones de software y una formación inadecuada del personal, junto a la ausencia de protocolos de respuesta a incidentes. Este hecho se enmarca en un hallazgo más profundo y recurrente en las tres categorías: una profunda desconexión entre las soluciones teóricas propuestas y las realidades prácticas y operativas del sector del agua. Los procesos a menudo carecen de validación en entornos reales; los marcos de trabajo tienden a ser demasiado genéricos; y los modelos se basan en supuestos simplificados que limitan su aplicabilidad.

Con el fin de evaluar sistemáticamente las propuestas existentes, se definieron diez características fundamentales

de ciberseguridad derivadas de marcos consolidados como NIST Cybersecurity Framework e IEC 62443, así como de literatura especializada en sistemas SCADA/IoT para infraestructuras críticas. Estas características representan las capacidades elementales que todo sistema ciberfísico debe integrar para garantizar protección, detección y respuesta efectiva ante ciberamenazas.

Las características que se ha evaluado son: i) AUT. Autenticación robusta (validación segura de la identidad). ii) ENC. Cifrado (protección de datos en tránsito y en reposo). iii) AUTD. Autorización dinámica (control de permisos basado en el contexto). iv) MON. Monitorización continua (detección de amenazas en tiempo real). v) INT. Integridad de los datos (asegurar que los datos no sean alterados). vi) DISP. Disponibilidad (garantizar el acceso ininterrumpido a los sistemas). vii) ESC. Escalabilidad (capacidad para soportar un crecimiento de usuarios/dispositivos). viii) APT. Protección contra Amenazas Persistentes Avanzadas.ix) ANM. Análisis de malware (detección de software malicioso). x) RESP. Respuesta a incidentes (estrategias para mitigar y resolver ataques). Se mapeó sistemáticamente la presencia de estas características en los 14 artículos revisados, y los resultados se consolidan en la Tabla 1, la cual ofrece una instantánea comparativa de las capacidades de las propuestas actuales, resaltando los aspectos comúnmente abordados y aquellos sistemáticamente descuidados.

Como se observa en la Tabla I, los resultados de la comparativa son reveladores. Es notable la baja adopción de capacidades de Respuesta a Incidentes (RESP) y Análisis de Malware (ANM), lo que sugiere una postura de seguridad predominantemente reactiva. La mayoría de los trabajos se centran en la autenticación y el cifrado, pero descuidan la monitorización continua y la protección contra amenazas avanzadas (APT). Esta disparidad demuestra que la literatura actual ofrece soluciones parciales y subraya la necesidad de un marco que integre las diez características de forma equilibrada, tal como lo propone TRITON.

V. Propuesta TRITON

Como respuesta a la brecha identificada en la literatura, la falta de un marco de ciberseguridad holístico y práctico para el sector del agua, se propone el marco de trabajo TRITON. Aunque su nombre (*Technologies for Intelligent, Transparent, Optimised and Neutral-impact Networks*) refleja una visión amplia que integra tres pilares (Ciberseguridad, Sostenibilidad e Inteligencia Artificial), en este artículo se presenta su pilar fundamental: un marco integral y modular de ciberseguridad diseñado específicamente para los desafíos de las redes de agua inteligentes.

V-A. Principios de Diseño y Comparación con Marcos Existentes

Basándose directamente en las brechas identificadas, TRITON se fundamenta en seis principios clave: *(1) Modularidad y escalabilidad* para implementación incremental según recursos disponibles; *(2) Especificidad sectorial*, considerando sistemas SCADA legacy, infraestructura distribuida y personal con formación limitada; *(3) Implementación pragmática por fases* con tres etapas de madurez progresiva; *(4) Defensa en profundidad integrada*, superando enfoques fragmentados; *(5) Orientación a la acción*, proporcionando guías de priorización e implementación; y *(6) Habilitación tecnológica inteligente*, empleando IA como acelerador de detección, no como fin en sí mismo.

La Tabla II compara TRITON con marcos consolidados (NIST CSF e IEC 62443). Mientras que NIST CSF ofrece un lenguaje común de alto nivel e IEC 62443 especificaciones técnicas para sistemas OT, TRITON se posiciona como un puente pragmático entre estos estándares y las realidades operativas del sector agua, particularmente en contextos con limitaciones de recursos. Su diferenciación principal radica en la implementación incremental diseñada específicamente para el sector, considerando ciclos presupuestarios limitados, infraestructura distribuida con conectividad variable, y la necesidad crítica de no interrumpir el servicio durante mejoras de seguridad.

V-B. Arquitectura Conceptual del Marco

El marco TRITON se articula en torno a tres capacidades estratégicas interconectadas que actúan de forma sinérgica, organizadas según un modelo de defensa en profundidad por capas (Figura 2).

- Capacidades de Protección: Constituyen la base del marco y actúan como primera línea de defensa. Incluyen: control de acceso robusto con autenticación multifactor adaptada a entornos SCADA (AUT); cifrado de datos en tránsito entre sensores IoT, PLCs y centros de control mediante protocolos ligeros (ENC); seguridad específica para dispositivos IoT (sensores de caudal, presión, calidad) con arranque seguro y actualizaciones OTA protegidas; y segmentación de red OT/IT con zonas desmilitarizadas (DMZ). Estas medidas garantizan la integridad (INT) y confidencialidad de la información.
- Capacidades de Detección: Complementan la protección con un enfoque proactivo mediante: monitorización continua 24/7 con integración SIEM adaptado al sector agua (MON); detección de anomalías mediante modelos de machine learning entrenados con patrones normales de operación hídrica para identificar ataques FDI o manipulación de válvulas/bombas; análisis de malware específico para entornos OT (ANM); e identificación de amenazas persistentes avanzadas (APT) mediante correlación de eventos. Es aquí donde la Inteligencia Artificial actúa como habilitador táctico, permitiendo análisis de grandes volúmenes de datos operacionales para detectar patrones sutiles.
- Capacidades de Respuesta y Resiliencia: Aseguran la continuidad del servicio mediante: protocolos formalizados de respuesta a incidentes específicos del sector agua (manipulación de calidad, interrupción de suministro, robo de datos) con cadenas de notificación a autoridades (RESP); garantía de disponibilidad con redundancia de sistemas críticos y failover automático (DISP); escalabilidad para mantener niveles de seguridad durante expansión (ESC); y procesos post-incidente para restauración segura y lecciones aprendidas.

V-C. Modelo de Implementación por Fases

Para garantizar su aplicabilidad en regiones con recursos limitados, TRITON se concibe con un enfoque de implemen-

Tabla I
TABLA COMPARATIVA: PRESENCIA DE CARACTERÍSTICAS POR ARTÍCULO

Artículo	AUT	ENC	AUTD	MON	INT	DISP	ESC	APT	ANM	RESP
Szádeczky [16]	Y	N	Y	Y	N	Y	N	N	P	P
Clark et al. [6]	P	Y	N	N	Y	Y	Y	P	N	Y
Bosco et al. [4]	Y	Y	N	Y	Y	N	Y	Y	Y	P
Muttaqin et al. [13]	Y	N	P	Y	Y	Y	N	N	Y	Y
Lin et al. [14]	Y	Y	Y	P	P	Y	Y	Y	P	Y
Bello et al. [10]	Y	N	Y	Y	Y	Y	P	Y	Y	N
Alghassab [9]	Y	Y	N	N	Y	N	P	P	Y	N
Kumar et al. [11]	P	P	N	Y	N	Y	Y	N	N	P
Małka et al. [18]	Y	Y	N	P	N	Y	P	P	N	Y
Tuptuk et al. [3]	P	Y	Y	N	N	N	P	Y	Y	Y
Murillo et al. [17]	Y	Y	Y	N	Y	N	Y	N	Y	P
Kovács [15]	N	Y	N	P	Y	P	N	N	P	Y
Józefowicz et al. [19]	P	Y	Y	P	N	N	Y	Y	N	Y
Adedeji et al. [20]	P	N	N	Y	Y	N	N	P	Y	N
TRITON (T)	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Y = Cumple; N = No cumple; P = Parcialmente; T = TRITON

Tabla II
COMPARACIÓN CON MARCOS DE CIBERSEGURIDAD ESTABLECIDOS

Característica	NIST CSF	IEC 62443	TRITON
Específico sector agua	No	No	Sí
Implementación por fases definida	No	Parcial	Sí (3 fases)
Adaptado a recursos limitados	No	Parcial	Sí
Rol de IA explícito	No	No	Sí
Protocolos respuesta específicos agua	No	No	Sí
Validación empírica	Amplia	Amplia	Trabajo futuro

tación incremental:

- Fase 1 - Fundacional (0-12 meses): Protección esencial de activos críticos. Prioridades: autenticación multifactor para SCADA, cifrado de comunicaciones críticas, segmentación inicial OT/IT, formación básica del personal. Inversión: baja-media. Resultado: reducción significativa de superficie de ataque.
- Fase 2 - Detección Proactiva (12-24 meses): Monitorización continua y detección de anomalías. Prioridades: despliegue de SIEM, modelos de IA para detección, SOC básico (puede ser tercerizado), inteligencia sobre amenazas. Inversión: media. Resultado: capacidad de detectar ataques en curso.
- Fase 3 - Resiliencia Total (24+ meses): Protocolos avanzados de respuesta. Prioridades: procedimientos formalizados con simulacros, redundancia y failover, integración con autoridades, ciclo continuo de mejora. Inversión: media-alta. Resultado: organización ciber-resiliente.

Esta hoja de ruta progresiva permite madurar la postura de seguridad de forma controlada, con retorno de inversión demostrable en cada etapa y alineada con ciclos presupuestarios típicos del sector público.

V-D. Alineación con Brechas Identificadas

Todo el marco está orientado al objetivo estratégico de asegurar la *Sostenibilidad y Continuidad del Servicio*, garantizando que el suministro de agua sea resiliente y esté siempre



Figura 2. Estructura de defensa en profundidad

disponible. La ciberseguridad es un medio para este fin, no un objetivo aislado.

La Tabla III sintetiza cómo cada componente de TRITON responde directamente a deficiencias documentadas en el estado del arte, consolidando el marco como una solución dirigida y aplicable.

Tabla III
ALINEACIÓN DE LOS COMPONENTES DE TRITON CON LAS BRECHAS DE CIBERSEGURIDAD

Brecha Identificada	Componente TRITON
Falta de validación y aplicabilidad práctica	Implementación por Fases gradual
Marcos genéricos, no adaptados a recursos limitados	Diseño modular y escalable
Carencia de protocolos de respuesta	Capacidades de Respuesta y Resiliencia formalizadas
Seguridad deficiente en IoT y SCADA	Capacidades de Protección con énfasis en IoT
Detección reactiva y simplista	Capacidades de Detección proactiva con IA
Ausencia de guías de priorización	Modelo de 3 fases con objetivos claros
Desconexión teoría-práctica	Principios centrados en pragmatismo sectorial

Con la presentación de esta propuesta conceptual funda-

mentada en el análisis sistemático de la literatura, se culmina la contribución principal de esta investigación. La siguiente sección expone las conclusiones y las líneas futuras de validación e implementación del marco TRITON.

VI. Conclusiones

Esta revisión sistemática confirma que la digitalización de las redes de agua mediante IoT y sistemas ciberfísicos ha creado una superficie de ataque sin precedentes, convirtiendo la ciberseguridad en un desafío primordial para garantizar la continuidad del suministro.

El hallazgo más revelador es la notable escasez de soluciones de ciberseguridad robustas, integrales y prácticas para redes de agua inteligentes. El riguroso proceso de selección, que redujo 467 artículos a solo 14 estudios primarios evaluados contra diez características fundamentales de ciberseguridad, evidencia una desconexión recurrente entre marcos teóricos y realidades operativas del sector, particularmente en respuesta a incidentes (RESP), análisis de malware (ANM) y protección contra amenazas persistentes avanzadas (APT).

Como respuesta, este trabajo presenta la arquitectura conceptual del marco TRITON, fundamentada en seis principios de diseño derivados de las brechas identificadas. TRITON se diferencia de marcos genéricos (NIST CSF, IEC 62443) por su especificidad sectorial, implementación pragmática por fases adaptada a recursos limitados, e integración cohesionada de protección, detección y respuesta orientadas a la sostenibilidad del servicio. Este artículo constituye la fase fundacional del proyecto, estableciendo bases teóricas mediante análisis riguroso.

Como trabajo futuro, se plantea la validación empírica del marco TRITON mediante su implementación en un escenario real de red de agua potable, evaluando su aplicabilidad práctica, costes, impacto operativo y aceptación del personal técnico. Adicionalmente, se explorarán extensiones mediante métricas cuantitativas de resiliencia ciber-física, tecnologías emergentes como blockchain para integridad de datos, y el desarrollo completo de los pilares de Sostenibilidad e Inteligencia Artificial, completando así la visión holística de TRITON.

Agradecimientos

Esta publicación no habría sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MICIU/AEI/10.13039/501100011033 y la Unión Europea (Chist-Era Program); KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MICIU/AEI/10.13039/501100011033 /FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la JCCM, y FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; y Grant RED2024-154240-T financiado por MICIU/AEI/10.13039/501100011033.

Referencias

[1] A. di Nardo, D. Boccelli, M. Herrera, E. Creaco, A. Cominola, R. Sitzenfrei, R. Taormina: "Smart Urban Water Networks: Solutions, Trends and Challenges", en *Water*, vol. 13, n. 4, pp. 501, 2021. DOI: 10.3390/w13040501

[2] M. Cuguero-Escofet, V. Puig: .Advances in the Monitoring, Diagnosis and Optimisation of Water Systems", en *Sensors*, vol. 23, n. 6, pp. 3256, 2023. DOI: 10.3390/s23063256

[3] N. Tuptuk, P. Hazell, J. Watson, S. Hailes: .A Systematic Review of the State of Cyber-Security in Water Systems", en *Water*, vol. 13, n. 1, pp. 81, 2021. DOI: 10.3390/w13010081

[4] C. Bosco, G.S. Raspati, K. Tefera, H. Rishovd, R. Ugarelli: "Protection of Water Distribution Networks against Cyber and Physical Threats: The STOP-IT Approach Demonstrated in a Case Study", en *Water*, vol. 14, n. 23, pp. 3895, 2022. DOI: 10.3390/w14233895

[5] A. Hassanzadeh, A. Rasekh, S. Galelli, M. Aghashahi, R. Taormina, et al.: .A Review of Cybersecurity Incidents in the Water Sector", en *Journal of Environmental Engineering*, vol. 146, n. 5, pp. 04020033, 2020. DOI: 10.1061/(ASCE)EE.1943-7870.0001686

[6] R.M. Clark, S. Panguluri: "Protecting water and wastewater utilities from cyber-physical threats", en *Water and Environment Journal*, vol. 32, n. 3, pp. 339-350, 2018. DOI: 10.1111/wej.12340

[7] R. Taormina, S. Galelli, N. Tippenhauer, E. Salomons, A. Ostfeld, et al.: "Battle of the Attack Detection Algorithms: Disclosing Cyber Attacks on Water Distribution Networks", en *Journal of Water Resources Planning and Management*, vol. 144, n. 8, pp. 04018059, 2018. DOI: 10.1061/(ASCE)WR.1943-5452.0000969

[8] F.A. Batarseh, A. Kulkarni, C. Sreng, J. Lin, S. Maksud: .ACWA: an AI-driven cyber-physical testbed for intelligent water systems", en *Water Practice & Technology*, vol. 18, n. 12, pp. 3399-3414, 2023. DOI: 10.2166/wpt.2023.197

[9] M. Alghassab: .Analyzing the Impact of Cybersecurity on Monitoring and Control Systems in the Energy Sector", en *Energies*, vol. 15, n. 1, pp. 218, 2021. DOI: 10.3390/en15010218

[10] A. Bello, S. Jahan, F. Farid, F. Ahamed: .A Systemic Review of the Cybersecurity Challenges in Australian Water Infrastructure Management", en *Water*, vol. 15, n. 1, pp. 168, 2022. DOI: 10.3390/w15010168

[11] A. Kumar, K. Sharma, S. Jain, D.K. Sharma, A. Aggarwal: "Trends in Existing and Emerging Cyber Threat Intelligence Platforms", en *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, n. 12, pp. 3194-3203, 2019. DOI: 10.35940/ijitee.L3188.1081219

[12] S. Ali: Çybersecurity management for distributed control system: systematic approach", en *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, n. 11, pp. 10091-10105, 2021. DOI: 10.1007/s12652-020-02775-5

[13] H. Muttaqin, Kalamullah Ramli: "Designing An Information Security Framework For The Indonesia Water Industry Sector", en *Cakrawala Repositori IMWI*, vol. 6, n. 3, pp. 771-782, 2023. DOI: 10.52851/cakrawala.v6i3.352

[14] C. Lin, Y.E. Yang, F. Moazeni: "Flood Risks of Cyber-Physical Attacks in a Smart Storm Water System", en *Water Resources Research*, vol. 60, n. 1, pp. e2023WR034827, 2024. DOI: 10.1029/2023WR034827

[15] A.M. Kovács: "Migration Perspectives of Water Sector Cybersecurity", en *Journal of Environmental & Earth Sciences*, vol. 6, n. 3, pp. 217-229, 2024. DOI: 10.30564/jees.v6i3.7020

[16] T. Szádeczky: "Water 4.0 in Hungary: Prospects and Cybersecurity Concerns", en *Acta Polytechnica Hungarica*, vol. 20, n. 7, pp. 211-230, 2023. DOI: 10.12700/APH.20.7.2023.7.12

[17] A.F. Murillo, R. Taormina, N.O. Tippenhauer, S. Galelli: "BATADAL 2.0: A comprehensive dataset for intrusion detection systems in smart water networks", en *Proceedings - 2nd International Joint Conference on Water Distribution System Analysis (WDSA) & Computing and Control in the Water Industry (CCWI)*, pp. 1-10, 2022. DOI: 10.4995/WDSA-CCWI2022.2022.14793

[18] P. Małka, K. Gaska, E. Wysowska, K. Kudlik, J. Ciuła: Ïntelligent infrastructure (critical) of the water supply network for collective water supply systems – a case study", en *Desalination and Water Treatment*, vol. 274, pp. 1-9, 2022. DOI: 10.5004/dwt.2022.28917

[19] I. Józefowicz, H. Michniewicz-Ankiersztajn: "Digital Tools for Water Resource Management as a Part of a Green Economy in Rural Areas", en *Sustainability*, vol. 15, n. 6, pp. 5231, 2023. DOI: 10.3390/su15065231

[20] K.B. Adedeji, A.A. Ponnle, A.M. Abu-Mahfouz, A.M. Kurien: "Towards Digitalization of Water Supply Systems for Sustainable Smart City Development—Water 4.0", en *Applied Sciences*, vol. 12, n. 18, pp. 9174, 2022. DOI: 10.3390/app12189174

The Sponge Mesozoic: The Explosion in Sponge-Based Constructions

Mario Marhuenda Beltrán
Radboud University
The Netherlands
mmarhuenda@cs.ru.nl

***Abstract*—In this work we give an extended summary of recent results ins sponge-based cryptography. Since its introduction in 2007 by Bertoni et al. [1], the sponge construction has become a fundamental building block in modern cryptography, underpinning various cryptographic primitives such as hash functions, stream ciphers, and authenticated encryption schemes. From its inception, a huge literature has been developed around sponges. Particularly in the last two years a plethora of new sponge-based constructions has emerged, that improves both the efficiency and security of previous designs. On the hash function side, there have been three main branches of development: Arithmetization-oriented sponges, mainly oriented towards removing the need for padding, sponges with stronger first preimage, second preimage and collision resistance properties, and the first construction achieving beyond birthday bound security in the indifferentiability framework [2]. On the keyed primitives side, `MacaKey`, the first variable-output length sponge-based PRF with full state throughput has been introduced. Afterwards, the variants `KeyMacaKey` and `pMacaKey` were also proposed. Finally, there has been interesting progress in the leakage resilience setting [3].**

***Index Terms*—sponge, indifferentiability, indistinguishability, leakage resilience, PRF, hash function**

I. Introduction

The sponge construction is a permutation-based cryptographic hash function, that was later adapted as a building block for PRF, MAC, and AE constructions. It was first introduced by Bertoni et al. [1], and it quickly gained in popularity, in particular in light of the SHA-3 competition [4], which was won by the Keccak sponge function [5]. The sponge operates on a b-bit state, which is split into a c-bit inner part, where c is called the "capacity", and an r-bit outer part, where r is called the "rate". On input of a message, the sponge first injectively pads this message and splits it into r-bit chunks. These chunks are then absorbed one by one by adding them to the outer part of the state, where each addition is interleaved with a permutation of the entire state. After the message is absorbed, digests are squeezed r bits at a time by extracting them from the outer part.

a) Work prior to 2024: Under the assumption that the permutation is random, Bertoni et al. [6] proved that the sponge behaves like a random oracle up to around $2^{c/2}$ queries in the indifferentiability framework [7]. Later, Naito and Ohta [8] proved an improved result for a slightly more general setting where the initial message block can be $r + c/2$ bits, and the squeezing is performed with around $r + c/2 - \log_2(c)$ bits at a time.

Soon after, sponges began to be used in the keyed setting, first by prepending the key to the message, then by adding it to the capacity part of the initial value. Ultimately, this led to the full-state keyed sponge (`FSKS`) [9], which absorbs over the entire state. At first sight, this seems insecure, since an adversary could control the entire state by xoring chosen message blocks. However, since the state is unknown to the adversary due to the key addition at the start, the adversary cannot set the state to any value of interest up to the security bound. This construction is proven secure up to $\min\left\{2^c/\mathcal{M}, 2^k\right\}$ permutation calls, where $\mathcal{M}$ denotes the online complexity.

Another example of a keyed sponge can be found in `Ascon-PRF` [10], which improves over the `FSKS` generic security at the cost of a lower rate of absorption. From now on, we will use r and c to denote the rate and capacity during the squeezing phase. The sponge has a sister construction, the duplex [11], which allows interleaving of absorb and squeeze calls. A further example of sponge based constructions is `Kirby` [12], which blinds the state with a feed forward, allowing for fixed length full state output. However, since the output length is fixed to b bits, it requires an additional mode of operation on top to extend their output length.

The analysis of the leakage resilience of sponge-based constructions has been particularly tricky, with only a few results shining through [13], [14].

The above works are only the beginning, the last couple of years have seen a new surge of sponge-based constructions results, that we expose in the remainder of this work.

b) Outline: This *survey* is organized as follows. In Section II, we explore the recent advances in padding-free sponge constructions, which was kick-started by `SAFE`, and continued with `GSponge`, `sponge-pi`, `duplex-pi` and `fks-pi`. Section III presents the first sponge-based construction that achieves indifferentiability beyond the birthday bound. Section IV, explores `MacaKey`, the first sponge-based PRF with full-state throughput, and its variants `KeyMacaKey` and `pMacaKey`. Section V presents the recent improvements in first and second preimage resistance of sponge-based hash functions. Section VI presents the recent improvements on the leakage resilience of the suffix keyed sponge `SuKS`. Finally, Section VII concludes the paper and suggests future research directions.

II. Padding-free sponges

The sponge construction needs an injective padding function to provide security. Padding causes the message to expand, which can cause one extra evaluation of the permutation. In the binary setting, this is not a big issue, since this only happens when the message length is a multiple of the rate. However, in many settings, namely those related to finite

fields, this is more problematic. The reason is that often the padding is implemented field-wise, and these fields can be significantly big, which causes extra permutation evaluations more frequently.

The first sponge construction to remove padding was SAFE [15], which encoded an IO pattern via a hash function into the capacity of the initial value. This has the benefit of allowing to mix the absorb and the squeeze phases, but this can only be done as prescribed by the IO pattern: one must know the length of the message and the length of the output in advance, at least, for the intermediate output values. Nevertheless, in zero-knowledge proof systems, such as Fiat-Shamir related ones and SNARKs, this is often the case. SAFE comes at *no* generic security loss compared to the standard sponge. Another alternative is GSponge [16], which encodes the length of the message into the capacity of the initial value.

Both of these constructions reduce the absorb rate by $c/2$ on the first call, since this space is needed either for the IO pattern encoding, or for the message length encoding. More recently, sponge-pi [17] solved this issue by rescuing a previous idea from Hirose[18]. Namely, if the message is short enough, sponge-pi pads the message. If the padding message causes an extra permutation call, then the capacity of the last squeeze call is permuted by a non-cryptographic permutation, most commonly, by xoring a non-zero constant (c.f. Fig. 1).

III. Permutation based hashing beyond the birthday bound

Suppose you sample q balls, with replacement, out of a bin containing N balls. The birthday problem is the problem of finding the probability that you sampled the same ball twice. The birthday paradox is the surprising fact that this probability becomes significant for $q^2 \approx N$. In cryptographic terms, this has the undesirable result that many constructions that are based in block ciphers of size k bits, can only offer up to $k/2$ bits of security. Analogously, in the permutation based setting, one often keeps only a part of the state secret, the capacity c, which means that the birthday bound limits the security to around $c/2$ bits.

Breaking the birthday bound for permutation-based hash constructions, let alone sponge-based ones, has been an open problem for some time. At last, in 2024, Lefevre and Mennink [2] proposed the first sponge-based construction achieving indifferentiability beyond the birthday bound. The double sponge construction is shown in Fig. 2. This was only possible thanks to improved proof techniques, mainly the chi-squared technique [20], which has been used with success in proving the security of the summation-truncation hybrid [21] and the truncated permutation [22] constructions. Another proof technique that has seen new light is that of multicollisions [19].

The double-sponge provably achieves indifferentiability up to $2^{2c/3}$ permutation calls. Furthermore, it has a very limited throughput of only around $r/2b$ bits.

Finally, the possibility of duplexing double-sponge [23], remains an open problem, and while it seems a duplexed double-sponge would be secure, proving so with current techniques seems to be a challenging. Remarkably, it was only recently that the optimal security of the duplex construction was proven, in a work presented by Degabriele et al. [24] in ASIACRYPT 2023. Previous results contained a reduction to the sponge, which meant that security was only guaranteed if every block was independently padded, which added unnecessary overhead.

IV. MacaKey, pMacaKey and KeyMacaKey: sponge-based PRFs with full-state throughput

We now move to the PRF setting. The full state keyed sponge (FSKS) [9] is the first sponge based construction allowing absorption over the entire state, instead of just the rate part of the state. This leverages the fact that thanks to the key being added at the start, the state is secret, and thus it is not possible for an adversary to overwrite the state to anything useful, even though it can xor over the whole state. This is not true for the squeezing phase, since the rate part of the state is revealed as part of the output.

Recently, a solution was proposed by Beltrán and Lefevre, which uses the last absorb state to blind the output, allowing for full-state squeeze [19]. The original construction contained a subtle oversight, as pointed out by Chakraborty and Dhar [25], which also proposed two fixed variants of MacaKey: pMacaKey and KeyMacaKey. However, the original MacaKey construction can be easily fixed by using adding domain separation, this achieves a higher security bound than pMacaKey, and requires less memory than KeyMacaKey (c.f. Section IV). Furthermore, KeyMacaKey requires storing the key in memory for longer, which is not ideal since mitigation of side-channel attacks becomes more expensive. Withal, prefix-free padding fixes MacaKey, while improving its security bound, and uniquely at almost no overhead. A description of MacaKey can be found in Fig. 3.

V. Improvement in preimage and collision resistance of sponges

Lefevre [26] proved that the sponge construction has tight first preimage resistance of $\min\{n, \max\{n-r, c/2\}\}$. Furthermore, second preimage resistance of the sponge construction is well known to be $\min\{n, c/2\}$. In 2025, a twoo way race, between Guo et al. [27], Sun et al. [28], began to design sponge-based constructions with improved second preimage and collision resistance. In short, despite these works presenting slightly different constructions, they all achieve the same bound: $\min n$ for first preimage and almost $\min\{n, c\}$ second preimage security. This "*almost*" is due to the so-called "long message" attack. Currently there is no way to prevent such attack.

VI. Leakage resilience of SuKS

The suffix keyed sponge or SuKS for short [3] is a hash-then-encrypt MAC based on the sponge construction. It first processes the message in the sponge fashion, then it adds the key and tweaks the state slightly before doing a final permutation call. Finally, the tag is computed from a truncation of this final state.

In general, the study of leakage resilient of permutation based constructions is extremely tricky and subtle, partly because side-channel attacks are hard to model. This leads to models that are often generous to the attacker, resulting in loose bounds. Two prominent models are position leakage and Hamming weight leakage, where Hamming weight denotes

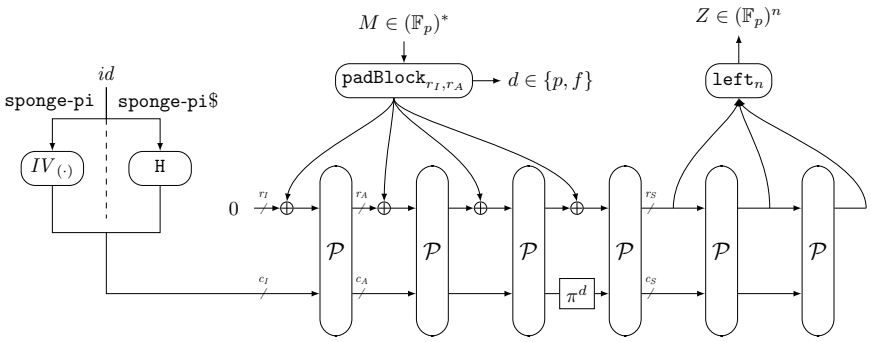


Fig. 1: The sponge-pi and sponge-pi$ constructions

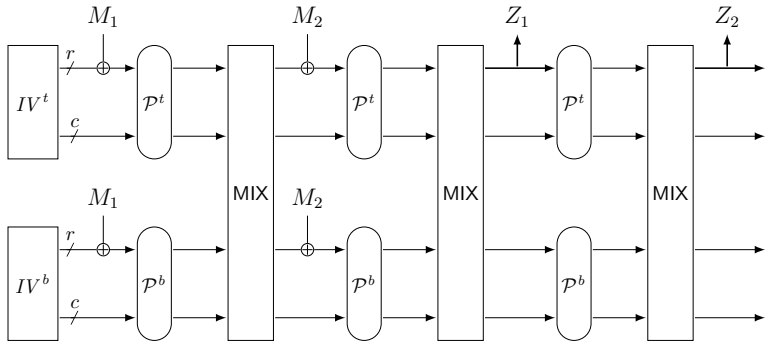


Fig. 2: The double sponge construction [2]. For simplicity we assume the message $M = M_1 \| M_2$ is already padded.

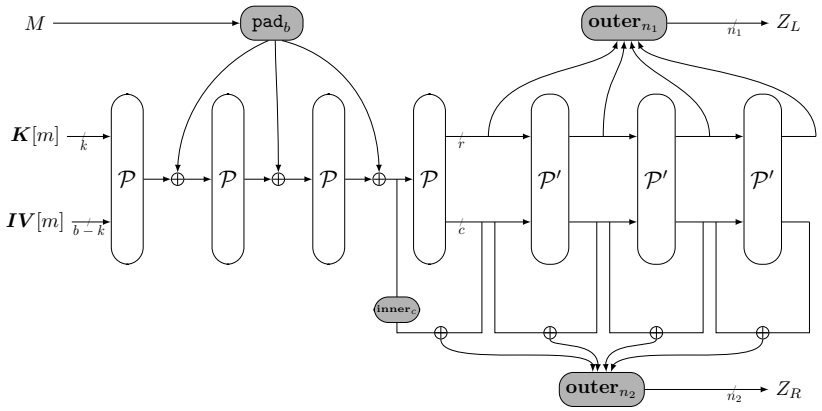


Fig. 3: The MacaKey [19] construction. The output string is $Z_L \| Z_R$. And $\mathtt{pad}_b^{\mathrm{PF}}$ is a prefix-free padding function.

Construction	Security bound (big O)	Memory Cost	Throughput
FSKS	$\frac{M}{2^c}$	b	$\frac{r}{b}$
MacaKey	$\frac{MN}{2^b} + \frac{N}{2^c}$	$b + r$	1
pMacaKey	$\frac{M}{2^c}$	$b + r$	1
KeyMacaKey	$\frac{MN}{2^b} + \frac{N}{2^c}$	$b + r + k$	1

Fig. 4: Comparison of the different full state sponges. Here, $\mathcal{M}$ the online complexity, $\mathcal{N}$ is the offline complexity, and k is the key size.

the number of non-zero bits. In the first model, the adversary learns λ bits of the state, while in the second, the adversary learns the Hamming weight of a particular state. The first one is overly generous, since this is usually not achievable via side-channel. The second one is more realistic, but the distribution of Hamming weights does not follow a uniform distribution, which makes the analysis significantly harder. Provable security tools are oriented towards distinguishing from uniformly random, rather than the binomial distribution of Hamming weights, hence why this paper poses a landmark in this topic.

VII. Conclusion

We have given an overview of the most recent advances in sponge-based constructions. Undoubtedly, sponges have become a fundamental building block in modern cryptography, and were popular since their introduction. Nevertheless, the last couple years have seen a veritable surge of new constructions.

Still, some areas remain largely unexplored. One area which has not yet received proper attention is that of formal verification of the proof of security of cryptographic constructions. In the area of sponges, only SHA-3 has been formally verified [29]. Remarkably, not even the modifications made by Naito and Ohta have been formally verified. This is especially important since these proofs are often complex, error-prone and the errors subtle, as we show in Section IV. As a side effect, they are hard and time-consuming to verify by reviewers. It is not uncommon for mistakes to go unnoticed, only to be found after publication, one compilation of such errors is [30], a survey by Koblitz and Menezes on the many faulty papers that had "correct" proofs.

Another possible research direction is the design of permutation based accordion modes, as current candidates are mostly block cipher based. This is of particular interest due to NIST's recent interest in such modes.

Acknowledgments. The author is supported by the Netherlands Organisation for Scientific Research (NWO) under grant VI.Vidi.203.099. The author would like to thank Bart Mennink and Charlotte Lefevre for the fruitful discussions and their overall guidance. Furthemore, the author would like to thank the reviewers for their insightful comments and suggestions.

References

[1] G. Bertoni, J. Daemen, M. Peeters, and G. V. Assche, "On the indifferentiability of the sponge construction," ser. Lecture Notes in Computer Science, N. P. Smart, Ed., vol. 4965, 2008, pp. 181–197.

[2] C. Lefevre and B. Mennink, "Permutation-based hashing beyond the birthday bound," *IACR Trans. Symmetric Cryptol.*, vol. 2024, no. 1, pp. 71–113, 2024. [Online]. Available: https://doi.org/10.46586/tosc.v2024.i1.71-113

[3] H. Berendsen and B. Mennink, "Tightening leakage resilience of the suffix keyed sponge," *IACR Trans. Symmetric Cryptol.*, vol. 2024, no. 1, pp. 459–496, 2024. [Online]. Available: https://doi.org/10.46586/tosc.v2024.i1.459-496

[4] NIST, "SHA-3 Competition," in *International Conference on the Theory and Application of Cryptographic Techniques*, 2007-2012.

[5] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Keccak," in *International Conference on the Theory and Application of Cryptographic Techniques*, 2013.

[6] ——, "On the Indifferentiability of the Sponge Construction," ser. Lecture Notes in Computer Science, vol. 4965. Springer, 2008, pp. 181–197.

[7] U. M. Maurer, R. Renner, and C. Holenstein, "Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology," M. Naor, Ed., vol. 2951, 2004, pp. 21–39.

[8] Y. Naito and K. Ohta, "Improved Indifferentiable Security Analysis of PHOTON," M. Abdalla and R. D. Prisco, Eds., vol. 8642, 2014, pp. 340–357.

[9] B. Mennink, R. Reyhanitabar, and D. Vizár, "Security of Full-State Keyed Sponge and Duplex: Applications to Authenticated Encryption."

[10] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, "Ascon MAC, PRF, and Short-Input PRF - Lightweight, Fast, and Efficient Pseudorandom Functions," ser. Lecture Notes in Computer Science, vol. 14643. Springer, 2024, pp. 381–403.

[11] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Duplexing the sponge: Single-pass authenticated encryption and other applications," ser. Lecture Notes in Computer Science, A. Miri and S. Vaudenay, Eds., vol. 7118, 2011, pp. 320–337.

[12] C. Lefevre, Y. Belkheyar, and J. Daemen, "Kirby: A Robust Permutation-Based PRF Construction," *IACR Cryptol. ePrint Arch.*, p. 1520, 2023. [Online]. Available: https://eprint.iacr.org/2023/1520

[13] C. Dobraunig and B. Mennink, "Leakage Resilience of the Duplex Construction," ser. Lecture Notes in Computer Science, vol. 11923, 2019, pp. 225–255.

[14] C. Dobraunig, B. Mennink, and R. Primas, "Leakage and tamper resilient permutation-based cryptography," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS 2022, Los Angeles, CA, USA, November 7-11, 2022*, H. Yin, A. Stavrou, C. Cremers, and E. Shi, Eds. ACM, 2022, pp. 859–873. [Online]. Available: https://doi.org/10.1145/3548606.3560635

[15] D. Khovratovich, M. M. Beltrán, and B. Mennink, "Generic security of the SAFE API and its applications," ser. Lecture Notes in Computer Science, vol. 14445, 2023, pp. 301–327.

[16] T. Ashur and A. S. Bhati, "Generalized indifferentiable sponge and its application to polygon miden VM," Cryptology ePrint Archive, Paper 2024/911, 2024. [Online]. Available: https://eprint.iacr.org/2024/911

[17] C. Lefevre, M. M. Beltrán, and B. Mennink, "To pad or not to pad? padding-free arithmetization-oriented sponges," *IACR Trans. Symmetric Cryptol.*, vol. 2025, no. 1, pp. 97–137, 2025. [Online]. Available: https://doi.org/10.46586/tosc.v2025.i1.97-137

[18] S. Hirose, "Sequential Hashing with Minimum Padding," *Cryptogr.*, vol. 2, no. 2, p. 11, 2018. [Online]. Available: https://doi.org/10.3390/cryptography2020011

[19] C. Lefevre and M. M. Beltrán, "Macakey: Full-state keyed sponge meets the summation-truncation hybrid," *IACR Cryptol. ePrint Arch.*, p. 893, 2025. [Online]. Available: https://eprint.iacr.org/2025/893

[20] W. Dai, V. T. Hoang, and S. Tessaro, "Information-theoretic indistinguishability via the chi-squared method."

[21] A. Gunsing and B. Mennink, "The Summation-Truncation Hybrid: Reusing Discarded Bits for Free."

[22] W. Choi, B. Lee, and J. Lee, "Indifferentiability of Truncated Random Permutations," ser. Lecture Notes in Computer Science, vol. 11921. Springer, 2019, pp. 175–195.

[23] B. Mennink, "Understanding the Duplex and Its Security," *IACR Transactions on Symmetric Cryptology*, vol. 2023, no. 2, p. 1–46, Jun. 2023. [Online]. Available: https://tosc.iacr.org/index.php/ToSC/article/view/10976

[24] J. P. Degabriele, M. Fischlin, and J. Govinden, "The indifferentiability of the duplex and its practical applications."

[25] B. Chakraborty and C. Dhar, "Breaking and fixing MacaKey," Cryptology ePrint Archive, Paper 2025/2038, 2025. [Online]. Available: https://eprint.iacr.org/2025/2038

[26] C. Lefevre and B. Mennink, "Tight Preimage Resistance of the Sponge Construction," Y. Dodis and T. Shrimpton, Eds.

[27] C. Guo, K. Hu, Y. Fan, Y. Fu, and M. Wang, "Adding feeding forward back to the sponge construction," Cryptology ePrint Archive, Paper 2025/1006, 2025. [Online]. Available: https://eprint.iacr.org/2025/1006

[28] S. Sun, S. Li, Z. Zhang, C. Lefevre, B. Mennink, Z. Qin, and D. Feng, "Permutation-based hashing with stronger (second) preimage resistance," Cryptology ePrint Archive, Paper 2025/963, 2025. [Online]. Available: https://eprint.iacr.org/2025/963

[29] J. B. Almeida, C. Baritel-Ruet, M. Barbosa, G. Barthe, F. Dupressoir, B. Grégoire, V. Laporte, T. Oliveira, A. Stoughton, and P. Strub, "Machine-checked proofs for cryptographic standards: Indifferentiability of sponge and secure high-assurance implementations of SHA-3."

[30] N. Koblitz and A. Menezes, "Critical perspectives on provable security: Fifteen years of "another look" papers," Cryptology ePrint Archive, Paper 2019/1336, 2019. [Online]. Available: https://eprint.iacr.org/2019/1336

Privacidad online: evaluando la incertidumbre del adversario en técnicas de perturbación de perfiles

César Gil
Department of Network Engineering
Universitat Politècnica de Catalunya
Barcelona, Spain
cesar.gil@upc.edu

Javier Parra-Arnau
Department of Network Engineering
Universitat Politècnica de Catalunya
Barcelona, Spain
javier.parra@upc.edu

Jordi Forné
Department of Network Engineering
Universitat Politècnica de Catalunya
Barcelona, Spain
jordi.forne@upc.edu

***Resumen*—En este trabajo estudiamos las vulnerabilidades de las técnicas de protección de perfiles, cuyo principal ámbito de aplicación general son los sistemas de información personalizados. Desarrolladas y validadas desde la primera década del milenio hasta el presente, estas técnicas implementan mecanismos que simulan la ofuscación de los perfiles de los usuarios, entendidos como histogramas normalizados que recogen sus intereses sobre categorías finitas y predefinidas, y que son moneda de cambio de servicios personalizados para los que se elaboran perfiles con una precisión que puede poner en compromiso la privacidad. Abordamos el reto a partir del ejemplo paradigmático en cuanto a estrategia de perturbación, la falsificación, en su versión optimizada desde el punto de vista del compromiso entre privacidad y utilidad. Nuestra principal contribución radica en caracterizar la incertidumbre que enfrenta un adversario al estimar el perfil real del usuario a partir del perfil perturbado. Para ello definimos diversos modelos de adversario y una metodología de evaluación de esta incertidumbre, y presentamos resultados cuantitativos para la falsificación óptima de consultas. Los resultados obtenidos muestran que el mecanismo analizado es vulnerable a estos ataques y requiere su revisión.**

***Index Terms*—Sistemas de información personalizados, tecnologías de mejora de la privacidad, perturbación de datos, perfilado de usuarios, entropía, incertidumbre.**

I. Introducción

El feedback continuo entre los usuarios y los sistemas de información personalizados (PISs), como Google, Amazon o Flickr, supone un potencial riesgo para la privacidad en cada transacción. Imaginemos un sencillo sistema de consulta web en que un usuario envía preguntas al servidor y recupera una determinada respuesta en función de su interés. Pasado el tiempo, si el usuario no ha protegido sus consultas, cualquier tercero puede conocer sus intereses, por ejemplo, en la forma de un histograma discreto perfilado con absoluta precisión. Alternativamente, el usuario puede usar algún mecanismo de protección de la privacidad, con el objetivo de entorpecer la mirada de terceras partes no autorizadas. Este sencillo ejemplo es el marco donde, de forma tan compleja y sofisticada como podamos pensar, se juega una millonaria batalla por conocer los datos personales de los usuarios. No en vano, a principios del milenio se acuña el término *economía digital* [1], situando los datos en el centro del proceso productivo, altamente automatizado y orientado a productos y servicios cada vez más personalizados. Sin embargo, casos como el de Facebook y Cambridge Analytica [2] demuestran que el interés por el perfilado de usuarios de Internet puede trascender los fines estrictamente comerciales.

En el ámbito de los PISs, es importante resaltar la naturaleza de esta relación transaccional: la personalización [1] y el perfilado de usuarios son dos caras de la misma moneda, es decir, sin perfilado no hay personalización [3], [4]. Y sabemos que la elaboración de perfiles [2] constituye el riesgo de privacidad fundamental de los PISs, con dos objetivos exclusivos para un atacante de la privacidad: la identificación y la clasificación.

Esta dualidad revela que las transacciones entre usuarios y sistemas deben entenderse como un compromiso entre la privacidad y la utilidad de los datos privados que el usuario comparte. Además, integrar este enfoque puede proporcionar una ventaja competitiva en el diseño de sistemas personalizados. No es casual que en las últimas décadas, el interés por las tecnologías de mejora de la privacidad (PETs) haya impulsado una fértil producción de aplicaciones y literatura especializada.

Este trabajo se centra en las técnicas perturbativas de protección de perfiles que optimizan el compromiso entre la privacidad del usuario y la utilidad de sus datos. Estas técnicas operan en un entorno de privacidad local (user-side) bajo un modelo de cero confianza, en que cualquier tercero, incluido el propio sistema, se considera un potencial atacante. Su diseño se enmarca en la *privacidad dura*, guiada por el principio de *minimización de datos*, según el cual los datos personales no deben divulgarse innecesariamente. El objetivo es, por tanto, reducir al mínimo la información que el usuario comparte sin comprometer la funcionalidad del sistema.

Nuestra principal contribución consiste en el abordaje de las vulnerabilidades de estas técnicas. Estudiamos la técnica más representativa, la falsificación en su versión optimizada, y cuantificamos la incertidumbre que enfrenta un adversario, previamente caracterizado, al estimar el perfil real de un usuario a partir de su versión ofuscada. Como resultado, cuestionamos la adecuación del modelo de compromiso y de las métricas de privacidad implementadas. Este estudio nos sitúa ante un nuevo problema abierto que representa al mismo tiempo una novedad, un reto y una oportunidad en aras de la mejora de los mecanismos revisados.

[1] Usamos de manera intercambiable los términos utilidad y personalización.

[2] Art. 4 RGPD 2016/679. Definiciones. *4) ≪elaboración de perfiles≫: toda forma de tratamiento automatizado de datos personales consistente en utilizar datos personales para evaluar determinados aspectos personales de una persona física, en particular para analizar o predecir aspectos relativos al rendimiento profesional, situación económica, salud, preferencias personales, intereses, fiabilidad, comportamiento, ubicación o movimientos de dicha persona física.*

El documento se organiza de la siguiente forma. Tras la introducción, describimos el estado del arte relativo a las DPTs optimizadas, con especial detalle sobre la falsificación de datos óptima. A continuación, analizamos y cuantificamos la incertidumbre que diferentes tipos de adversarios predefinidos enfrentan al estimar el perfil real de un usuario, para seguir con una discusión sobre los resultados obtenidos previamente mediante un sencillo ejemplo ilustrativo. Finalizamos este trabajo con una serie de conclusiones y proponemos una primera intervención sobre el mecanismo analizado con el propósito de completar y validar la mejora de las técnicas evaluadas.

II. Estado del arte

En el amplio contexto de los PISs, existen innumerables tecnologías de mejora de la privacidad (PETs). Una posible clasificación establece cinco categorías básicas [4], a saber, (i) *tecnologías antiseguimiento básicas*, (ii) *enfoques basados en terceras partes de confianza*, (iii) *mecanismos colaborativos*, (iv) *métodos criptográficos para recuperación de información privada* y (v) *técnicas de perturbación de datos (DPTs)*. Es importante destacar que muchas de estas tecnologías pueden combinarse sinérgicamente.

Como alternativa al resto de tecnologías, las DPTs tienen por objetivo dificultar el perfilado preciso de los usuarios ofuscando la información que comparten explícita o implícitamente con un PIS. Por contra, su impacto en costes sobre la funcionalidad del sistema y la utilidad de los datos, dejan patente un compromiso entre estos aspectos y la protección de la privacidad que, sin embargo, ha sido escasamente estudiado [5] en el caso de la protección local del perfil.

En este trabajo nos centramos en las DPTs dedicadas a la protección local de perfiles y enfocadas en optimizar el compromiso entre privacidad y utilidad, para fijarnos especialmente en la más paradigmática, la falsificación de datos, pero en su versión optimizada.

II-A. DPTs optimizadas

Para su objetivo de compromiso óptimo, las técnicas perturbativas de protección de perfiles optimizadas se sirven de métricas procedentes principalmente del campo de la teoría de la información [6] en conjunción con la resolución de problemas de optimización convexa [7]. Inspiradas en el campo del control de divulgación estadística, actualmente disponemos de un completo abanico de estas técnicas, a saber, falsificación, supresión, la combinación de ambas y generalización, que detallamos a continuación. Cabe destacar que se trata de *perturbaciones deterministas*, en contraposición con aquellas técnicas que se basan en perturbaciones aleatorias.

Falsificación. La falsificación reúne técnicas que mezclan datos genuinos con datos falsos para distorsionar el perfil de intereses del usuario. Una de las primeras propuestas optimizadas aparece en la búsqueda web privada (PWS). En [8], el inyector de consultas NISSP minimiza la filtración de privacidad mediante información mutua entre consultas reales y falsas generadas localmente.

La principal desventaja de la falsificación es la sobrecarga de tráfico, abordada en [9] con un modelo que optimiza el equilibrio entre la tasa de consultas falsas y el riesgo de privacidad, medido mediante divergencia de Kullback-Leibler (KL) [6]. Este enfoque ha sido posteriormente validado con datos reales en sistemas de recomendación de recursos (RRSs) basados en contenido [10] y en el tráfico DNS de aplicaciones móviles [11].

Supresión. La supresión agrupa técnicas que reducen actividad en el perfil del usuario, en contraste con la falsificación. Igual que su contraria, la simplicidad y bajo coste de implementación han permitido validarla en distintos escenarios. En [12], aplicada a la web semántica y a los sistemas de etiquetado, se analiza el equilibrio óptimo entre privacidad y utilidad usando, respectivamente, la entropía de Shannon del perfil perturbado y la proporción de etiquetas que el usuario elimina. Una perspectiva especialmente relevante para las DPTs es su aplicación en RRSs bajo contextos exigentes como el control parental, tal como muestra [13], donde la evaluación considera tanto la degradación de datos como la precisión de las políticas de filtrado.

Falsificación y supresión combinadas. La combinación de falsificación y supresión combinadas representa la versatilidad de estas técnicas para complementarse entre sí. En [12] se presenta una solución cerrada al problema de la falsificación y supresión óptimas y simultáneas aplicada en RRSs y validada con datos reales, permitiendo a los usuarios enviar calificaciones falsas y/o retener calificaciones de los elementos de interés. El estudio demuestra que el uso combinado de ambas estrategias puede ser más eficiente que aplicarlas por separado.

Generalización. La generalización de datos es la técnica perturbativa más novedosa y su estrategia se basa en la organización de las categorías de intereses de los usuarios según una taxonomía jerárquica. Propuesta y evaluada con datos reales en [14], demostró eficacia en sistemas que usan taxonomías semánticas y en servicios basados en localización, donde las coordenadas de los puntos de interés se agrupan jerárquicamente en áreas de interés convenientemente particionadas. Como en propuestas previas, el problema de optimización convexa que modela la compensación entre privacidad y utilidad permite deducir una tasa crítica de perturbación a partir de la cual la privacidad deja de mejorar.

II-B. Falsificación de datos óptima (OQF)

La propuesta de [9] presenta un mecanismo de falsificación óptima de consultas (OQF) para la protección de perfiles cuyo objetivo es proporcionar una versión distorsionada del perfil de intereses de un usuario. Inicialmente aplicado al escenario de la PWS, OQF comparte elementos comunes con la mayoría de DPTs óptimas revisadas en la *Sección II-A*. En esta sección hacemos un breve recorrido por todos ellos.

OQF se formula sobre perfiles, es decir, asumiendo los datos que un usuario comparte con un PIS (i.e., consultas, etiquetas, geolocalizaciones, etc.) como una sucesión de variables aleatorias i.i.d. $\{X_k\}$, $k \in \mathbb{Z}^+$, que toman valores en un alfabeto finito y común de categorías de interés $\mathscr{X} = \{1, \ldots, n\}$, $n \in \mathbb{Z}^+, n \geq 2$. De acuerdo con este modelo, un *perfil de usuario* es la función de masa de probabilidad según la que se distribuye $\{X_k\}$, una representación ampliamente aceptada en la literatura sobre privacidad y seguridad en el contexto de los PISs [9], [15], [16], [17], [18] que puede aplicarse a escenarios donde los datos privados del usuario no

sean específicamente consultas. Conceptualmente, interpretamos un perfil como un histograma de frecuencias relativas de los datos privados del usuario dentro del conjunto predefinido, discreto y finito de categorías de interés.

En general, un usuario decidirá compartir de forma controlada una parte de su perfil a cambio de servicios personalizados. Dado el caso, su perfil aparente, observado por terceros, se parecerá en mayor o menor medida a sus intereses reales. En el caso de OQF, la estrategia de falsificación se explicita en el perfil aparente del usuario t como resultado de una combinación lineal convexa equivalente a mezclar el perfil real q con el perfil falso r en una proporción $\rho \in [0, 1]$, a la que nos referimos como la *tasa de falsificación* o *redundancia*.

$$t = (1 - \rho)q + \rho r. \tag{1}$$

Es notable que ρ determina el grado de distorsión del perfil aparente, pasando de datos totalmente genuinos con $\rho = 0$, a datos totalmente falsos, con $\rho = 1$.Por tanto, dado un valor ρ, la clave de este mecanismo consiste en determinar apropiadamente el perfil falso r.

Las métricas empleadas por OQF para caracterizar la privacidad de un usuario miden la discrepancia entre el perfil aparente t y el perfil objetivo o de referencia p del usuario, entendido como aquel perfil al que desea parecerse. Contra la amenaza de individuación de un usuario, QQF implementa la divergencia KL [6], también denominada *entropía relativa*, como medida del riesgo de privacidad $\mathcal{R}$ de un usuario y, en consecuencia, se define la función de riesgo de privacidad de un usuario $\mathcal{R}(\rho)$ para caracterizar el compromiso óptimo entre privacidad (riesgo) y redundancia.

$$\mathcal{R}(\rho) = \min_{r \in \Delta_n} \mathrm{D}((1 - \rho)q + \rho r || p). \tag{2}$$

En este caso, la utilidad se interpreta directamente como la redundancia ρ, que cuantifica la sobrecarga de tráfico y la degradación de los datos, equivalente a la consiguiente pérdida de personalización.

Se demuestra que la función $\mathcal{R}(\rho)$ es decreciente y convexa y que existe una redundancia crítica ρ_{crit} a partir de la cual se alcanza la privacidad perfecta, esto es, $\mathcal{R}(\rho) = 0$. Este parámetro solo depende del perfil real q y de referencia p del usuario.

$$\rho_{crit} = 1 - \frac{p_n}{q_n}. \tag{3}$$

Por otra parte, la solución óptima de OQF para el perfil de datos falsos r^* resulta de resolver el problema convexo formulado para el compromiso en la *Ec. 2* dado un valor ρ, y que no es otro que una asignación óptima de recursos [7], donde el perfil falso r es la variable de decisión. La solución óptima explícita, de tipo *"water-filling"* y que por tanto es cerrada aunque por tramos, permite analizar comportamientos relevantes como casos de baja redundancia o bajo riesgo de privacidad.

Para acercar OQF al lector tomamos un sencillo ejemplo exclusivamente ilustrativo obtenido por instanciación directa del modelo y de expresiones derivadas, con $n = 3$ categorías, en que $q = (1/6, 1/3, 1/2)$ y $p = (5/12, 1/3, 1/4)$. En la *Fig. 1* ilustramos el comportamiento del mecanismo de falsificación

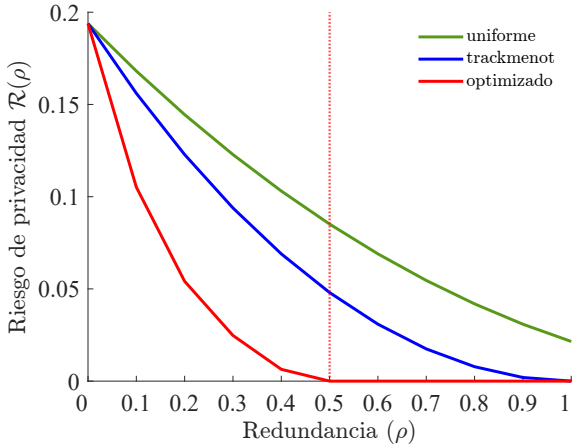


Figura 1. Comportamiento del mecanismo de falsificación para las tres opciones de valoración del perfil de datos falsos r: uniforme, TrackMeNot y optimizado.

mediante la representación del riesgo de privacidad $\mathcal{R}(\rho)$ para tres formas de generación del perfil de datos falsos r: el perfil uniforme u, el perfil promedio del conjunto de usuarios, que denotamos como $\bar{q}$, como aproximación al comportamiento de la conocida PET denominada TrackMeNot (TMN) [19], y el perfil óptimo de OQF. Resulta evidente observar que el mecanismo optimizado siempre ofrece un riesgo igual o menor que las dos opciones sub-optimas. Además, la línea vertical roja nos informa del valor crítico de la redundancia del mecanismo optimizado, $\rho_{crit} = 0{,}5$. Con este valor sabemos que añadir más del 50 % de datos falsos no mejora la privacidad del usuario. A partir este punto, el perfil aparente t coincide con perfil objetivo p y la solución óptima r^* carece de componentes nulas.

II-C. Limitaciones y vulnerabilidades de OQF

A la par de sus beneficios, las DPTs presentan limitaciones y OQF no es una excepción. Por ejemplo, el modelo de usuario basado en perfiles se limita a las frecuencias relativas, relevantes frente a ataques basados en el contenido pero no aborda las diferencias en las frecuencias absolutas, que podrían ser explotadas en el análisis de tráfico.

En [20], los autores presentan un marco analítico sobre el que evalúan seis DPTs sub-óptimas y óptimas, conocidas por sus siglas, PRAW [21], [22], [23], [24], GooPIR [25], TMN [19], PDS [26], NISPP [8] y OQF, y destacan vulnerabilidades que implican la sobreestimación del nivel de privacidad ofrecido, aunque sin presentar resultados experimentales.

En el caso de OQF, la vulnerabilidad vendría dada por la limitada diversidad del espacio de estimaciones en que un adversario buscaría el valor del perfil real q del usuario, y que a su vez tendría su principal origen en la reversibilidad del mecanismo de falsificación y en el determinismo y la forma de tipo *"water-filling"* de la solución óptima r^*. Estas circunstancias podrían ser aprovechadas en mayor o menor grado por un adversario de la privacidad para perfilar eficientemente al usuario.

Dedicamos la siguiente sección a estudiar la incertidumbre que afronta un adversario de OQF al estimar el perfil real del usuario y que supone la principal contribución de nuestra propuesta.

III. Incertidumbre del adversario

Nuestra propuesta se centra en estudiar el desempeño de OQF desde el punto de la incertidumbre que enfrenta un adversario al estimar el perfil real de un usuario, una vez ofuscado y compartido con el sistema en forma de perfil aparente. Se trata de un enfoque novedoso y un reto para la mejora de las técnicas de perturbación de datos que constituye nuestra principal contribución.

La robustez de un mecanismo de privacidad frente a potenciales brechas de información depende de qué asunciones hacemos sobre un adversario y su conocimiento del mecanismo. Estas asunciones constituyen lo que conocemos por modelo de adversario.

En general, un adversario de OQF desea obtener la mejor estimación $\hat{q}$ posible del perfil real q a partir de la observación del perfil aparente t. Asumiendo que un adversario conoce el mecanismo de falsificación y su funcionamiento, la estimación explícita del perfil real se obtiene directamente por inversión del perfil aparente.

$$\hat{q} = \frac{t^* - \rho \hat{r^*}}{1 - \hat{\rho}}, \tag{4}$$

donde t^* es el perfil aparente que observa el adversario, mientras que $\hat{\rho}$ y $\hat{r^*}$ son las estimaciones del adversario sobre la redundancia ρ y la distribución óptima del perfil falso del usuario r^*, respectivamente. Esta representación nos permite empezar a hipotetizar sobre la pericia del adversario para estimar el perfil real del usuario q.

III-A. Tipos de adversarios

En función del conocimiento sobre la forma de la solución óptima r^*, que representamos como el espacio de valores $\hat{R}$, $\hat{R} \subseteq \mathcal{B}^n$, donde $\mathcal{B}^n$ es la base canónica de $\mathbb{R}^n$, definimos tres tipos de adversarios de OQF:

1. `naive`, el *adversario ingenuo* que desconoce completamente la forma de la solución óptima y, por tanto, r^* puede ser cualquier valor del simplex unidad ($\hat{R} \equiv \mathcal{B}^n$).
2. `rogue`, el *adversario curioso* que conoce el valor del perfil de referencia p y por tanto puede saber que si $t \neq p$, la última componente de la solución óptima es nula ($\hat{R} \equiv \mathcal{B}^{n-1}$).
3. `smart`, el *adversario astuto* que conoce la forma de la solución óptima en todo momento, esto es, el número de componentes nulas i, $1 \leq i \leq n$ ($\hat{R} \equiv \mathcal{B}^i$).

Además, para cada tipo de adversario definimos su precisión respecto de la redundancia ρ:

a) Si $\hat{\rho} = \rho$ entonces el adversario es ρ-preciso.
b) En caso contrario, $\hat{\rho} \in [\text{máx}\{0, \rho - \epsilon\}, \text{mín}\{1, \rho + \epsilon\}]$ con $\epsilon > 0$ y el adversario es ϵ-preciso.

Naturalmente, un adversario ρ-preciso es superior al mismo adversario ϵ-preciso. En total, teniendo en cuenta estas definiciones, caracterizamos seis tipos de adversarios diferentes, con mayor o menor pericia para atacar OQF.

III-B. Incertidumbre

Caracterizados los adversarios de OQF, consideramos el espacio de estimación $\hat{Q}$ de un adversario de OQF sobre el perfil real del usuario q como una región geométrica de valores dentro del cual se espera encontrar el perfil real q y resulta de intersecar la región factible de la estimación $\hat{q}$ y el simplex unidad, representado por la base canónica $\mathcal{B}^n$ a la derecha de la ecuación,

$$\hat{Q} = \frac{t^* - \hat{\rho}\hat{R}}{1 - \hat{\rho}} \cap \mathcal{B}^n. \tag{5}$$

En este punto ya podemos intuir que la incertidumbre que encierra esta estimación constituirá otro criterio de calidad para el mecanismo, añadido a la privacidad y la utilidad de los datos.

Definición 1 (Incertidumbre del adversario): Dado un valor $\rho \in [0, \rho_{crit}]$, la incertidumbre I de un adversario sobre la estimación del perfil real del usuario q es una métrica que equivale al área del espacio de estimación $\hat{Q}$, es decir, $I_{\hat{Q}} = Area(\hat{Q})$. □

III-C. Resultados

Equipados con esta métrica, pasamos a evaluar numéricamente la incertidumbre que presentan los adversarios que hemos definido. Partimos del mismo ejemplo empleado en la *Sección II-B*, analizando el intervalo $\rho \in [0, \rho_{crit}]$. En la *Fig. 2* ilustramos en función de la redundancia ρ la incertidumbre $I_{\hat{Q}}$ de un adversario `naive`, y que por tanto desconoce por completo la forma de la solución óptima de OQF. Mostramos además en *Fig. 3 (a)* y *(b)* la variación que se produce sobre la incertidumbre cuando este adversario tiene menor o mayor precisión sobre la redundancia, ϵ-preciso y ρ-preciso, respectivamente. En general, en sus dos versiones, este es el adversario con menor peligro para OQF.

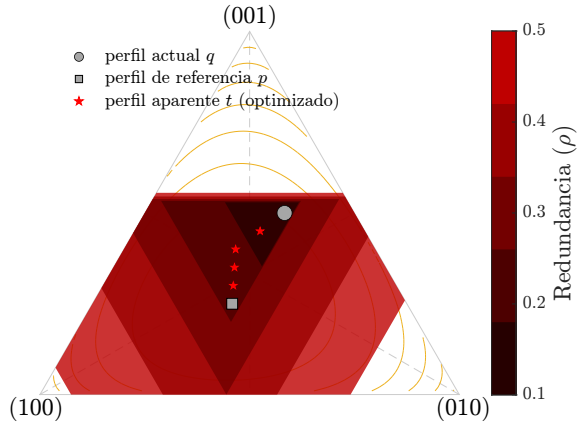
Siguiendo con la senda de la peligrosidad, en la *Fig. 3* ilustramos en función de la redundancia ρ la incertidumbre $I_{\hat{Q}}$ de los adversarios `rogue` y `smart` ϵ-precisos, con $\epsilon = 0{,}05$. Se trata de adversarios con un menor o mayor grado de información sobre la forma de la solución óptima de OQF. Ambos adversarios resultan muy similares pero claramente resultan más peligrosos que los anteriores.

Por último, nos encontramos a los homólogos ρ-precisos de los dos adversarios anteriores, de los que no ofrecemos ilustración alguna dado que su incertidumbre $I_{\hat{Q}}$ es nula para cualquier valor de ρ. Claramente representan la mayor amenaza a la privacidad que ofrece OQF. En la *Fig. 4* resumimos la incertidumbre $I_{\hat{Q}}$ de los seis adversarios definidos, con valores entre el 0 % y 31.60 %, donde claramente podemos observar y comparar la potencia de la amenaza que representan para OQF.

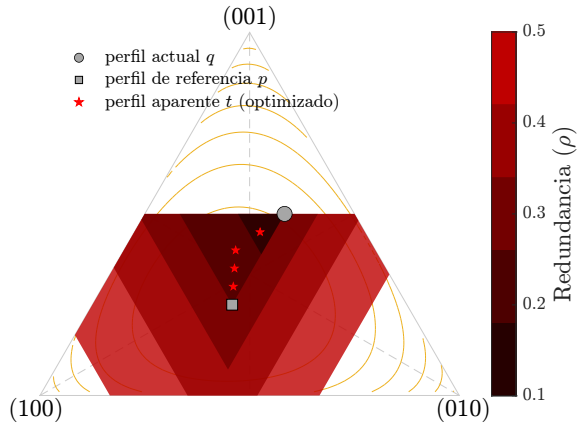
IV. Discusión

Los resultados obtenidos sobre el ejemplo sencillo que presentamos nos indican que a pesar de las bondades de OQF, el criterio de privacidad empleado para evaluar el compromiso óptimo frente a la utilidad podría no resultar adecuado. Dado el caso, el mecanismo resulta vulnerable en su objetivo fundamental de protección de la privacidad del usuario frente a adversarios capaces que pretenden conocer su perfil con precisión.

No obstante, podemos clasificar los tipos de adversarios según su conocimiento sobre el mecanismo y obtener una medida que por el momento nos permite evaluar numéricamente la incertidumbre de su estimación sobre el perfil real del usuario. Dejando aparte la tasa de falsificación ρ, hemos

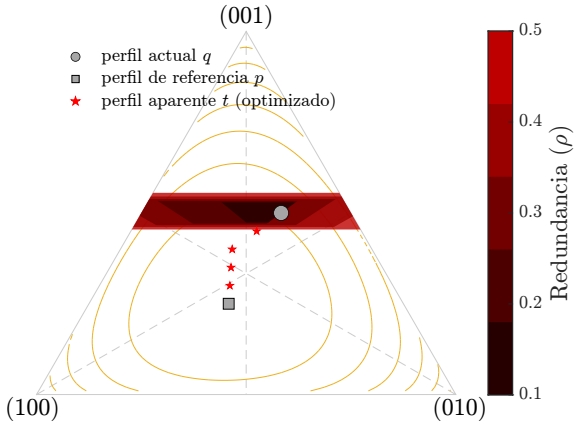


(a) adversario `naive` ϵ-preciso, $\epsilon = 0{,}05$

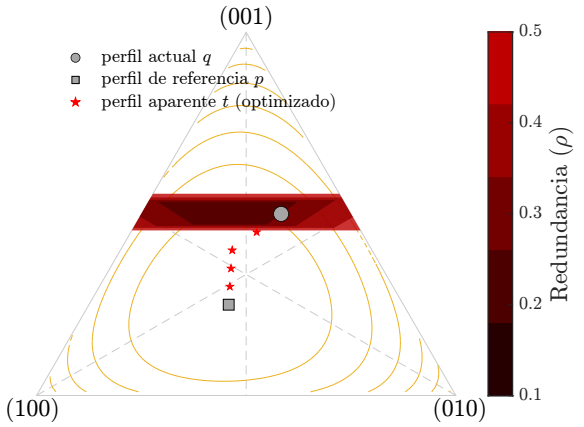


(b) adversario `naive` ρ-preciso

Figura 2. Incertidumbre $I_{\hat{Q}}$ para un adversario `naive` ϵ-preciso (a) y ρ-preciso (b). Este adversario desconoce por completo la forma de la solución óptima r^* del mecanismo. La incertidumbre, que aumenta en función de ρ, varía entre el 1 % y el 32 % para el adversario ϵ-preciso y entre el 0.5 % y el 26 % en el caso del adversario ρ-preciso. Los valores están calculados sobre la superficie del simplex, con valor $\frac{\sqrt{3}}{4}$.



(a) adversario `rogue` ϵ-preciso, $\epsilon = 0{,}05$



(b) adversario `smart` ϵ-preciso, $\epsilon = 0{,}05$

Figura 3. Incertidumbre $I_{\hat{Q}}$ para un adversario `rogue` ϵ-preciso (a) y un adversario `smart` ϵ-preciso (b). Estos adversarios conocen en menor o mayor medida la forma de la solución óptima r^* del mecanismo. La incertidumbre, que aumenta en función de ρ, varía entre el 0.6 % y el 4 % para el adversario `rogue` y entre el 0 % y el 4 % en el caso del adversario `smart`. Los valores están calculados sobre la superficie del simplex, con valor $\frac{\sqrt{3}}{4}$.

visto que el conocimiento más potente que un adversario puede tener sobre OQF es la forma de la solución óptima r^*, esto es el número de componentes nulas que la caracterizan, dada su naturaleza determinista y su forma *"water-filling"*. Y que dentro del rango crítico de mejora de la privacidad, esta circunstancia reduce dramáticamente, incluso anula, el espacio en que el adversario busca el perfil real del usuario en su tarea de estimación, expresado en términos de incertidumbre.

Evaluar los criterios de privacidad y utilidad implementados hasta ahora en conjunción con la potencia de reconstrucción del perfil real del usuario por parte del adversario que analizamos en esta propuesta nos sitúa de nuevo frente a un problema abierto que requiere de nuevas estrategias, generales o específicas, con el objetivo de mejorar la protección ofrecida por OQF.

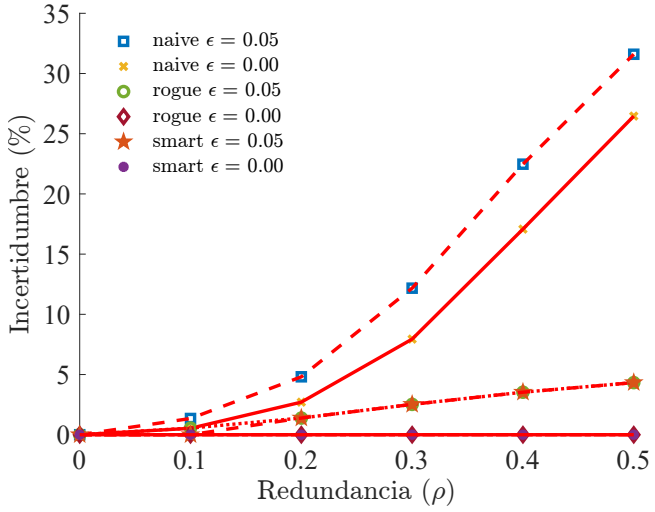
En este punto, lo primero que pensamos es en una intervención que aumente el área de incertidumbre de la estimación y de la que podrían beneficiarse todo el conjunto de DPTs, tanto las optimizadas que hemos revisado como las que no implementan esta propiedad. Una forma posible consiste en introducir aleatoriedad en el modelo mediante la adición de ruido calibrado en los diferentes momentos que se produce la perturbación, por ejemplo en la función objetivo [27], y estudiar el comportamiento de la solución óptima r^* y de la incertidumbre del adversario.

V. Conclusiones y oportunidades

En este trabajo presentamos un análisis de vulnerabilidad de OQF, la falsificación óptima de consultas, como técnica paradigmática y representativa de las DPTs optimizadas. En primer lugar hemos definido el contexto en que, entre las diferentes tipologías de PETs, estas técnicas resultan útiles. Sabemos que el ámbito de los PISs es un contexto de alta pervasividad y demanda general, vigencia y producción literaria. En contrapartida, también de elevado riesgo para la privacidad de los usuarios debido a los ingentes recursos destinados al perfilado sobre los datos privados que compartimos con los sistemas a cambio de servicios personalizados.

También hemos delimitado brevemente el estado del arte de estas técnicas que utilizan la optimización como marco analítico para modelar y resolver el compromiso entre privacidad y utilidad inherente en el ámbito de los PISs a través del uso de métricas de privacidad robustas y validadas procedentes del campo de la teoría de la información.

Finalmente, mediante un ejemplo sencillo hemos caracterizado los diferentes tipos de adversarios a que puede enfrentar-



Incertidumbre (%)

adversario	ϵ	ρ = 0	0.1	0.2	0.3	0.4	0.5
▸ naive	0.05	0	1.30	4.80	12.20	22.50	31.60
	0	0	0.50	2.70	7.90	17.10	26.50
▸ rogue	0.05	0	0.60	1.40	2.50	3.50	4.30
	0	0	0	0	0	0	0
▸ smart	0.05	0	0	1.40	2.50	3.50	4.30
	0	0	0	0	0	0	0

Figura 4. Comparación de la incertidumbre $I_{\hat{Q}}$ de los diferentes adversarios definidos. El peor adversario al que se enfrenta OQF es el que menor incertidumbre presenta, esto es, adversarios `rogue` o `smart` ρ-precisos, con incertidumbre nula en su estimación del perfil real del usuario q.

se OQF en función de su conocimiento sobre el mecanismo y hemos presentado una medida que cuantifica su incertidumbre al estimar el perfil real de un usuario. Observamos claramente que a mayor capacidad del adversario, menor, incluso nula, incertidumbre.

Los resultados obtenidos nos sitúan frente a un problema abierto que debe redundar en la mejora de la protección de la privacidad no solo del mecanismo analizado sino de sus análogos optimizados así como de aquellos que no incluyen en sus desempeños la optimización del compromiso entre privacidad y utilidad. Un futuro primer paso en la revisión de estos mecanismos se basa en inyectar aleatoriedad en diferentes momentos de la perturbación y estudiar el comportamiento de la incertidumbre del adversario.

Agradecimientos

Javier Parra-Arnau es el beneficiario de un contrato postdoctoral Ramón y Cajal (ref. RYC2021-034256-I) financiado por el Ministerio de Ciencia e Innovación de España y la Unión Europea — "NextGenerationEU/PRTR (Plan de Recuperación, Transformación y Resiliencia). Este trabajo ha sido parcialmente financiado por el Gobierno de España mediante el proyecto "DIstributed Smart Communications with Verifiable EneRgy-optimal Yields (DISCOVERY)"PID2023-148716OB-C32, y por la Generalitat de Catalunya, mediante la ayuda AGAUR "2021 SGR 01413".

Referencias

[1] E. Brynjolfsson and B. Kahin, *Understanding the digital economy: data, tools, and research*, 2002.

[2] N. Confessore, "Cambridge analytica and facebook: The scandal and the fallout so far," *The New York Times*, vol. 4, p. 2018, 2018.

[3] J. Parra-Arnau, D. Rebollo-Monedero, and J. Forné, "Measuring the privacy of user profiles in personalized information systems," *Future Generation Computer Systems*, vol. 33, pp. 53–63, 2014.

[4] ——, "Privacy-enhancing technologies and metrics in personalized information systems," in *Advanced Research in Data Privacy*. Springer, 2015, pp. 423–442.

[5] N. Kaaniche, M. Laurent, and S. Belguith, "Privacy enhancing technologies for solving the privacy-personalization paradox: Taxonomy and survey," *Journal of Network and Computer Applications*, p. 102807, 2020.

[6] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[7] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[8] S. Ye, F. Wu, R. Pandey, and H. Chen, "Noise injection for search privacy protection," in *2009 International Conference on Computational Science and Engineering*, vol. 3. IEEE, 2009, pp. 1–8.

[9] D. Rebollo-Monedero and J. Forné, "Optimized query forgery for private information retrieval," *IEEE Transactions on Information Theory*, vol. 56, no. 9, pp. 4631–4642, 2010.

[10] S. Puglisi, J. Parra-Arnau, J. Forné, and D. Rebollo-Monedero, "On content-based recommendation and user privacy in social-tagging systems," *Computer Standards & Interfaces*, vol. 41, pp. 17–27, 2015.

[11] A. Jimenez-Berenguel, C. Gil, C. Garcia-Rubio, J. Forné, and C. Campo, "Dns query forgery: a client-side defense against mobile app traffic profiling," *arXiv preprint arXiv:2505.09374*, 2025.

[12] J. Parra-Arnau, D. Rebollo-Monedero, J. Forné, J. L. Muñoz, and O. Esparza, "Optimal tag suppression for privacy protection in the semantic web," *Data & Knowledge Engineering*, vol. 81, pp. 46–66, 2012.

[13] J. Parra-Arnau, A. Perego, E. Ferrari, J. Forne, and D. Rebollo-Monedero, "Privacy-preserving enhanced collaborative tagging," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 1, pp. 180–193, 2012.

[14] C. Gil, J. Parra-Arnau, and J. Forne, "Privacy protection against user profiling through optimal data generalization," *Computers & Security*, vol. 148, p. 104178, 2025.

[15] Y. Xu, K. Wang, B. Zhang, and Z. Chen, "Privacy-enhancing personalized web search," in *Proceedings of the 16th international conference on World Wide Web*, 2007, pp. 591–600.

[16] V. Toubiana, A. Narayanan, D. Boneh, H. Nissenbaum, and S. Barocas, "Adnostic: Privacy preserving targeted advertising," in *Proceedings Network and Distributed System Symposium*, 2010.

[17] M. Fredrikson and B. Livshits, "Repriv: Re-envisioning in-browser privacy," in *Proc. IEEE Symp. Security, Privacy (SP)(May 2011)*, 2010.

[18] J. Parra-Arnau, J. P. Achara, and C. Castelluccia, "Myadchoices: Bringing transparency and control to online advertising," *ACM Transactions on the Web (TWEB)*, vol. 11, no. 1, pp. 1–47, 2017.

[19] D. Howe and H. Nissenbaum, "Lessons from the identity trail: Privacy, anonymity and identity in a networked society. ny: Oxford univ. press, 2009, ch. trackmenot: Resisting surveillance in web search," *http://mrl.nyu.edu/ dhowe/trackmenot*, pp. 417–436, 2009.

[20] E. Balsa, C. Troncoso, and C. Diaz, "Ob-pws: Obfuscation-based private web search," in *2012 IEEE Symposium on Security and Privacy*. IEEE, 2012, pp. 491–505.

[21] Y. Elovici, B. Shapira, and A. Maschiach, "A new privacy model for hiding group interests while accessing the web," in *Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*, 2002, pp. 63–70.

[22] ——, "A new privacy model for web surfing," in *International Workshop on Next Generation Information Technologies and Systems*. Springer, 2002, pp. 45–57.

[23] Y. Elovici, C. Glezer, and B. Shapira, "Enhancing customer privacy while searching for products and services on the world wide web," *Internet Research*, 2005.

[24] Y. Elovici, B. Shapira, and A. Meshiach, "Cluster-analysis attack against a private web solution (praw)," *Online Information Review*, 2006.

[25] J. Domingo-Ferrer, A. Solanas, and J. Castellà-Roca, "h(k)-private information retrieval from privacy-uncooperative queryable databases," *Online Information Review*, 2009.

[26] M. Murugesan and C. Clifton, "Providing privacy through plausibly deniable search," in *Proceedings of the 2009 SIAM International Conference on Data Mining*. SIAM, 2009, pp. 768–779.

[27] M. Avriel, *Nonlinear programming: analysis and methods*. Courier Corporation, 2003.

Resumen Extendido: Rewardable Naysayer Proofs

Gennaro Avitabile
IMDEA Software Institute
Madrid, Spain
gennaro.avitabile@imdea.org

Luisa Siniscalschi
Technical University of Denmark (DTU)
Copenhagen, Denmark
luisi@dtu.dk

Ivan Visconti
Sapienza University of Rome
Rome,Italy
ivan.visconti@uniroma1.it

***Resumen*—Este es un resumen extendido de la publicación con el título "Short Paper: Rewardable Naysayer Proofs" y los mismos autores que aparece en las actas de convenio FC 2025, Part II, LNCS 15752, https://doi.org/10.1007/978-3-032-07035-7_15. Combining verifiable computation with optimistic approaches is a promising direction to scale blockchain applications. The basic idea consists of saving computations by avoiding the verification of proofs unless there are complaints.**

A key tool to design systems in the above direction has been recently proposed by Seres, Glaeser and Bonneau [FC'24] who formalized the concept of a Naysayer proof: an efficient to verify proof disproving a more demanding to verify original proof.

In this work, we discuss the need of rewarding naysayer provers, the risks deriving from front-running attacks, and the failures of generic approaches trying to defeat them. Next, we introduce the concept of verifiable delayed naysayer proofs and show a construction leveraging proofs of sequential work, without relying on any additional infrastructure.

***Index Terms*—Rollups, Naysayer Proofs, Front-running**

I. Introduction

In most blockchains the cost of a transaction for the end user depends on the amount of storage and computations of the invoked on-chain programs. To make an on-chain application successful and scalable, developers are therefore highly incentivized to minimize such costs for the end users. To this end, verifiable computation [1] has become a very popular tool. Verifiable computation outsources an expensive computation to an untrusted party which provides the result together with a *succinct* non-interactive proof (SNARK) that the result is correct. In the blockchain setting, this leads to a system where smart contracts verify proofs, which is much more efficient than executing the computations themselves.

An important application of this paradigm is zk-rollups (e.g., [2]). A zk-rollup is a scaling solution that involves batching several transactions into a *single* one that updates the state of the blockchain to the one resulting from the execution of all the transactions. To guarantee that the state is not compromised, a succinct proof that the new state is correct is also provided by a rollup coordinator. However, verifying such proofs is often wasteful, as in blockchain applications an incentive mechanism can always be put in place to punish whoever publishes incorrect proofs. Therefore, optimistic paradigms [3, 4, 5] assuming that coordinators are unlikely to publish incorrect information on-chain seem a promising avenue to reduce costs.

Seres et al. [5] recently proposed a new paradigm called *naysayer proofs*. In naysayer proofs, the verifier (i.e., the smart contract) optimistically accepts a proof without verifying it, opening a dispute period in which the proof can be rejected and the state reverted. Instead, off-chain validators will verify the proof and, if it is incorrect, they will submit a *naysayer proof* certifying its *incorrectness*. The naysayer verifier checks the naysayer proof and, if it is correct, rejects the original proof. We call these off-chain validators naysayer provers. The advantage of naysayer proofs is two-fold: first, in most cases, the naysayer proof will not be generated at all since the sole fact that they can be generated disincentivizes attempts of generating incorrect proofs; second, should the proof be challenged, naysayer proofs are more efficient to verify than the original proofs.

Naysayer proofs vs fraud proofs: A concept related to naysayer proofs is the one of *fraud proofs* that is used in the context of *optimistic rollups* [4, 3]. Fraud proofs operate under the optimistic assumption that a computation is correct unless someone challenges it. In case of a challenge, (some of) the steps of the computation are re-executed to resolve the dispute. Although naysayer proofs and fraud proofs are conceptually similar, there are some differences coming from the fact that what is disputed is a proof about a computation rather than the computation itself. In naysayer proofs, unlike fraud proofs, dispute resolution is *witness independent* [5]. Indeed, for fraud proofs, the entire input of the computation (i.e., the witness) must be made available at least to the challenger. On the other end, to verify a naysayer proof only the original statement and proof are required, which are usually much shorter than the witness.

As observed in [5], witness-independent resolutions can lead to considerable savings for large witnesses such as transaction data of a rollup. For example, consider a zk-rollup batching several transactions in a succinct commitment (e.g., a Merkle tree) and a SNARK certifying that the committed transactions are correct and produce a certain final state. Due to the succinctness of these cryptographic tools, a single rollup transaction could batch hundreds of thousands of transactions. Such a proof can be extremely resource intensive and the rollup coordinator might actually implement it via hundreds of commitment/SNARK pairs where each pair deals with thousands of transactions and, according to the optimistic approach, is provisionally accepted. The advantage of naysayer proofs over fraud proofs becomes clear: every rollup update could contain a large number of transactions while a corresponding naysayer proof consisting of simply pointing to a single commitment/SNARK pair is fast to verify; nevertheless, unlike optimistic rollups, the corresponding transactions, thanks to witness independence, do not need to be made available to verifiers and validators.

Naysayer proofs and front-running attacks: To work in a permissionless setting, the naysayer paradigm requires an incentive mechanism ensuring that participation is profitable for the naysayer provers. The naysayer provers who post

correct naysayer proofs should be rewarded and should be penalized in case they transmit incorrect proofs. Additionally, whoever posts a proof which is later proved incorrect by a naysayer prover must also be penalized[1].

Generally, all applications in which any user can obtain a reward by submitting a transaction before other participants are targets of *front-running* attacks [6, 7]. Here the front-runner uses knowledge of unprocessed transactions to get her transactions executed ahead of the unprocessed ones. Therefore, when considering systems rewarding naysayer provers, the attacker could steal the naysayer proof from the transactions pool and rush the publication of her naysayer proof to cash the reward, without doing the hard work of a naysayer prover (i.e., without validating huge amounts of original proofs).

Solutions to front-running: Several techniques with various trade-offs have been proposed to mitigate generic and specific front-running attacks [8]. A common approach consists of batching blinded transactions, so that all transactions submitted within a confirmation period are independent from each other, preventing any adversarial ordering of the transactions. For instance, *commit and reveal* approaches [9, 10, 11] involve a first round where all inputs are committed and a second round where such commitments are opened so that rewards can be paid accordingly. This technique requires multiple rounds of interaction between users and the blockchain. Finally, while preventing front-running attacks, the commit-and-reveal approach introduces selective opening attacks that can be mitigated through additional complexity such as the use of timed-release cryptography (e.g., timed commitments [12] or delay encryption [13]), ensuring that the inputs can always be recovered after a certain time, or by relying on threshold encryption [14] and a committee where the majority of the members is assumed to be honest, or by a multi-party computation of an anonymous committed broadcast functionality [15].

Ensuring delay to prevent front-running: Another method to prevent front-running does not blind the transactions' inputs, but it instead focuses on making the creation of a valid transaction slow [16, 17]. The central tool in this approach are verifiable delay functions (VDFs) [18]. A VDF takes a prescribed amount of time t (even in the presence of massive parallelism) to compute and can be verified in time much less than t. The overall idea is to tie the input of the VDF to the transaction as well as some user's identifying information. A transaction is then considered valid only if it comes with a proof of the VDF. Therefore, every time attackers want to front-run a transaction, they have to start a long-running computation. The time parameter t is set so that by the time the attacker finalizes her transaction, the transaction of the honest user will already be confirmed in the blockchain. The advantage of this approach is that it does not require a user to submit a second transaction. Unfortunately, the above generic use of a VDF to defeat front-running attacks suffers from the following technical caveat: the sequentiality of a VDF is only guaranteed when it is evaluated on a random input. This generally requires an external trusted infrastructure providing a publicly verifiable randomness beacon [19]. It is therefore interesting to study scenarios where no trusted external infrastructure (i.e., no additional points of failure) is available and still front-running can be defeated via delays.

[1]The reward for posting correct naysayer proofs must be lower than the penalty given to users who post incorrect proofs. Otherwise, one could profit by posting an incorrect proof and shortly after a corresponding naysayer proof.

I-A. Our Contribution

In this paper, we study the problem of front-running attacks against rewarding mechanisms associated to naysayer proofs. We make the following contributions:

- *Front-running resistant naysayer proofs:* we define the notion of verifiable delayed naysayer proofs (VDNs). VDNs are naysayer proofs that when plugged in a natural reward-based system are resilient to front-running attacks. We design a compiler which, starting from proofs of sequential works (PoSWs) [20] and a naysayer proof with minimal additional requirements, gives a VDN. Our solution does not add any setup assumption and, being tied to the setting of naysayer proofs, it does not require randomness beacons.
- *Generic front-running protection approaches can be harmful:* We show that applying a generic front-running protection technique can hurt the requirements of the underlying reward-based system. In particular, we observe that naysayer proofs have a crucial efficiency requirement (i.e., they should be faster to verify than the original proofs) that is not necessarily preserved when a front-running protection approach is applied on top of them.

Verifiable delayed naysayer proofs: In VDNs, the naysayer prover takes as input an additional time parameter t and a prover-chosen watermark μ. The naysayer verifier also takes such additional values as input. Apart from completeness and soundness, we require the VDN to be sequential. By sequential we mean that given a sufficiently unpredictable statement/proof pair (x, π) that can be naysayed, it is unfeasible to compute an accepting naysayer proof for (x, π), with a watermark μ, in less than time t. This holds even when the adversary is provided with oracle access to fast generation of VDN proofs, except of course on (x, π, μ). When using the VDN in a reward-based setting, it suffices to additionally verify that the prover-chosen watermark matches with the public key which submits the transaction. Indeed, sequentiality guarantees that even if the front-runner $\mathcal{F}$ sees a VDN proof of an honest user $\mathcal{H}$ (w.r.t. her public key $\mathsf{pk}_{\mathcal{H}}$), $\mathcal{F}$ will not be able to produce a valid proof (w.r.t. her public key $\mathsf{pk}_{\mathcal{F}}$) before the transaction submitted by $\mathcal{H}$ is finalized in the blockchain.

Our VDN: We combine naysayer proofs with PoSWs to construct a VDN. PoSWs are publicly verifiable proofs attesting that a certain amount of sequential work was performed starting from a specific challenge. VDFs can be seen as a special case of PoSWs offering the additional guarantee of a unique output. We observe that PoSWs suffice for front-running prevention. Indeed, we do not need a unique output since such output is merely verified to ensure that a long sequential computation was carried out, but the result of the computation is not used elsewhere. Unlike VDFs, we know PoSWs that are unconditionally secure in the random oracle model and do not require any setup [20, 21].

Our construction is very simple: the prover of the VDN runs the underlying naysayer prover over (x, π) and gets a

naysayer proof π_{nay}. Then, it calls a random oracle (RO) over $(\pi_{\mathsf{nay}}||x||\pi||\mu)$ to get the challenge χ for the PoSW. The PoSW π_{PoSW} is then computed w.r.t. χ and time parameter t. The VDN proof simply consists of both π_{nay} and π_{PoSW}, which are then individually verified by the verifier of the VDN. The caveat of this solution is that the security of PoSWs is formulated for uniformly random χ. Even if χ is derived from a RO, χ is a random string only if $(\pi_{\mathsf{nay}}, x, \pi, \mu)$ has high min-entropy. Generally, it is unclear how to find a high min-entropy source for all the users of a system.

Whenever a reward is connected to the punishment of another party, as in the naysayer setting, it is actually possible to find a source of high min-entropy. For example, in the rollup application discussed above, x is a succinct representation (e.g., a cryptographic hash) of several transactions, which likely contain cryptographic material with high min-entropy such as digital signatures. Furthermore, there are generally many valid proofs π for the same x. The malicious rollup coordinator is incentivized to use good entropy to create (x, π), so that this pair is distributed similarly to honest proofs, and thus less likely to be verified by a naysayer prover. Moreover, we can include in π the digital signature used to authorize the rollup transaction. On an intuitive level, such a signature must be unpredictable; otherwise it would not be unforgeable and an attacker could damage the coordinators posting wrong proofs on their behalf.

Front-running and efficiency in naysayer proofs: The verification algorithm of our VDN is slower than the one of the underlying naysayer proof system since it additionally verifies a PoSW. Therefore, to prove that our VDN is a naysayer proof system we have to assume that the verification algorithm of the PoSW is asymptotically faster than the verification algorithm of the original proof[2].

This is not a problematic assumption, since the verification time of state-of-the-art PoSWs [21, 22] scales logarithmically in t and as a small polynomial in the security parameter. However, this highlights that not all front-running compilers are necessarily compatible with every reward-based system. For example, one could artificially create a PoSW whose verification time depends on a high-degree polynomial in the security parameter while still being logarithmic in t, making it asymptotically slower than the verification of the original proof. The resulting construction would still prevent front-running attacks but it would also make the naysayer reward-based system useless since verifying regular proofs would be less expensive than verifying naysayer proofs.

Obviously, as in all applications of timed primitives, estimating the right value of t is very challenging. However, we feel that this issue will naturally be better understood as these primitives get used in practice.

II. Preliminaries

Definition 1: A non-interactive proof system Π for an NP relation $\mathcal{R}$ consists of the following PPT algorithms:

- $\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)$: on input parameter λ, output crs.
- $\pi \leftarrow \mathsf{Prv}(\mathsf{crs}, x, w)$: on input crs, x, and w, output π.
- $0/1 \leftarrow \mathsf{Vfy}(\mathsf{crs}, x, \pi)$: on input crs, x, and π output a bit.

[2]Recall that the verification of a naysayer proof must be faster than the verification of the original proof.

We require Π to be correct and sound, we defer to full version for formal definitions.

Definition 2: Given a non-interactive proof system $\Pi = (\mathsf{Setup}, \mathsf{Prv}, \mathsf{Vfy})$ for some NP relation $\mathcal{R}$, the corresponding naysayer proof system Π_{nay} consists of the PPT algorithms defined as follows. We define τ to be a tuple (crs, x, π), where crs, x, π are a common reference string, a statement, and a proof of the proof system Π. We call τ a naysayer tuple.

- $\mathsf{crs}_{\mathsf{nay}} \leftarrow \mathsf{NSetup}(1^\lambda)$: on input security parameter 1^λ, output $\mathsf{crs}_{\mathsf{nay}}$.
- $\pi_{\mathsf{nay}} \leftarrow \mathsf{Naysay}(\mathsf{crs}_{\mathsf{nay}}, \tau)$: on input $\mathsf{crs}_{\mathsf{nay}}$, and a naysayer tuple τ, output π_{nay}.
- $0/1 \leftarrow \mathsf{NVfy}(\mathsf{crs}_{\mathsf{nay}}, \tau, \pi_{\mathsf{nay}})$: on input $\mathsf{crs}_{\mathsf{nay}}$, a naysayer tuple τ, and π_{nay}, outputs 1 to accept and 0 to reject.

We define the polynomial-time language $\mathcal{L}^{\Pi}_{\mathsf{nay}} := \{\tau = (\mathsf{crs}, x, \pi) : \Pi.\mathsf{Vfy}(\mathsf{crs}, x, \pi) = 0\}$ and require the following properties to hold:

- Naysayer Completeness: for all $\tau \in \mathcal{L}^{\Pi}_{\mathsf{nay}}$:

$$\Pr\left[\mathsf{NVfy}(\mathsf{crs}_{\mathsf{nay}}, \tau, \pi_{\mathsf{nay}}) = 1 \;\middle|\; \begin{array}{l}\mathsf{crs}_{\mathsf{nay}} \leftarrow \mathsf{NSetup}(1^\lambda);\\ \pi_{\mathsf{nay}} \leftarrow \mathsf{Naysay}(\mathsf{crs}_{\mathsf{nay}}, \tau)\end{array}\right] = 1.$$

- Naysayer Soundness: for all PPT adversaries $\mathcal{A}$:

$$\Pr\left[\begin{array}{l}\mathsf{NVfy}(\mathsf{crs}_{\mathsf{nay}}, \mathsf{crs}, \tau) = 1 \wedge\\ \tau \notin \mathcal{L}^{\Pi}_{\mathsf{nay}}\end{array} \;\middle|\; \begin{array}{l}\mathsf{crs}_{\mathsf{nay}} \leftarrow \mathsf{NSetup}(1^\lambda);\\ (\pi_{\mathsf{nay}}, \tau) \leftarrow \mathcal{A}(\mathsf{crs}_{\mathsf{nay}})\end{array}\right] \leq \mathsf{negl}(\lambda).$$

- Efficiency: NVfy must be asymptotically faster than Vfy.

Definition 3: A PoSW is a pair of PPT oracle-aided algorithms $\mathsf{Prv}, \mathsf{Vfy}$, run by a prover P and a verifier V as follows:

Input: P and V get as common input $\lambda, t \in \mathbb{N}$. All parties have access to a random oracle $\mathsf{H} : \{0,1\}^* \to \{0,1\}^\lambda$.
Statement: V samples a random statement $\chi \leftarrow \{0,1\}^\lambda$ and sends it to P.
Prove: P computes $\pi \leftarrow \mathsf{Prv}^{\mathsf{H}}(\chi, t)$ and sends π to V.
Verify: V computes and outputs $\mathsf{Vfy}^{\mathsf{H}}(\chi, t, \pi)$.

We require the PoSW to be complete, sound, and succinct:

- Completeness: for all $\lambda, t \in \mathbb{N}$, and all $\chi \in \{0,1\}^\lambda$:

$$\Pr\left[\mathsf{Vfy}^{\mathsf{H}}(\chi, t, \pi) = 1 \;\middle|\; \pi \leftarrow \mathsf{Prv}^{\mathsf{H}}(\chi, t)\right] = 1.$$

- Soundness: for all $\lambda, t \in \mathbb{N}$ and $\alpha > 0$, for all $\mathcal{A}$ making $(1-\alpha)t$ sequential[3] queries to H:

$$\Pr\left[\mathsf{Vfy}^{\mathsf{H}}(\chi, t, \pi) = 1 \;\middle|\; \chi \leftarrow \{0,1\}^\lambda, \pi \leftarrow \mathcal{A}^{\mathsf{H}}(\chi)\right] \leq \mathsf{negl}(\lambda).$$

- Succinctness: for all $\lambda, t \in \mathbb{N}$ and honestly generated proof π, we have $|\pi| \leq \mathsf{poly}(\lambda, \log t)$, $\mathsf{Prv}^{\mathsf{H}}$ runs in time[4] $\mathsf{poly}(\lambda, t)$, and $\mathsf{Vfy}^{\mathsf{H}}$ runs in time $\mathsf{poly}(\lambda, \log t)$.

III. Verifiable Delayed Naysayer Proofs

Definition 4: Given a non-interactive proof system $\Pi = (\mathsf{Setup}, \mathsf{Prv}, \mathsf{Vfy})$ for some NP relation $\mathcal{R}$, the corresponding verifiable delayed naysayer proof system Π_{VDN} consists of the oracle-aided PPT algorithms defined as follows:

- $\mathsf{crs}_{\mathsf{VDN}} \leftarrow \mathsf{VDNSetup}(1^\lambda)$: on input security parameter 1^λ, output a common reference string $\mathsf{crs}_{\mathsf{VDN}}$.

[3]Notice that no restriction is posed on the number of *parallel* queries $\mathcal{A}$ can make. In a nutshell, the budget of sequential queries gets reduced whenever a new query includes the output of a previously asked RO query as a sub-string.

[4]To simplify the notation, in accordance to [22], we give t as input to all the algorithms. However, we implicitly mean that, whenever required to comply with the efficiency requirements, the time parameter t is in unary.

- $\pi_{\mathsf{VDN}} \leftarrow \mathsf{VDNaysay}^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}}, t, \tau, \mu)$: on input common reference string $\mathsf{crs}_{\mathsf{VDN}}$, time parameter t, naysayer tuple τ, and watermark μ, output a proof π_{VDN}.
- $0/1 \leftarrow \mathsf{VDNVfy}^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}}, t, \tau, \mu, \pi_{\mathsf{VDN}})$: on input common reference string $\mathsf{crs}_{\mathsf{VDN}}$, time parameter t, naysayer tuple τ, watermark μ, and a proof π_{VDN}, output 1 to accept or 0 to reject.

Completeness, soundness, and efficiency are analogous to the ones of Def. 2. while $\mathcal{Z}$-*Watermarked Sequentiality* is defined as follow: For all $\lambda, t \in \mathbb{N}$ and $\alpha > 0$, and for an auxiliary input distribution $\mathcal{Z}$, we define the following game applied to an adversary $\mathcal{A} := (\mathcal{A}_0, \mathcal{A}_1)$:

$$\mathsf{crs}_{\mathsf{VDN}} \leftarrow \mathsf{VDNSetup}(1^\lambda), \mathsf{st} \leftarrow \mathcal{A}_0^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}}, t), \tau \leftarrow \mathcal{Z}(\mathsf{crs}_{\mathsf{VDN}}, t)$$
$$(\pi^*_{\mathsf{VDN}}, \mu^*) \leftarrow \mathcal{A}_1^{\mathsf{H}(\cdot),\mathsf{VDNaysay}^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}},t,\cdot,\cdot)}(\mathsf{st}, \tau)$$

We say that $\mathcal{A}$ wins the game if $\mathsf{VDNVfy}^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}}, t, \tau, \mu^*, \pi^*_{\mathsf{VDN}}) = 1$ and $\mathcal{A}$ did not query the oracle over (τ, μ^*). Π_{VDN} satisfies $\mathcal{Z}$-watermarked sequentiality if no pair of randomized algorithm $\mathcal{A}_0$ which runs in total time $\mathsf{poly}(t, \lambda)$, and $\mathcal{A}_1$ that makes at most $(1-\alpha)t$ sequential queries to H, can win the above game with probability greater than $\mathsf{negl}(\lambda)$.

Our PoSW-based construction:

- $\mathsf{crs}_{\mathsf{VDN}} \leftarrow \mathsf{VDNSetup}(1^\lambda)$: $\mathsf{crs}_{\mathsf{nay}} \leftarrow \Pi_{\mathsf{nay}}.\mathsf{Setup}(1^\lambda), \mathsf{crs}_{\mathsf{VDN}} := \mathsf{crs}_{\mathsf{nay}}$.
- $\pi_{\mathsf{VDN}} \leftarrow \mathsf{VDNaysay}^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}}, t, \tau, \mu)$: parse $\mathsf{crs}_{\mathsf{VDN}} := \mathsf{crs}_{\mathsf{nay}}$, $\pi_{\mathsf{nay}} \leftarrow \Pi_{\mathsf{nay}}.\mathsf{Naysay}(\mathsf{crs}_{\mathsf{nay}}, \tau)$, $\chi = \mathsf{H}(\pi_{\mathsf{nay}}||\tau||\mu)$, $\pi_{\mathsf{PoSW}} \leftarrow \mathsf{PoSW.Prv}^{\mathsf{H}}(\chi, t)$, $\pi_{\mathsf{VDN}} := (\pi_{\mathsf{nay}}, \pi_{\mathsf{PoSW}})$.
- $0/1 \leftarrow \mathsf{VDNVfy}^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}}, t, \tau, \mu, \pi_{\mathsf{VDN}})$: parse $\mathsf{crs}_{\mathsf{VDN}} := \mathsf{crs}_{\mathsf{nay}}$ and $\pi_{\mathsf{VDN}} := (\pi_{\mathsf{nay}}, y, \pi_{\mathsf{PoSW}})$, $\chi = \mathsf{H}(\pi_{\mathsf{nay}}||\tau||\mu)$, output $\mathsf{PoSW.Vfy}^{\mathsf{H}}(\chi, t, \pi_{\mathsf{PoSW}}) \wedge \Pi_{\mathsf{nay}}.\mathsf{NVfy}(\mathsf{crs}_{\mathsf{nay}}, \tau, \pi_{\mathsf{nay}})$.

Theorem 1: If Π_{nay} is a naysayer proof system w.r.t. Π, if PoSW is a proof of sequential work, if $\mathcal{Z}$ is the distribution of naysayable proofs generated with high min-entropy, and if $\mathsf{PoSW.Vfy}^{\mathsf{H}}$ is asymptotically faster than $\Pi.\mathsf{Vfy}$, then the above construction is a VDN.

Proof: We give a short proof sketch.

Completeness: it follows by inspection.

Soundness: it follows from the soundness of Π_{nay}.

$\mathcal{Z}$-Watermarked Sequentiality: it follows from the fact that a $\pi_{\mathsf{nay}}||\tau||\mu$ with high min-entropy gives a uniformly distributed χ when hashed down with a random oracle, and from the soundness of the PoSW. Intuitively, all the polynomially many queries of $\mathcal{A}_0$ do not give it any insight on χ, as well as the polynomially many queries done by $\mathcal{A}_1$ to $\mathsf{VDNaysay}^{\mathsf{H}}(\mathsf{crs}_{\mathsf{VDN}}, \mathsf{crs}, t, \cdot, \cdot)$ over $(\tau', \mu') \neq (\tau, \mu^*)$. Therefore, one can simply reduce watermarked sequentiality to the soundness of PoSW.

Efficiency: the running time of $\mathsf{VDNVfy}^{\mathsf{H}}$ is the sum of the running times of the verification algorithms of Π_{nay} and PoSW, which are both asymptotically faster than $\Pi.\mathsf{Vfy}$, the same holds for $\mathsf{VDNVfy}^{\mathsf{H}}$.

■

IV. Acknowledgments

We thank Hamza Abusalah for insightful discussions and the anonymous reviewers of FC 2025 and RECSI 2026 for their feedback. This work has been supported by the grants PID2022-142290OB-I00 (ESPADA project co-funded by FEDER, EU), JDC2023-050791-I (co-funded by the ESF+), and TED2021-132464B-I00 (PRODIGY project, co-funded by the European Union NextGenerationEU/PRTR), all funded by MCIN/AEI/10.13039/501100011033. It has also received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (PICOCRYPT project, Grant No. 101001283). Ivan Visconti is member of the Gruppo Nazionale Calcolo Scientifico-Istituto Nazionale di Alta Matematica and his contribution on this work was in part supported under the National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.1, Call for tender No. 104 published on 2.2.2022 by the Italian Ministry of University and Research (MUR), funded by the European Union -NextGenerationEU - Project Title "PARTHENON" - CUP D53D23008610006 - Grant Assignment Decree No. 959.

Referencias

[1] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in *CRYPTO 2010*, ser. LNCS, T. Rabin, Ed., vol. 6223. Springer, Berlin, Heidelberg, Aug. 2010, pp. 465–482.

[2] Linea, "Documentation," https://docs.linea.build/.

[3] Optimism, "Docs," https://docs.optimism.io/.

[4] Arbitrum, "Docs," https://docs.arbitrum.io.

[5] I. A. Seres, N. Glaeser, and J. Bonneau, "Short paper: Naysayer proofs," in *FC 2024, March 2024, Revised Selected Papers, Part II*, ser. Lecture Notes in Computer Science, J. Clark and E. Shi, Eds., vol. 14745. Springer, 2024, pp. 22–32. [Online]. Available: https://doi.org/10.1007/978-3-031-78679-2_2

[6] P. Daian, S. Goldfeder, T. Kell, Y. Li, X. Zhao, I. Bentov, L. Breidenbach, and A. Juels, "Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability," in *2020 IEEE Symposium on Security and Privacy*. IEEE Computer Society Press, May 2020, pp. 910–927.

[7] S. Eskandari, S. Moosavi, and J. Clark, "Sok: Transparent dishonesty: Front-running attacks on blockchain," in *FC 2019*, A. Bracciali, J. Clark, F. Pintore, P. B. Rønne, and M. Sala, Eds. Cham: Springer International Publishing, 2020, pp. 170–189.

[8] C. Baum, J. H. Chiang, B. David, T. K. Frederiksen, and L. Gentile, "Sok: Mitigation of front-running in decentralized finance," in *FC 2022 International Workshops*, ser. LNCS, S. Matsuo, L. Gudgeon, A. Klages-Mundt, D. P. Hernandez, S. Werner, T. Haines, A. Essex, A. Bracciali, and M. Sala, Eds., vol. 13412. Springer, 2022, pp. 250–271. [Online]. Available: https://doi.org/10.1007/978-3-031-32415-4_17

[9] J. Bebel and D. Ojha, "Ferveo: Threshold decryption for mempool privacy in BFT networks," Cryptology ePrint Archive, Report 2022/898, 2022. [Online]. Available: https://eprint.iacr.org/2022/898

[10] A. R. Choudhuri, S. Garg, J. Piet, and G.-V. Policharla, "Mempool privacy via batched threshold encryption: Attacks and defenses," in *USENIX Security 2024*, D. Balzarotti and W. Xu, Eds. USENIX Association, Aug. 2024.

[11] I. Abraham, B. Pinkas, and A. Yanai, "Blinder - scalable, robust anonymous committed broadcast," in *ACM CCS 2020*, J. Ligatti, X. Ou, J. Katz, and G. Vigna, Eds. ACM Press, Nov. 2020, pp. 1233–1252.

[12] D. Boneh and M. Naor, "Timed commitments," in *CRYPTO 2000*, ser. LNCS, M. Bellare, Ed., vol. 1880. Springer, Berlin, Heidelberg, Aug. 2000, pp. 236–254.

[13] J. Burdges and L. De Feo, "Delay encryption," in *EUROCRYPT 2021, Part I*, ser. LNCS, A. Canteaut and F.-X. Standaert, Eds., vol. 12696. Springer, Cham, Oct. 2021, pp. 302–326.

[14] R. Canetti and S. Goldwasser, "An efficient threshold public key cryptosystem secure against adaptive chosen ciphertext attack," in *EUROCRYPT'99*, ser. LNCS, J. Stern, Ed., vol. 1592. Springer, Berlin, Heidelberg, May 1999, pp. 90–106.

[15] D. Lu, T. Yurek, S. Kulshreshtha, R. Govind, A. Kate, and A. Miller, "Honeybadgermpc and asynchromix: Practical asynchronous mpc and its application to anonymous communication," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 887–903. [Online]. Available: https://doi.org/10.1145/3319535.3354238

[16] E. Chen and A. Chon, "Injective protocol: A collision resistant decentralized exchange protocol," 2018.

[17] E. Sariboz, G. Panwar, R. Vishwanathan, and S. Misra, "FIRST: frontrunning resilient smart contracts," *CoRR*, vol. abs/2204.00955, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2204.00955

[18] D. Boneh, J. Bonneau, B. Bünz, and B. Fisch, "Verifiable delay functions," in *CRYPTO 2018, Part I*, ser. LNCS, H. Shacham and A. Boldyreva, Eds., vol. 10991. Springer, Cham, Aug. 2018, pp. 757–788.

[19] K. Choi, A. Manoj, and J. Bonneau, "SoK: Distributed randomness beacons," in *2023 IEEE Symposium on Security and Privacy*. IEEE Computer Society Press, May 2023, pp. 75–92.

[20] M. Mahmoody, T. Moran, and S. P. Vadhan, "Publicly verifiable proofs of sequential work," in *ITCS 2013*, R. D. Kleinberg, Ed. ACM, Jan. 2013, pp. 373–388.

[21] B. Cohen and K. Pietrzak, "Simple proofs of sequential work," in *EUROCRYPT 2018, Part II*, ser. LNCS, J. B. Nielsen and V. Rijmen, Eds., vol. 10821. Springer, Cham, Apr. / May 2018, pp. 451–467.

[22] N. Döttling, R. W. F. Lai, and G. Malavolta, "Incremental proofs of sequential work," in *EUROCRYPT 2019, Part II*, ser. LNCS, Y. Ishai and V. Rijmen, Eds., vol. 11477. Springer, Cham, May 2019, pp. 292–323.

De reseñas a reputación: un pipeline auditable, explicable y robusto para servicios en línea

Elmira Saeedi Taleghani, Ronald Iván Maldonado Valencia, Ana Lucila Sandoval Orozco,
Luis Javier García Villalba
Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Emails: {elmirasa, ronaldim, asandov, javierv}@ucm.es

***Resumen*—Las valoraciones por estrellas, usadas como proxy de reputación, son ruidosas, manipulables y ciegas al sentimiento por aspectos. Esta fragilidad puede distorsionar rankings, perjudicar la experiencia del usuario e inducir decisiones de negocio equivocadas. Se presenta *Scrape2Repute*, una canalización compacta y reproducible que aborda estas carencias: (i) ingiere reseñas cumpliendo políticas; (ii) limpia y normaliza texto/metadatos; (iii) aprende un sentimiento calibrado por reseña; (iv) fusiona estrellas y texto mediante una etiqueta híbrida ajustable; (v) atenúa reseñas sospechosas con Isolation Forest; y (vi) agrega evidencia con decaimiento temporal e intervalos de incertidumbre. El diseño prioriza auditoría y explicabilidad (racionales top-k, resúmenes por aspecto) y es eficiente en hardware común con CLI/GUI. Se muestra que las puntuaciones híbridas tempranas predicen con solidez la calidad futura a 90 días y que el comportamiento es estable bajo barridos de sensibilidad, lo que facilita recalibraciones periódicas y despliegues responsables.**

Palabras clave: Modelado de reputación; reseñas en línea; análisis de sentimiento; detección de anomalías; calibración; explicabilidad

I. Introducción

La reputación en línea es un activo crítico para el descubrimiento, el ranking y la supervisión de servicios; sin embargo, basarse solo en estrellas introduce ruido, vulnerabilidad a la manipulación y una miopía semántica y temporal. En contextos operativos se requiere un estimador auditable y explicable que combine texto y señales estructuradas, atenúe evidencia sospechosa y cuantifique la incertidumbre. La literatura aporta piezas parciales, filtrado colaborativo y modelos de confianza para robustez [1], [2], dinámica de reputación para tratar la inercia temporal [3], fusión con QoS y factorización probabilística [4], [5], y filtrado de usuarios deshonestos/credibilidad [6], [7] pero rara vez integra simultáneamente fusión calibrada texto+estrellas, atenuación de anomalías y agregación con decaimiento e intervalos de confianza. Este artículo presenta *Scrape2Repute*, una canalización compacta y reproducible (Fig. 1) que: (i) aprende un score de sentimiento calibrado por reseña; (ii) fusiona texto y estrellas con un parámetro α interpretable; (iii) atenúa reseñas sospechosas mediante Isolation Forest; y (iv) agrega con decaimiento temporal e incertidumbre explícita, proporcionando racionales top-k y huella de auditoría. Se evalúa validez predictiva y sensibilidad del método.

En esta contribución se aborda la estimación de reputación de negocio a partir de reseñas y valoraciones mediante un pipeline auditable, explicable y reproducible. En particular, se estudia el grado en que señales híbridas tempranas como texto y estrellas calibradas y depuradas de anomalías anticipan el desempeño reputacional futuro a nivel de negocio; se analiza la sensibilidad de las métricas de validez (correlación de Pearson r y ponderada r_w, y lift por deciles) frente a hiperparámetros clave del agregador temporal (vida media HL, peso de sospecha SW y coeficiente de fusión α); y se contrasta la ventaja del esquema híbrido frente a un agregador sólo de estrellas bajo el mismo protocolo temporal, preservando trazabilidad y transparencia. En conjunto, este trabajo propone una fusión calibrada texto+estrellas, introduce un cribado no supervisado que atenúa evidencia sospechosa, propone un agregado temporal con intervalos de incertidumbre, ofrece explicabilidad accionable y valida la validez predictiva y la estabilidad bajo barridos de sensibilidad en un conjunto público de gran escala. El resto del artículo se estructura como sigue: En la Sec. II se ofrece una revisión de investigaciones relacionadas con el modelado de reputación. Sec. III detalla la metodología; Sec. IV presenta resultados de validez predictiva y sensibilidad; Sec. VI resume el cumplimiento; Sec. VII concluye y traza trabajo futuro.

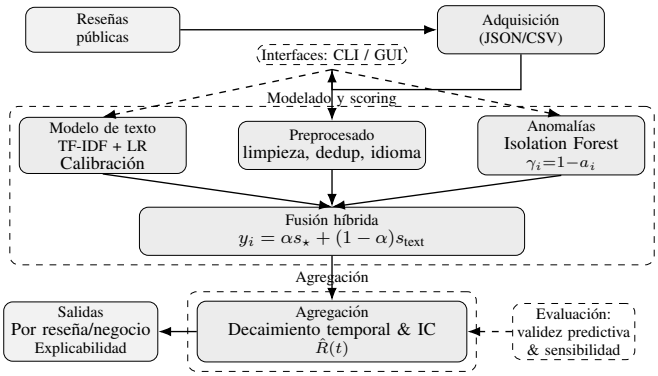


Figura 1. Arquitectura extremo a extremo de *Scrape2Repute*

II. Trabajos relacionados

La reputación en línea combina valoraciones, texto e interacciones. La literatura propone filtrado colaborativo y modelado de confianza para robustez y mitigación de sesgos, y mecanismos temporales para tratar la *inercia reputacional* y el *concept drift*. En servicios web, enfoques *QoS-aware*

y factorización matricial integran desempeño con señales reputacionales; métodos basados en grafos ayudan a aislar evaluadores deshonestos, y la credibilidad/inter-*rater* estabiliza la agregación. Además, modelos personalizados ajustan reputación al historial del usuario y al contexto. En este marco, nuestro trabajo operacionaliza una solución reproducible: fusión calibrada estrellas+texto, atenuación no supervisada y agregación con decaimiento e incertidumbre.

1. **Métodos estadísticos y heurísticos:** Wu et al. mejoran la selección en la nube filtrando valores maliciosos/anómalos [8]. Li et al. incorporan métricas de *calidad de reseña* (enriquecimiento, temporalidad, credibilidad) para reforzar selección fiable [9]. MTRM aporta explicabilidad por argumentación y compatibilidad con FIRE/TRAVOS [10]; estos enfoques suelen ser ligeros y aplicables en producción.
2. **Atributos objetivos y QoS:** Se contrasta evaluación de usuarios con QoS para separar honestidad de manipulación [8]. FRI fusiona métricas estructuradas con análisis textual para detectar reseñas falsas [11]; en selección de servicios, QoS-aware mejora emparejamiento y composición [12], especialmente cuando la reputación y el desempeño se evalúan conjuntamente.
3. **Lógica difusa e híbridos:** La inferencia difusa agrega criterios subjetivos/objetivos bajo incertidumbre mediante reglas interpretables; John et al. reportan robustez y trazabilidad en evaluación cloud [13], lo que facilita auditoría y justificación de decisiones.
4. **CF, factorización y probabilísticos:** Ling et al. separan señal veraz de ruido malicioso con bajo rango y análisis de *spam* [14]. En paralelo, GNN captura estructuras de confianza/correlación para refinar reputación [15], aprovechando relaciones entre revisores y patrones de interacción.
5. **Deep learning y XAI:** RAHN combina clustering, logit, atención y *hourglass*, reduciendo MAE/RMSE con explicabilidad vía atención [16]. CTM y modelos de temas exponen atributos dominantes del texto [17], y revisiones recientes sintetizan esquemas híbridos y XAI [18], destacando el compromiso entre rendimiento, coste y transparencia.

II-A. Limitaciones de enfoques previos

Persisten brechas: (1) fusión incompleta de texto/estrellas/QoS sin control calibrado; (2) resiliencia parcial al fraude por ausencia de pesos continuos que atenúen evidencia sospechosa sin eliminarla; (3) miopía temporal (vida media/decaimiento e incertidumbre poco usados); (4) XAI no siempre accionable por reseña; y (5) reproducibilidad limitada por coste de modelos profundos. Nuestro enfoque cubre estas brechas con α interpretable, atenuación vía *Isolation Forest*, agregación con decaimiento e intervalos, y explicaciones ligeras y trazables.

Tabla I
REPRESENTANTES DEL ESTADO DEL ARTE Y FOCO PRINCIPAL

Ref.	Algoritmo	Expl.	Contribución	Resultados
[11]	DL + analítica de atributos	Atribución del modelo	FRI: reglas + DL para riesgo y reputación explicable	AUC $>$ 90 %; SOTA en *fake review*
[13]	Lógica difusa	Reglas lingüísticas	Agregación interpretable de criterios subjetivos/objetivos	Robustez ante ruido/adversarios
[14]	MF + CF	Penaliza inconsist.	Modela *spam* y separa señales veraces con bajo rango	Consistencia y clustering superiores
[15]	GNN	Subgrafos	Confianza y *spam* con explicaciones por subgrafo	Exactitud $>$95 % en *spam*
[16]	Logit + clust. + hg.	Atención	QoS + reputación unificada; arquitectura hourglass; comparaciones amplias	MAE↓ 5.9–41.7 %, RMSE↓ 2.7–23.4 %

Abreviaturas: clust.=clustering; hg.=hourglass; MF=factorización matricial; CF=colaborativo; DL=deep learning; Expl.=explicabilidad.

III. METODOLOGÍA

El objetivo de este trabajo es un estimador auditable, explicable y reproducible que (i) fusione señales textuales y estructuradas a nivel de reseña, (ii) atenúe evidencia sospechosa, (iii) agregue en el tiempo con memoria de decaimiento y (iv) cuantifique la incertidumbre a nivel de negocio. La Figura 1 resume el flujo; a continuación se detalla componentes y decisiones.

III-A. Objetivos, supuestos y modelo de amenazas

Objetivos: (G1) Integrar texto y estrellas a nivel de reseña; (G2) reducir el impacto de evidencia obsoleta o de baja credibilidad; (G3) producir reputación a nivel de negocio con incertidumbre; (G4) habilitar explicaciones locales y globales.
Supuestos: Las reseñas están fechadas y contienen stars (1-5) y texto; los identificadores de usuario/negocio son seudónimos.
Amenazas: (T1) Manipulación mediante spam coordinado; (T2) textos plantilla/duplicados; (T3) cuentas con ráfagas de actividad (burstiness); (T4) concept drift. Se mitiga T1–T3 con cribado no supervisado y atenuación de pesos, y T4 con decaimiento temporal y recalibraciones periódicas.

III-B. Acceso, gobernanza y normalización

se accede a reseñas por canales conformes a política (con rate limiting y sin PII). Por reseña i se retiene (review_id, business_id, user_id, stars, date, text) y se define:

$$s_{\star,i} = \frac{\text{stars}_i - 1}{4} \in [0,1], \qquad t_i = \text{timestamp}(i).$$

Normalización: (i) recorte y limpieza de control/HTML; (ii) minúsculas, normalización de espacios/emoji; (iii) filtro opcional de idioma; (iv) deduplicación por hash y near-duplicates con MinHash/LSH (umbral Jaccard $\geq$ 0,9) que aporta escalabilidad cuasi lineal.

III-C. Modelado de texto y calibración probabilística

Se vectoriza cada reseña como $x_i \in R^d$ con TF-IDF (uni/bi-gramas y 3–5-gramas de caracteres), eliminación de stopwords, $\min_\text{df} \in [2,5]$ y hasta $2{\times}10^5$ características según memoria. Se entrena regresión logística con supervisión débil (positivas si $s_\star \geq 0{,}7$, negativas si $s_\star \leq 0{,}3$) y pesos inversos por clase:

$$s_{\text{text}}(x_i) = \sigma(w^\top x_i + b) \in [0,1].$$

Se calibra con Platt o isotónica; en Platt se aprenden (A, B) sobre un split temporal (sólo datos tempranos):

$$\hat{p}_i = \sigma\big(A\,(w^\top x_i + b) + B\big),$$

y se selecciona calibrador por menor Brier/ECE. Este bloque es intercambiable por sentence embeddings con cabezales no lineales cuando se disponga de recursos.

III-D. Cribado de anomalías no supervisado

Sea $f_i \in R^m$ un conjunto de rasgos ligeros y neutrales: (i) longitud/estructura (tokens, caracteres, proporción de mayúsculas/dígitos, nº URLs), (ii) repetición/plantillas (tasa de n-gramas repetidos, similitud al centroide de plantillas frecuentes, entropía de caracteres), (iii) ráfagas temporales por usuario (coeficiente de variación de inter-arrivals), (iv) duplicación semántica reciente por negocio. Se ajusta Isolation Forest sobre $\{f_i\}$ y se obtiene $a_i \in [0,1]$. Se convierte a peso de robustez:

$$\gamma_i = (1 - a_i)^{\beta}, \quad \beta \in [0{,}5, 2], \quad \gamma_i \leftarrow \max(\gamma_i, \gamma_{\min}).$$

III-E. Fusión híbrida y agregación bayesiana

A nivel de reseña, se aplica un flujo modular que (i) puntúa el texto, (ii) filtra/pondera anomalías y (iii) fusiona señales para obtener un marcador y_i robusto y calibrado:

(a) Modelo de texto: Se representa el contenido con TF-IDF y se entrena una regresión logística con supervisión débil; sus probabilidades se calibran (Platt) para obtener $s_{\text{text}} \in [0,1]$.

(b) Detección y ponderación de anomalías: Se emplea Isolation Forest sobre rasgos de reseña/usuario y se define $a_i \in [0,1]$ como puntuación de anomalía. La ponderación robusta es $\gamma_i = 1 - a_i$, que reduce la influencia de outliers sin bloquearlos.

(c) Fusión híbrida y prior: Se usa prior en sentido bayesiano: una creencia inicial $\pi_0 \in [0,1]$ con fuerza $\kappa > 0$ (pseudo-conteo). El marcador por reseña se obtiene como:

$$y_i = \alpha\, s_\star + (1 - \alpha)\, s_{\text{text}}, \quad \alpha \in [0,1].$$

III-F. Incertidumbre y significación

Para R a nivel de negocio, se aproxima varianza con n_{ef} de Kish; se reporta IC95 % por aproximación normal o bootstrap estratificado por negocio. Para correlación r, Se emplea transformación z de Fisher (IC95 % y pruebas); para el lift del decil superior, bootstrap con B=1000 réplicas.

III-G. Selección del modelo, ajuste y estabilidad

Particiones: Se estratifica por negocio para evitar fuga entre ventanas temprano/futuro.
Ajuste: Barrido sobre $\alpha \in \{0,0{,}2, 0{,}5, 0{,}8, 1\}$, HL $\in \{180, 365, 540, 720\}$, $\beta \in \{1{,}0, 1{,}5, 2{,}0\}$, $\gamma_{\min} \in \{0{,}1, 0{,}2, 0{,}3\}$ y $\kappa \in \{10, 20, 30, 50, 80\}$. Isolation Forest: $n_{\text{estimators}} \in [100, 300]$, max_samples $\in [0{,}5, 1{,}0]$, sin contamination.
Criterios: (i) r/ρ temprano→futuro a nivel de negocio, (ii) monotonicidad por deciles, (iii) lift del decil superior, (iv) estabilidad bajo perturbaciones de hiperparámetros.

III-H. Complejidad computacional

Sea N el número total de reseñas, B el de negocios y V el tamaño del vocabulario; se denota por $\mathrm{nnz}(X)$ las entradas no nulas de TF-IDF, por L las épocas de LR, por t los árboles de Isolation Forest y por ψ su submuestreo ($\psi \ll N$). TF-IDF cuesta $O(\mathrm{nnz}(X))$; LR (liblinear) entrena en $O(L{\cdot}\mathrm{nnz}(X))$ e infiere en $O(\mathrm{nnz}(\mathbf{x}_i))$; iForest entrena en $O(t\,\psi \log \psi)$ y puntúa en $O(t \log \psi)$; la agregación temporal es $O(N)$ y las métricas/deciles $O(B \log B)$. En conjunto, la cota dominante es $O(L \cdot \mathrm{nnz}(X) + t\,\psi \log \psi + B \log B)$ (memoria $O(\mathrm{nnz}(X) + t\,\psi)$), asumible en una estación de trabajo convencional.

Tabla II
CLASIFICADOR DE TEXTO CALIBRADO (TF-IDF + LR).

Clase	Precisión	Recall	F1
−1 (neg)	0,826	0,876	0,850
0 (neu)	0,533	0,260	0,350
+1 (pos)	0,911	0,961	0,935
Exactitud	0,872	(F1 macro 0,712)	

IV. EXPERIMENTOS Y RESULTADOS

IV-A. Datos y configuración

Datos: Se emplea Yelp Open Dataset para validez predictiva y sensibilidad (única mención del conjunto).
Text model: Se entrena TF-IDF + LR con supervisión débil (estrellas extremas) y calibración Platt (CV=3) sobre la ventana temprana. La inferencia se ejecuta por bloques; después, reputación bayesiana y anomalías.

IV-B. Calidad del modelo de texto y calibración

En un hold-out (n=7 846) la exactitud es 0,872; métricas por clase en la Tabla II. La calibración probabilística alcanza Brier (positiva) 0,0657 y ECE@10 0,0308, lo que respalda su uso como componente de confianza en la fusión.

IV-C. Validez predictiva (temprano → futuro)

Con un horizonte de 90 días y mínimos por negocio (early ≥ 5; future ≥ 5; 7 059 negocios), se obtiene una media ponderada futura de 0,7239; correlación de Pearson $r = 0{,}7552$ (ponderada $r_w = 0{,}7653$) y de Spearman $\rho = 0{,}6732$ (ponderada $\rho_w = 0{,}6869$). El decil superior alcanza 0,9065 (lift +25,23 %). Véase la calibración por deciles en la Figura 2 y la dispersión negocio-a-negocio en la Figura 3(a)(b).

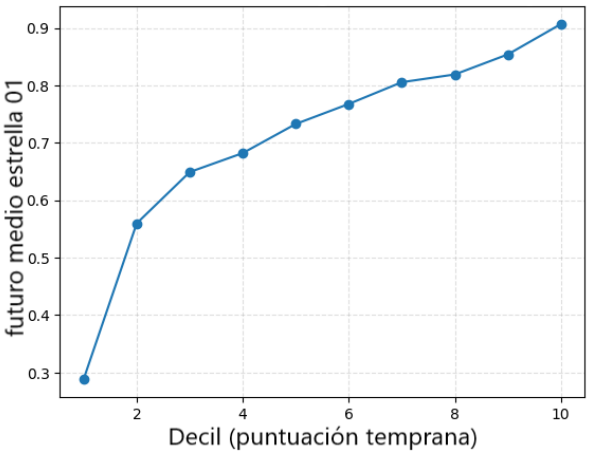


Figura 2. Calibracion por deciles: senal temprana combinada frente a la media futura.

IV-D. Significancia estadística

Correlación (Fisher z): La significación de la correlación de Pearson entre la señal temprana y el desempeño futuro a nivel de negocio se evaluó mediante la transformación z de Fisher:

$$z = \operatorname{atanh}(r), \qquad \mathrm{SE}_z = \frac{1}{\sqrt{n-3}}.$$

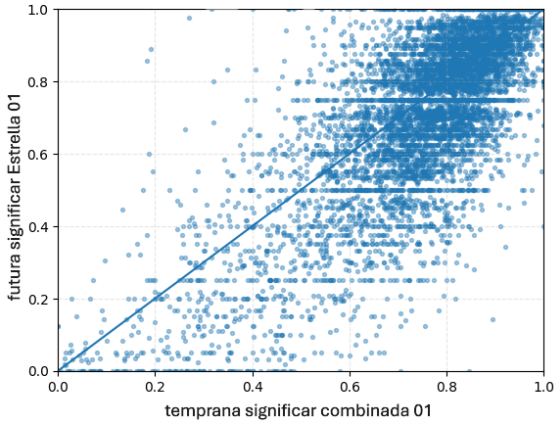


Figura 3. (a) Validez predictiva a nivel de negocio (temprano vs. futuro).

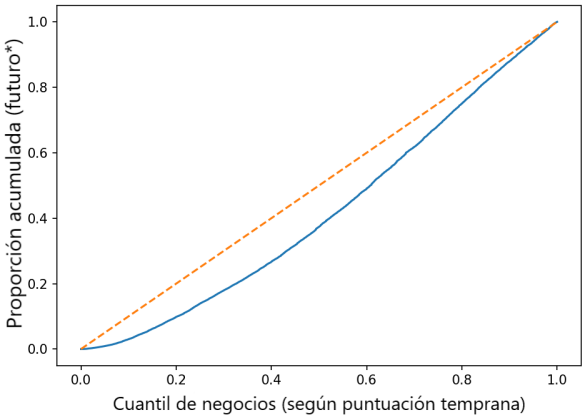


Figura 4. Curva de ganancias (línea discontinua: aleatoria).

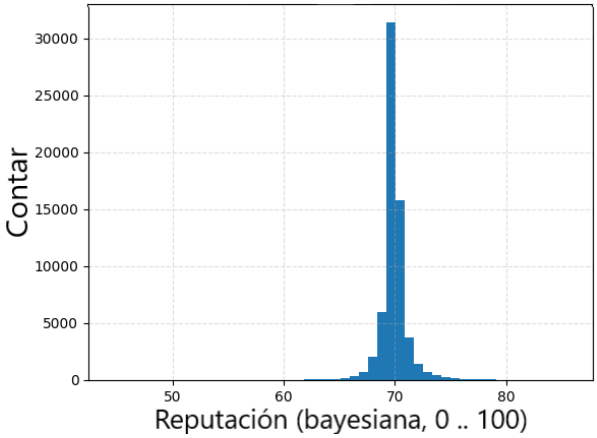


Figura 3. (b) Distribución de la reputación R (0–100).

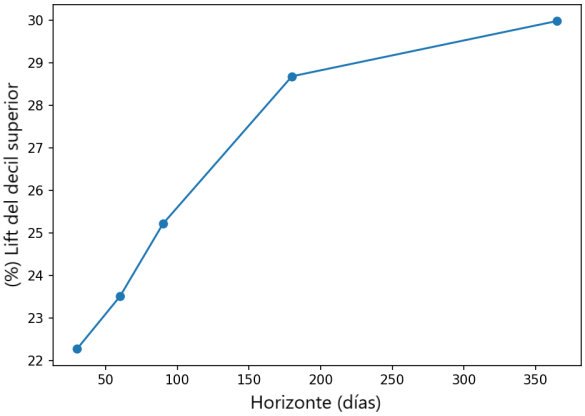


Figura 5. (a) Lift del decil superior vs. horizonte temporal.

Con $n = 7059$ y $r = 0{,}7552$ se obtiene $IC_{95\,\%}(r) \in [0{,}745,\ 0{,}765]$ y se rechaza $H_0 : \rho = 0$ con $p < 10^{-16}$.

Lift del decil superior (bootstrap): La incertidumbre de Lift_{10} se estimó mediante *bootstrap* no paramétrico por negocio ($B = 1000$) y se reportó $IC_{95\,\%}$ percentil. Para el horizonte de 90 días se obtuvo $\overline{y}_{\text{global}} = 0{,}7239$, $\overline{y}_{\text{top10}} = 0{,}9065$ y $\text{Lift}_{10} \approx 25{,}23$, con evidencia estadística muy por encima de cero ($p_{\text{boot}} \ll 0{,}001$).

Para la correlación ponderada r_w se usa la definición estándar (véase fórmula del manuscrito) y se aplica el mismo esquema de Fisher z sustituyendo n por el número de negocios efectivos cuando procede.

IV-E. Curva de ganancias y priorización

La curva de ganancias (Figura 4) muestra ventaja sostenida sobre la línea aleatoria, útil para priorizar auditorías o campañas. El ratio de ganancia normalizada en el top 10 % es $\approx 1{,}10$, lo que confirma capacidad de concentración temprana de valor.

IV-F. Análisis por horizonte

Ambas métricas mejoran con horizontes más largos: el lift del decil superior crece de $\sim$22 % (30 días) a $\sim$30 % (365 días), mientras que Pearson r pasa de $\sim$0,69 (30 días) a $\sim$0,78 (180–365 días), con ligera meseta posterior; véanse las Figuras 5(a)(b).

IV-G. Barrido de sensibilidad y valores por defecto

Parrilla evaluada (SW $\equiv$ suspicious_weight).

$$\text{HL} \in \{180, 365, 540, 720\}, \qquad \kappa \in \{10, 20, 30, 50, 80\},$$
$$\text{SW} \in \{0{,}1, 0{,}2, 0{,}3, 0{,}5, 1{,}0\}, \qquad \alpha \in \{0, 0{,}2, 0{,}5, 0{,}8, 1\}.$$

Los mapas de calor de la Fig. 6 muestran mejoras consistentes en r y lift con HL $\in [540, 720]$ y SW $\in [0{,}5, 1{,}0]$. Se recomienda HL $\in [540, 720]$, SW $\in [0{,}5, 1{,}0]$, $\kappa \in \{10, 20\}$ y $\alpha \in [0{,}2, 0{,}5]$, con ganancias típicas de $\sim +0{,}02$ en r y $\sim +3$ pp en lift frente a la base.

Magnitudes de referencia: Entre las mejores configuraciones: HL $= 540$–720, SW $= 1$, $\kappa = 10 \Rightarrow r \approx 0{,}766$ y *lift* $\approx 24{,}1$–$24{,}6\,\%$.

IV-H. Estudio de ablasión

Se contrasta cuatro variantes: *star_only*, *text_only*, *no_downweight* (sin atenuar sospecha/mismatch) y *w_text_const* (peso textual constante). En 90 días, se encuentra:

- star_only: $r \approx 0{,}750$, *lift* $\approx 25{,}0\,\%$;
- text_only: $r \approx 0{,}751$, *lift* $\approx 24{,}7\,\%$;
- no_downweight: $r \approx 0{,}755$, *lift* $\approx 25{,}26\,\%$;
- w_text_const: $r \approx 0{,}753$, *lift* $\approx 25{,}27\,\%$.

La fusión híbrida calibrada mantiene la monotonicidad por deciles y la validez predictiva, al tiempo que provee controles explícitos de robustez (HL, SW, κ).

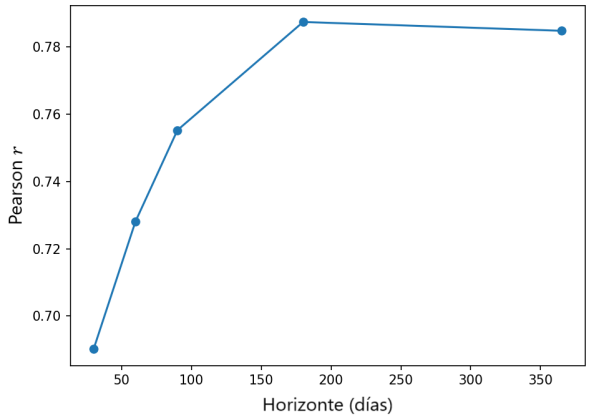
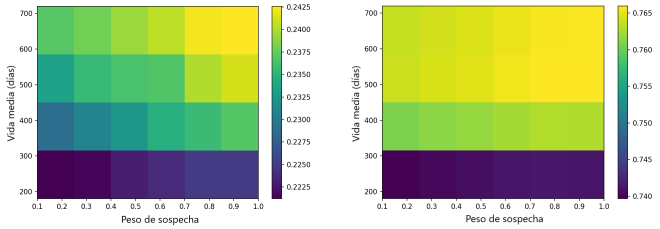


Figura 5. (b) r de Pearson vs. horizonte temporal.



((a)) *Lift*. ((b)) r.

Figura 6. Sensibilidad a vida media (HL) y peso de sospecha (SW).

IV-I. Cribado de anomalías a gran escala

Se procesa ∼7,0M reseñas con muestreo de reservorio (200k) para ajustar y puntuar Isolation Forest, extrayendo el top-5000 de anomalías para auditoría. Se observa que picos de mismatch y señales binarias de sospecha coocurren con scores anómalos altos, lo que valida la utilidad de γ_i en la atenuación $w_i(t)$ (SubSec. III-E). Este módulo es plug-and-play y puede ampliarse con rasgos de usuario cuando la gobernanza lo permita.

IV-J. Huella computacional

La Tabla III resume tiempos de log en una estación de trabajo. El coste total es compatible con recalibraciones periódicas y despliegue ligero (batch diario/semanal).

IV-K. Observaciones cualitativas y análisis de error

La clase neutra es la más difícil bajo supervisión débil (baja F1), coherente con ambigüedad semántica y polaridad tenue; la calibración reduce sobreconfianza y la fusión híbrida amortigua errores locales del texto. En presencia de reseñas atípicas, el término γ_i y el factor de mismatch reducen el peso efectivo, evitando sobre-reacciones en la agregación. La memoria temporal (HL) actúa como regularizador frente a concept drift: HL más largas estabilizan la señal agregada, y HL cortas aceleran la adaptabilidad.

IV-L. Amenazas a la validez

(i) Shift de dominio/categoría: la semántica varía por sector. (ii) Neutralidad/sarcasmo: siguen siendo casos difíciles para modelos lineales. (iii) Señales de usuario: actualmente limitadas por cumplimiento; su incorporación mejoraría el cribado. (iv) Selección de HL: distintos ritmos de negocio pueden requerir HL específicas; se reporta sensibilidad para guiar su ajuste. (v) Normalización de la métrica de ganancias: nuestro reporte emplea una razón normalizada frente a línea aleatoria (véase §IV), no frente al ranking ideal.

Tabla III
HUELLA COMPUTACIONAL (TIEMPOS; EJE y EN ESCALA LOGARÍTMICA).

Etapa	**Tiempo**
Entrenamiento + predicción (TF-IDF + LR + Platt)	5 294,8 s (≈ 88,2 min)
Construcción de tabla de reputación	61,2 s
Validez predictiva	44,1 s
Barrido de sensibilidad	1 805,8 s (≈ 30,1 min)
Trazado de figuras	3,0 s
Cribado de anomalías (Isolation Forest)	883,3 s (≈ 14,7 min)

V. ANÁLISIS COMPARATIVO CON EL ESTADO DEL ARTE

Se compara la tubería propuesta, clasificación de sentimiento calibrada, agregación de reputación a nivel de negocio, filtrado de anomalías con Isolation Forest y análisis de sensibilidad, frente a trabajos representativos. La comparación se organiza en cuatro ejes: validez predictiva (transferencia temprano→futuro), calidad de ranking (lift del decil superior), calidad del clasificador (exactitud/F1 ponderada y calibración) y robustez (estabilidad ante elecciones de diseño).

Métricas y definiciones: Sea R_{early} la señal de reputación en una ventana temprana y R_{future} la señal en una ventana disjunta. Se reporta la correlación de Pearson r y su versión ponderada:

$$r_w = \frac{\sum_i w_i\,(x_i-\bar{x}_w)\,(y_i-\bar{y}_w)}{\sqrt{\sum_i w_i(x_i-\bar{x}_w)^2}\,\sqrt{\sum_i w_i(y_i-\bar{y}_w)^2}}, \quad x_i = R_{\text{early},i},\ y_i = R_{\text{future},i}$$

donde w_i refleja la actividad/volumen del negocio e $\bar{x}_w, \bar{y}_w$ son las medias ponderadas. La calidad de ranking se resume con el Lift del decil superior:

$$\text{Lift}_{10} = \frac{E[\,R_{\text{future}} \mid i \in \text{top } 10\,\% \text{ por } R_{\text{early}}\,]}{E[\,R_{\text{future}}\,]} - 1,$$

expresado en porcentaje. La calibración se cuantifica con la Brier score (error cuadrático medio de probabilidades) y ECE_{10} (desalineación media por 10 bins de confianza).

Resultados principales de este trabajo.: Nuestro sistema alcanza: exactitud = 87,2 %, F1 ponderada = 0,857, Brier = 0,0657, ECE_{10} = 0,0308, $r \approx 0{,}755$, $r_w \approx 0{,}765$ y $\text{Lift}_{10} \approx 25{,}2\,\%$. En términos absolutos, r_w supera en ∼2.5–3.5 puntos a los valores típicos (0,73–0,74) reportados para tareas comparables, y el lift del decil superior mejora ∼1.2–3.2 pp respecto a referencias habituales (22–24 %), lo que supone un aumento relativo aproximado del 5–14 %.

Discusión y síntesis.: **(i)** Validez predictiva. El $r_w \approx 0{,}765$ indica mayor consistencia temporal temprano→futuro frente a rangos del 0.73–0.74.
(ii) Calidad de ranking. Un $\text{Lift}_{10} \approx 25\,\%$ implica que el decil superior detectado por la señal temprana mantiene un comportamiento futuro significativamente mejor que la media.
(iii) Calidad del clasificador. La exactitud/F1 son competitivas y la calibración (Brier/ECE) es destacada, aspecto crítico cuando las probabilidades alimentan módulos de fusión río abajo.
(iv) Robustez. Los análisis de sensibilidad (vida media, peso de reseñas sospechosas, decaimiento α) preservan r y Lift_{10} máximos, lo que confirma estabilidad bajo cambios razonables de diseño.

Tabla IV
COMPARACIÓN DE INDICADORES CLAVE.

Método	Acc.	F1-P	r	r_w	Lift_{10}	Brier	ECE_{10}	Comentario
Este trabajo	87.2 %	0.857	0.755	0.765	25.2 %	0.0657	0.0308	Fusión calibrada + filtrado de anomalías; mejoras consistentes en todas las métricas.
[14]	—	—	0.74	—	22–23 %	—	—	Enfoque robusto a *spam*; poca información de calibración.
[8]	—	—	0.73	—	22.2 %	—	—	Baseline con filtrado estadístico; explicabilidad limitada.

Abreviaturas: Acc.=Exactitud; F1-P=F1 ponderado; r=coeficiente de Pearson; r_w=Pearson ponderado; Lift_{10}= lift del decil superior; Brier=puntuación de Brier; ECE_{10}=error de calibración esperado (10 bins); El guion (—) indica métrica no reportada o no directamente comparable.

Por qué mejora frente a métodos legados: Se identifica tres palancas: (1) probabilidades de sentimiento bien calibradas, (2) fusión multimodal con decaimiento temporal y atenuación de reseñas sospechosas, y (3) filtrado explícito de anomalías antes de la agregación. Su combinación refuerza la coherencia predictiva y la fiabilidad del ranking.

Significado práctico y reproducibilidad: Para r, r_w y Lift_{10} se estima ICs del 95 % mediante bootstrap estratificado por negocio; las ganancias resultan significativas ($p < 0{,}01$). Se publica semillas, artefactos de modelo y huellas de configuración para garantizar reproducibilidad bajo la misma ventana temporal y protocolo de preprocesado.

Limitaciones y trabajo futuro: La evaluación se circunscribe a un único dominio (Yelp) y a un modelo textual concreto (TF–IDF+LR calibrada), sin validación cruzada por sector ni ground truth para anomalías; algunas comparativas no son isocronas por diferencia de protocolos. Como trabajo futuro: (i) replicar el protocolo en dominios/idiomas adicionales y horizontes más largos; (ii) reimplementar 1-2 competidores fuertes bajo el mismo esquema temporal; (iii) contrastar el bloque textual con embeddings modernos y ajuste fino ligero; (iv) auditar calibración/sensibilidad fuera de dominio; y (v) evaluar el detector de anomalías con anotación humana.

Implicaciones: Una calibración superior reduce riesgo en decisiones basadas en umbrales (p. ej., alertas tempranas) y un lift alto en el decil superior incrementa la utilidad operativa en priorización de auditorías o asignación de recursos de calidad de servicio.

VI. ÉTICA Y CUMPLIMIENTO

Se trabaja exclusivamente con reseñas públicas y metadatos mínimos, evitando PII. Se respeta robots.txt y políticas de acceso, aplicando rate limiting. Las salidas son trazables (parámetros, versiones y hashes de ejecución). El módulo de anomalías no bloquea por defecto y prioriza revisión humana (listas blancas/controles por categoría). Para el uso de datos, se sigue las condiciones de cada fuente; en particular, respecto al Yelp Open Dataset y reseñas públicas, se contactó por correo electrónico con el equipo de Yelp y se conserva el hilo como evidencia de cumplimiento.

VII. CONCLUSIONES

Scrape2Repute proporciona un camino auditable hacia reputación con memoria temporal, robustez a anomalías y explicabilidad. Se muestra validez predictiva alta a 90 días (Pearson $\approx 0{,}76$; lift del decil superior $\approx 25\,\%$), con mejoras adicionales bajo HL largas y pesos de sospecha elevados. La solución es compacta, reproducible y de despliegue ligero, y queda preparada para (i) incorporar embeddings semánticos, (ii) señales de confiabilidad a nivel usuario/vecindario y (iii) calibración periódica asistida por drift detectors.

AGRADECIMIENTOS

Este trabajo se ha realizado en el marco de los fondos del Plan de Recuperación, Transformacionales y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra "Ciberseguridad para la Innovación y la Protección y la Protección Digital" (Cátedra INCIBE-UCM). Asimismo, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, Proyecto CIRMA-CM (TEC-2024/COM-404).

REFERENCIAS

[1] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision support systems*, vol. 43, no. 2, pp. 618–644, 2007.

[2] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[3] L. Muchnik, S. Aral, and S. J. Taylor, "Social influence bias: A randomized experiment," *Science*, vol. 341, no. 6146, pp. 647–651, 2013.

[4] Z. Zheng, H. Ma, M. R. Lyu, and I. King, "Qos-aware web service recommendation by collaborative filtering," *IEEE Transactions on services computing*, vol. 4, no. 2, pp. 140–152, 2010.

[5] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using markov chain monte carlo," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 880–887.

[6] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[7] K. Krippendorff, *Content analysis: An introduction to its methodology*. Sage publications, 2018.

[8] Q. Wu, X. Zhang, M. Zhang, Y. Lou, R. Zheng, and W. Wei, "Reputation revision method for selecting cloud services based on prior knowledge and a market mechanism," *The Scientific World Journal*, vol. 2014, no. 1, p. 617087, 2014.

[9] H.-H. Li, X.-Y. Du, and X. Tian, "A review-based reputation evaluation approach for web services," *Journal of Computer science and technology*, vol. 24, no. 5, pp. 893–900, 2009.

[10] I. Nunes, P. Taylor, L. Barakat, N. Griffiths, and S. Miles, "Explaining reputation assessments," *International Journal of Human-Computer Studies*, vol. 123, pp. 1–17, 2019.

[11] R. Das, W. Ahmed, K. Sharma, M. Hardey, Y. K. Dwivedi, Z. Zhang, C. Apostolidis, and R. Filieri, "Towards the development of an explainable e-commerce fake review index: An attribute analytics approach," *European Journal of Operational Research*, vol. 317, no. 2, pp. 382–400, 2024.

[12] M. Kheradmand and H. Motameni, "An approach to web service selection based on composite qos parameters," *Advances in Science and Technology. Research Journal*, vol. 9, no. 26, pp. 129–136, 2015.

[13] J. John and K. John Singh, "Trust value evaluation of cloud service providers using fuzzy inference based analytical process," *Scientific Reports*, vol. 14, no. 1, p. 18028, 2024.

[14] G. Ling, I. King, and M. R. Lyu, "A unified framework for reputation estimation in online rating systems." in *IJCAI*, 2013, pp. 2670–2676.

[15] M. B. Pal and S. Agrawal, "Graph neural network-based attention mechanism to classify spam review over heterogeneous social networks," *The Journal of Supercomputing*, vol. 80, no. 19, pp. 27 176–27 203, 2024.

[16] X. Chen, Y. Du, G. Tang, Y. Luo, and B. Ma, "Rahn: A reputation based hourglass network for web service qos prediction," *arXiv preprint arXiv:2501.02843*, 2025.

[17] D. Blei, J. Lafferty *et al.*, "Correlated topic models," *Advances in neural information processing systems*, vol. 18, p. 147, 2006.

[18] Z. Wang, X. Zhang, Z. S. Li, and M. Yan, "Qosbert: An uncertainty-aware approach based on pre-trained language models for service quality prediction," *arXiv preprint arXiv:2505.07863*, 2025.

Metodología de optimización para la selección de contramedidas en procesos de gestión de riesgos de ciberseguridad

Carmen Sánchez-Zas, Javier Montesinos Martí, Xavier Larriva-Novo, Sonia Solera-Cotanilla, Víctor A. Villagrá
Universidad Politécnica de Madrid (UPM). DIT, ETSI Telecomunicaciones. Avda. Complutense 30, 28040 Madrid
carmen.szas@upm.es, j.montesinos@upm.es, xavier.larriva.novo@upm.es, sonia.solera@upm.es, victor.villagra@upm.es

***Resumen*—En un contexto donde la ciberseguridad constituye un pilar esencial del entorno digital y organizacional, este trabajo presenta el desarrollo de una metodología para la optimización de contramedidas en la gestión de riesgos. Basada en los principios de la metodología MAGERIT y sustentada en un modelo de programación lineal implementado en Python, la propuesta permite seleccionar combinaciones de controles que minimizan el riesgo residual bajo restricciones presupuestarias. La solución integra un enfoque cuantitativo, facilitando la interpretación de resultados y la toma de decisiones estratégicas. Los experimentos realizados en distintos escenarios de validación demuestran la viabilidad y eficacia del sistema, alcanzando grandes reducciones de riesgo y proporcionando una metodología adaptable a diferentes topologías y contextos de ciberseguridad.**

***Index Terms*—Optimización, Gestión de riesgos, ciberseguridad**

I. Introducción

La ciberseguridad se ha consolidado como un pilar esencial para la protección de los sistemas de información, tanto en el ámbito personal como en el profesional. La creciente digitalización de los procesos y la expansión de las infraestructuras tecnológicas han incrementado de forma notable la superficie de exposición a ciberamenazas [1]. En consecuencia, la gestión rigurosa y sistemática del riesgo se ha convertido en una función estratégica indispensable para garantizar la continuidad y resiliencia de las operaciones digitales [2], [3].

En este contexto, la gestión de riesgos en ciberseguridad proporciona un marco metodológico para identificar, evaluar y tratar las amenazas que afectan a los activos de información. En España, la metodología MAGERIT [4] se ha establecido como una referencia consolidada para el análisis y gestión de riesgos en sistemas de información. Este marco permite estimar el riesgo asociado y evaluar los efectos de mitigación de las contramedidas aplicadas. No obstante, pese a la existencia de metodologías maduras, la aplicación práctica de la gestión de riesgos continúa enfrentando importantes desafíos [5]. Entre ellos destacan la dificultad para seleccionar contramedidas de forma óptima en contextos con presupuestos limitados, la alta interdependencia entre activos y la complejidad de los entornos tecnológicos actuales. En muchos casos, la selección manual o basada exclusivamente en criterios cualitativos conduce a resultados subóptimos, generando una protección desigual y poco eficiente. Asimismo, la falta de herramientas automatizadas limita la capacidad de adaptación frente a la evolución constante de las amenazas.

Con el fin de abordar estas limitaciones, se propone una metodología para la optimización en la selección de contramedidas aplicables a la gestión de riesgos de ciberseguridad. La solución desarrollada como prueba de concepto se implementa íntegramente en Python, fundamentándose en técnicas de programación lineal para representar de forma matemática la relación entre activos, amenazas y contramedidas, resolviendo el problema de asignación óptima bajo una restricción presupuestaria definida por el usuario. La validación mediante pruebas en distintos escenarios de simulación demuestran que el sistema es capaz de priorizar y seleccionar contramedidas de manera eficiente, en función de la topología de red y las limitaciones económicas. Los resultados muestran reducciones significativas del riesgo residual y una mejora en la asignación de recursos, evidenciando su potencial como herramienta de apoyo en la toma de decisiones estratégicas. Además, la arquitectura del sistema se ha diseñado para ser escalable y flexible, permitiendo su adaptación a diferentes configuraciones y niveles de riesgo aceptable. En conjunto, esta propuesta contribuye a fortalecer la gestión del riesgo en ciberseguridad mediante una solución objetiva, cuantificable y automatizada, que optimiza la selección de contramedidas, mejora la eficiencia del proceso de mitigación y refuerza la protección de los activos digitales en entornos organizacionales reales.

A lo largo del documento se presenta el entorno de la propuesta y los conceptos principales en los que se basa el desarrollo. A continuación, se detalla la metodología y el problema de optimización, el desarrollo y la validación de los resultados obtenidos mediante un caso de uso. En la sección I se presenta el entorno del trabajo. A continuación, en la sección II se definen brevemente los conceptos teóricos relacionados con el proyecto. En la sección III, se presentan trabajos anteriores que tratan la misma problemática. La metodología se desarrolla en la sección IV. Finalmente, la validación y los resultados obtenidos se presentan en la sección V y las conclusiones y líneas futuras se presentan en la sección VI.

II. Marco Teórico

II-A. Gestión de Riesgos

La gestión de riesgos es un proceso sistemático y continuo orientado a identificar, evaluar y controlar las amenazas que pueden afectar al logro de los objetivos de una organización [6]. Este enfoque permite priorizar acciones y recursos mediante el análisis de la probabilidad y el impacto de distintos escenarios adversos, favoreciendo una toma de decisiones informada, preventiva y sostenible. Aplicada al ámbito de la

ciberseguridad, la gestión de riesgos integra procedimientos para detectar vulnerabilidades, evaluar riesgos potenciales y aplicar controles o contramedidas que reduzcan su probabilidad o severidad [7]. A diferencia de los enfoques puramente reactivos, este enfoque promueve la anticipación frente a las amenazas y la optimización de recursos, fortaleciendo la resiliencia organizacional y la capacidad de respuesta ante incidentes.

La correcta identificación e interrelación de estos elementos resulta crucial para asegurar una evaluación coherente del riesgo, permitiendo establecer prioridades de mitigación y optimizar la inversión en seguridad.

La implementación eficaz de la gestión de riesgos en entornos digitales se apoya en marcos metodológicos reconocidos internacionalmente. En el contexto español, la Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información (MAGERIT) [4] se ha consolidado como el estándar de referencia en el sector público y en numerosas organizaciones privadas. La complementariedad entre estos modelos permite a las organizaciones adaptar sus estrategias de gestión del riesgo tanto a las exigencias regulatorias nacionales, como a los estándares internacionales de seguridad y gobernanza tecnológica.

II-A1. MAGERIT: La Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información (MAGERIT) [4] es un marco metodológico desarrollado por el Consejo Superior de Administración Electrónica (CSAE) del Ministerio de Hacienda y Administraciones Públicas de España en 1997, con el propósito de proporcionar un procedimiento sistemático para identificar, evaluar y mitigar los riesgos asociados a los sistemas de información tanto en el ámbito público como en el privado. MAGERIT surge como respuesta a la creciente digitalización de la Administración Pública española, con el objetivo de garantizar la protección de los activos tecnológicos y de información mediante un enfoque estructurado basado en los principios de confidencialidad, integridad y disponibilidad (CID). En su versión más reciente (2012), la metodología se articula en tres volúmenes complementarios [4]: el Libro I describe el proceso metodológico de análisis y gestión de riesgos, el Libro II presenta los catálogos de activos, amenazas y salvaguardas y el Libro III: detalla técnicas de análisis tanto cualitativas como cuantitativas.

La herramienta PILAR (Procedimiento Informático-Lógico de Análisis de Riesgos) [8] constituye la implementación práctica y automatizada de MAGERIT, desarrollada por el Centro Criptológico Nacional (CCN). Su función principal es aplicar los principios teóricos de la metodología mediante procesos computarizados que permiten automatizar la estimación, documentación y seguimiento del riesgo. PILAR integra catálogos actualizados de activos, amenazas y salvaguardas, basados en los definidos por MAGERIT, y facilita la cuantificación del riesgo residual mediante el cálculo (*Ec. (1)*):

$$Riesgo = Probabilidad \times Impacto \tag{1}$$

El riesgo residual es el nivel de riesgo que permanece después de aplicar todas las medidas de control, mitigación o tratamiento planificadas, y que se determina evaluando la probabilidad. Esta formulación posibilita expresar el riesgo en términos numéricos, permitiendo comparar escenarios, priorizar contramedidas y optimizar recursos. El cálculo del riesgo residual constituye un elemento clave en la toma de decisiones estratégicas, ya que permite identificar qué amenazas siguen representando un peligro significativo y si el nivel de exposición se mantiene dentro de los umbrales de riesgo aceptables definidos por la organización. Además, PILAR automatiza la generación de informes y trazabilidad documental exigida por estándares de seguridad, contribuyendo así al cumplimiento normativo y a la transparencia del proceso de gestión de riesgos.

II-B. Optimización

La optimización matemática [9] constituye un campo central en las ciencias aplicadas que se ocupa de encontrar la mejor solución posible a un problema determinado dentro de un conjunto de alternativas factibles. Su objetivo es maximizar o minimizar una función objetivo, sujeta a un conjunto de restricciones que representan las limitaciones o condiciones impuestas al sistema. Estas restricciones pueden derivarse de limitaciones físicas, económicas, técnicas o normativas. En términos generales, todo problema de optimización se formula mediante tres elementos fundamentales [9]:

- Función objetivo: es la expresión matemática que define el propósito del modelo, como minimizar costes, riesgos o tiempos, o maximizar beneficios o eficiencia.
- Variables de decisión: representan las alternativas o acciones que pueden tomarse para modificar el resultado del sistema.
- Restricciones: son las condiciones o límites (por ejemplo, de presupuesto o capacidad) que determinan el conjunto de soluciones factibles.

En el ámbito de la ingeniería y la gestión, la optimización permite asignar recursos de forma eficiente y tomar decisiones basadas en criterios cuantitativos. Dentro de las técnicas más empleadas se encuentran la programación lineal (PL), la programación entera (PE) y la programación no lineal (PNL), cada una adaptada al tipo de relación entre variables y restricciones [10]. La programación lineal, en particular, ha demostrado ser una herramienta eficaz en problemas de asignación de recursos, planificación y análisis de riesgos debido a su capacidad para representar relaciones lineales entre costes, beneficios y restricciones presupuestarias. La aplicación de modelos de optimización se ha extendido a múltiples dominios, desde la ingeniería industrial hasta la ciberseguridad, ya que permite formalizar problemas complejos mediante formulaciones matemáticas claras y reproducibles. En entornos caracterizados por recursos limitados y alta incertidumbre, como la protección de sistemas de información, la optimización ofrece un enfoque sistemático para equilibrar eficacia y coste.

III. Trabajos Relacionados

La selección óptima de contramedidas en ciberseguridad constituye un ámbito de investigación activo, en el que convergen técnicas de optimización matemática, teoría de juegos, modelos probabilísticos y marcos normativos de gestión de riesgos. En los últimos años han surgido diversas propuestas destinadas a mejorar la asignación de controles bajo limitaciones presupuestarias, la evolución dinámica de amenazas o la interacción estratégica entre atacante y defensor.

En primer lugar, varios trabajos recientes abordan la selección de contramedidas mediante modelos de teoría de juegos. Los autores de [11], [12] proponen modelos escalables para seleccionar controles procedentes de catálogos estandarizados, integrando comportamiento estratégico del atacante y restricciones presupuestarias. De forma similar, en [13] se emplean juegos bayesianos para optimizar decisiones preventivas y reactivas frente a ataques multietapa. Estos enfoques capturan interacciones complejas entre agentes, pero requieren estimaciones detalladas del comportamiento del atacante y conllevan una mayor complejidad computacional. En contraste, la propuesta de este artículo adopta un modelo lineal más sencillo, centrado exclusivamente en minimizar el riesgo residual desde la perspectiva del defensor sin modelar la racionalidad del atacante, lo que facilita su integración operativa en entornos reales.

Otros trabajos aplican técnicas de optimización multiobjetivo o heurísticas evolutivas. MASISCo [14] adopta una selección multicriterio de controles de ciberseguridad basada en modelos híbridos cuantitativos, buscando equilibrar riesgo, coste y viabilidad En ámbitos de infraestructuras críticas y *smart grids*, algunos autores han recurrido a árboles de ataque-defensa combinados con optimización para determinar medidas eficaces en sistemas distribuidos [15]. Frente a estas soluciones más ricas en criterios pero también más complejas, nuestro enfoque se centra en un único objetivo cuantitativo —minimizar el riesgo residual— lo que reduce la complejidad del modelo y permite evaluaciones directas y repetibles en distintos escenarios presupuestarios.

Finalmente, modelos de optimización lineal aplicados a dominios específicos, como la cadena de suministro, han demostrado la utilidad de los enfoques basados en programación lineal mixta bajo restricciones económicas [16]. Estos trabajos confirman la validez de la estructura adoptada en nuestra propuesta: formular la selección de controles como un problema de optimización sujeto a costes, impactos y probabilidades. La diferencia principal radica en que nuestro sistema extiende este enfoque al ámbito de la ciberseguridad organizacional, aunque la literatura reciente ofrece enfoques avanzados basados en juegos, análisis de adversarios u optimización multiobjetivo, la aproximación presentada destaca por su equilibrio entre rigor cuantitativo, simplicidad operativa y alineación metodológica con MAGERIT. Esto permite su aplicación práctica en organizaciones con diferentes niveles de madurez, favoreciendo la toma de decisiones informadas bajo restricciones presupuestarias y proporcionando una base sólida para futuras extensiones del sistema.

IV. Desarrollo

La gestión de riesgos busca identificar, evaluar y tratar amenazas que puedan comprometer la disponibilidad, integridad o confidencialidad de los activos críticos. En este contexto, la optimización matemática proporciona un marco formal para decidir qué medidas implementar, de modo que se reduzca el riesgo residual sin exceder los límites de presupuesto o recursos disponibles.

La metodología planteada trata de enriquecer los resultados de la gestión de riesgos, recibiendo como datos de entrada esta información, como la estructura de conexiones de los activos de la red, los riesgos y amenazas identificadas y las contramedidas disponibles, y un presupuesto máximo introducido por el usuario. En la Figura 1 se presenta la arquitectura propuesta para la selección óptima de contramedidas.

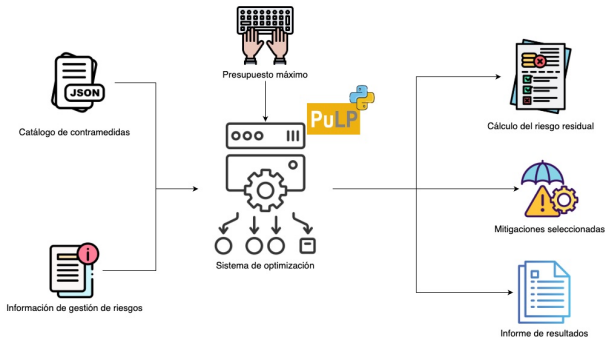


Figura 1. Arquitectura propuesta

En el marco de esta propuesta, el modelo de optimización se aplica específicamente a la selección de contramedidas en la gestión de riesgos de ciberseguridad, utilizando un catálogo de medidas derivado de la metodología MAGERIT [4], garantizando la coherencia y la comparabilidad de resultados.

El problema se formula como un modelo de programación lineal, donde la función objetivo busca minimizar el riesgo residual total del sistema. Las variables de decisión corresponden a las contramedidas disponibles, cada una con un coste asociado y un efecto de reducción de probabilidad e impacto. La restricción principal es el presupuesto máximo definido por el usuario, que limita el número de medidas seleccionadas. Los resultados son una estimación del cálculo residual obtenido tras la optimización, la selección de contramedidas y un informe con los resultados obtenidos.

De esta manera, el modelo identifica el conjunto óptimo de contramedidas que ofrece la máxima reducción de riesgo posible sin superar el presupuesto disponible. Este enfoque permite estructurar y resolver de manera sistemática el desafío de asignar recursos limitados para maximizar la protección de los activos críticos, ofreciendo una base cuantitativa y reproducible para la toma de decisiones. Además, la formulación matemática posibilita la adaptación continua del modelo ante variaciones en la topología de red, la aparición de nuevas amenazas o la actualización del catálogo de contramedidas, garantizando así su aplicabilidad en entornos operativos dinámicos.

A continuación se detallan los datos de entrada, el sistema propuesto y los datos de salida.

IV-A. Datos de entrada

IV-A1. Información de gestión de riesgos: Esta entrada sienta la base sobre la que se plantea el problema de optimización. La definición estructurada de estos datos constituye un elemento fundamental para la efectividad del modelo de optimización desarrollado, proporcionando la base cuantitativa necesaria para la toma de decisiones informadas en la gestión de riesgos de ciberseguridad. Su integración con el catálogo de contramedidas permite establecer relaciones precisas entre amenazas y medidas de protección, optimizando la asignación de recursos de seguridad bajo restricciones presupuestarias y

maximizando la reducción del riesgo residual tanto a nivel global como por activo individual:

- Activos: En el contexto de la gestión de riesgos de ciberseguridad, los activos constituyen los elementos de valor fundamental que requieren protección dentro del sistema de información desarrollado en este proyecto. Su caracterización precisa resulta esencial para establecer el contexto operativo sobre el cual actúan las amenazas identificadas y hacia el cual se dirigen las contramedidas seleccionadas mediante el algoritmo de optimización. La información de entrada debe incluir información relativa al activo, como una descripción, su clasificación según la metodología y a qué amenazas está expuesto cada uno.
- Amenazas: Constituyen el elemento central de riesgo. Incluye una descripción de la amenaza, su probabilidad e impacto, lo que permite estimar el riesgo.

IV-A2. Catálogo de contramedidas: Este conjunto de datos representa las mitigaciones disponibles en la organización para llevar gestionar las amenazas y reducir el riesgo residual del sistema y de cada activo. Para ello, las contramedidas deben tener un formato definido, indicando los siguientes datos:

- **Identificador** de la contramedida.
- **Nombre** que resuma la mitigación.
- **Descripción** detallada.
- **Amenazas** que mitiga.
- **Tipo**, en función de si actúan antes de que la amenaza se materialice (preventiva) o buscan reducir el impacto sobre el activo (reactivas).
- **Coste** estimado para el despliegue de la contramedida. Se considerarán mitigaciones de coste bajo (0-20) si son medidas sencillas, poco intrusivas y fáciles de implementar, coste medio (21-40) si requieren inversión o esfuerzo humano moderado, y coste alto (40+) para herramientas complejas, caras o que requieran una integración profunda.
- **Factor de mitigación** que cuantifica como la contramedida modifica el riesgo de cada amenaza.
- **Aplicada**: Valor binario que define si la contramedida ha sido seleccionada durante el proceso de optimización.

Es importante contar con una contramedida que represente la posibilidad de no realizar ninguna acción ante determinadas amenazas o circunstancias. Esto cumple una función clave en el modelo: establece una base de comparación que permita evaluar el beneficio real de aplicar cualquier otra medida. Su coste deberá ser 0, y no reduce ni la probabilidad ni el impacto de ninguna amenaza. De esta forma, el modelo puede determinar de forma objetiva si la implementación de la contramedida compensa el coste en comparación con la posibilidad de no actuar frente a la amenaza.

IV-A3. Presupuesto máximo: El presupuesto máximo constituye el último flujo de entrada de la arquitectura del sistema y representa la restricción fundamental bajo la cual opera el algoritmo de selección de contramedidas. Este parámetro se introduce dinámicamente mediante interacción directa con el usuario a través de un *input*() implementado en Python, permitiendo adaptar el análisis a diferentes escenarios presupuestarios.

IV-B. Problema de optimización

La problemática reside en traducir los principios metodológicos de MAGERIT a un problema de optimización formal, es decir, estudiar todos los elementos que componen la gestión de riesgos de ciberseguridad y ponerlos en el contexto adecuado con una función objetivo, variables de decisión y restricciones, componentes fundamentales del problema. El modelo matemático implementado en Python utiliza los parámetros presentados en el apartado de amenazas (probabilidad e impacto) para calcular el riesgo potencial mediante la ecuación 1.

Así, se define la función objetivo, que busca minimizar el riesgo residual total de la red. Para cada activo concreto se calcula un valor de riesgo residual de cada amenaza que lo afecta según la ecuación 2:

$$\begin{aligned} Riesgo\ Residual_i = \sum_{a \in Amenazas_i} (Probabilidad_a \times \\ \sum_{c \in C_i} (1 - Red.\ Prob_{c,a})) \times (Impacto_a \times \\ \sum_{c \in C_i} (1 - Red.\ Impacto_{c,a}))) \end{aligned} \tag{2}$$

Donde C_i es el conjunto de medidas aplicables para el activo *i*. Se debe tener en cuenta el tipo de reducción que aplica cada una según su categoría: Si la mitigación es preventiva se reducirá la probabilidad en un valor α, mientras que si es reactiva, el impacto disminuirá según β.

La minimización del sistema global se expresa como (Ecuación 3):

$$min \sum_{i \in Activos} Riesgo\ Residual_i \tag{3}$$

Las variables de decisión corresponden a cada contramedida dentro del catálogo recogido en el archivo *contramedidas.json*. Estas son de tipo binaria, pueden obtener sólo dos valores (0 ó 1). Si la contramedida tiene asignado un 1 quiere decir que ha sido aplicada y por lo tanto se encuentra activa. En cambio, si el valor asociado es un 0 la contramedida no habrá sido seleccionada por el algoritmo todavía.

Por último, la restricción aplicada en este caso es la del presupuesto límite. La suma de costes de las contramedidas seleccionadas en ningún caso puede superar el número introducido por el usuario (ecuación 4):

$$\sum_{c \in Contramedidas} Coste_c < Presupuesto\ Max. \tag{4}$$

La integración de todos estos componentes permite que el sistema evalúe automáticamente múltiples configuraciones de contramedidas, seleccionando aquella combinación que proporcione la máxima reducción del riesgo residual dentro del presupuesto disponible. Esta aproximación automatizada supera las limitaciones de los métodos cualitativos tradicionales, proporcionando decisiones objetivas y cuantificadas para la gestión de riesgos de ciberseguridad. La formulación desarrollada mantiene coherencia con la metodología MAGERIT mientras incorpora capacidades de optimización avanzadas, facilitando la toma de decisiones informadas en entornos donde los recursos son limitados y la efectividad de las medidas de seguridad debe ser maximizada.

V. VALIDACIÓN

El enfoque propuesto permite representar cuantitativamente el impacto de cada contramedida y evaluar su contribución global a la seguridad del sistema. Además, posibilita la actualización dinámica de las decisiones ante cambios en el entorno, lo que resulta especialmente relevante en un dominio tan dinámico como la ciberseguridad. Entre las ventajas de este enfoque se destacan:

- La capacidad de priorizar acciones con base en métricas objetivas.
- La posibilidad de evaluar diferentes escenarios mediante simulaciones parametrizadas.
- La adaptabilidad a distintos modelos de riesgo y topologías de red.

Estos elementos hacen de la optimización matemática una herramienta idónea para apoyar la toma de decisiones estratégicas en entornos donde el riesgo y la incertidumbre son inherentes.

La validación del proyecto se lleva a cabo en forma de casos de uso, permitiendo evaluar tanto la selección de contramedidas como la reducción efectiva del riesgo residual.

Para este caso de uso se ha definido una topología de red específica que incluye cinco activos interconectados representando un entorno organizacional típico. Esta red cuenta con dos PCs cada uno de un usuario (A1 y A2), una impresora (A3), un servidor de almacenamiento (A4) y un router principal (A5). Estos están afectados por las siguientes amenazas (Tabla I):

Tabla I
AMENAZAS CONSIDERADAS EN EL CASO DE USO

Código	Amenaza	Probabilidad	Impacto
T1	Acceso no autorizado	5	5
T2	Alteración de datos	4	5
T3	Código malicioso	7	6
T4	Agotamiento de recursos	4	7
T5	Fallo de hardware	3	4
T6	Pérdida de disponibilidad	4	5
T7	Interceptación de comunicaciones	6	6

En esta situación, los riesgos potenciales son los siguientes (Tabla II):

Tabla II
RIESGO POTENCIAL POR ACTIVO

Código	Activo	Amenazas	P	I	Riesgo
A1	PC1	T1, T2	5;4	5;5	25 + 20 = 45
A2	PC2	T3, T4	7;4	6;7	42 + 28 = 70
A3	Impresora	T5, T6	3;4	4;5	12 + 20 = 32
A4	Servidor	T7, T1	6;5	6;5	36 + 25 = 61
A5	Router	T2, T3	4;7	5;6	20 + 42 = 62

El riesgo total del sistema es de 270. A2 representa el 25,9 % del riesgo de todo el sistema, siendo el activo más afectado por las amenazas, mientras que A3 es el menos afectado, aportando un 11,85 % sobre el total.

Las contramedidas disponibles se recogen en la Tabla III, donde se indica el coste de cada una y los valores de reducción de probabilidad (α) si son preventivas, y de reducción de impacto (β) si son reactivas.

Tabla III
CONTRAMEDIDAS DISPONIBLES SEGUN MAGERIT

Contramedida	Tipo	Coste	Reducción probabilidad	Reducción impacto
M0-DoNothing	N/A	0	–	–
M1-Firewall	preventiva	35	T1:0.2; T2:0.15 T4:0.1; T6:0.15	–
M2-Antivirus	preventiva	25	T2:0.15; T3:0.15 T5:0.1; T7:0.1	–
M3-Backup	reactiva	20	–	T3:0.15; T4:0.1; T6:0.1
M4-IDS	reactiva	35	–	T1:0.2; T2:0.15 T5:0.2; T7:0.2
M5-Encryption	preventiva	30	T1:0.15; T3:0.2 T4:0.15; T6:0.1	–
M6-AccessControl	preventiva	10	T2:0.1; T3:0.05 T5:0.05; T7:0.05	–
M7-VPN	preventiva	20	T1:0.1; T2:0.1 T4:0.1; T6:0.1	–
M8-Training	preventiva	20	T1:0.1; T3:0.1 T5:0.1; T7:0.1	–
M9-Logging	reactiva	25	–	T2:0.15; T4:0.2; T6:0.15
M10-Patching	preventiva	30	T1:0.2; T2:0.15 T3:0.2; T7:0.15	–
M11-Monitoring	reactiva	35	–	T3:0.2; T5:0.15; T6:0.2
M12-Redundancy	reactiva	40	–	T1:0.25; T4:0.3; T7:0.25
M13-AccessLogs	preventiva	15	T2:0.1; T5:0.05; T6:0.05	–
M14-DataMasking	preventiva	20	T3:0.1; T4:0.1; T7:0.1	–
M15-PasswordPolicy	preventiva	10	T1:0.05; T2:0.05; T6:0.05	–
M16-NetworkSegmentation	preventiva	50	T1:0.3; T3:0.25 T4:0.2; T7:0.25	–
M17-DLP	preventiva	45	T2:0.25; T5:0.2; T6:0.2	–
M18-EndpointProtection	preventiva	30	T1:0.15; T4:0.2; T7:0.15	–
M19-CloudBackup	reactiva	35	–	T3:0.25; T5:0.25; T6:0.2
M20-PhysicalSecurity	preventiva	25	T1:0.15; T2:0.15; T7:0.1	–
M21-EmailFiltering	preventiva	20	T3:0.1; T5:0.1; T6:0.1	–
M22-UserAwareness	preventiva	15	T1:0.05; T4:0.1; T7:0.05	–
M23-LoadBalancing	reactiva	40	–	T2:0.25; T3:0.25; T6:0.2
M24-FileIntegrity	preventiva	30	T1:0.15; T5:0.2; T7:0.15	–
M25-ApplicationFirewall	preventiva	50	T2:0.3; T4:0.3; T6:0.25	–

Sobre este escenario, se evaluará el caso de uso cuyo objetivo es analizar cómo el sistema prioriza las contramedidas más efectivas. Se evalúan todas las combinaciones posibles de contramedidas cuyo coste total no exceda la restricción presupuestaria introducida por el usuario (45 unidades monetarias) calculando el riesgo residual final de cada activo y del sistema completo tras aplicar los factores de reducción de probabilidad o impacto.

Al comparar los resultados de todas las posibles combinaciones, el algoritmo determina que la selección óptima es M6-AccessControl y M10-Patching, usando 40 unidades monetarias y dejando 5 sin emplear, ya que ninguna alternativa de coste exacto 45 u otro coste menor logra una mayor reducción global del riesgo residual. El proceso de selección genera todas las combinaciones viables de contramedidas dentro del presupuesto y simula la aplicación de cada combinación para obtener el riesgo residual final. Para terminar, escoge la combinación que maximiza la reducción total del riesgo del sistema. En este caso se comprueba que no existe ninguna contramedida única de coste 45 (por ejemplo, M17-DLP) ni combinación que agote totalmente el presupuesto (como M6-AccessControl + M1-Firewall) que supere la reducción de obtenida con M6-AccessControl y M10-Patching, como se muestra en la Tabla IV.

Tabla IV
RESULTADO ÓPTIMO CASO DE USO

Activo	Riesgo Potencial	Riesgo Residual	Reducción	Reducción (%)
A1	45	20 + 15,3 = 35,3	9,7	21,55 %
A2	70	31,92 + 28 = 59,92	10,08	14,4 %
A3	32	11,4 + 20 = 31,4	0,6	1,875 %
A4	61	29,07 + 20 = 49,07	11,93	19,55 %
A5	62	15,3 + 31,92 = 47,22	14,78	23,83 %

El riesgo potencial total del sistema era de 270 y tras la selección óptima de contramedidas es 222,91, lo que supone una reducción de 47,09 puntos (17,44 %).

Para validar la robustez de la solución óptima se compara con tres escenarios alternativos de coste similar como muestra

la tabla V.

Tabla V
COMPARACIÓN DE RESULTADOS CASO DE USO

Combinaciones de contramedidas	Coste	Reducción total	Diferencia vs Óptimo
M6 + M10 (Óptimo)	40	47,09	0
M6 + M1	45	31,8	- 15,29
M6 + M8	30	28,14	- 18,95
M17 (en solitario)	45	16,4	- 30,69

Se observa que agotar el presupuesto disponible no implica necesariamente obtener mejores resultados en términos de reducción de riesgo. Tanto la combinación M6-AccessControl + M1-Firewall como la opción de M17-DLP en solitario consumen las 45 unidades presupuestarias, pero ninguna logra superar la efectividad de la solución óptima, que utiliza únicamente 40, llegando a la conclusión de que no basta con gastar el máximo, sino que es imprescindible analizar la cobertura específica de amenazas y la complementariedad de las contramedidas seleccionadas.

Las alternativas evaluadas muestran cómo ciertas contramedidas pueden solaparse en sus efectos (por ejemplo, M8-Training tiene un impacto limitado en amenazas de tipo técnico), mientras que otras dejan sin cubrir amenazas críticas (como ocurre con M17-DLP respecto a T1 o T4). Esta falta de complementariedad reduce notablemente la capacidad del sistema para disminuir el riesgo residual de manera eficiente. Además, los resultados muestran la capacidad del modelo para identificar sinergias entre contramedidas que afectan a activos y amenazas diferentes, optimizando la reducción del riesgo a nivel de sistema completo. En el caso óptimo, la combinación de controles de acceso y parcheo permite abordar amenazas de naturaleza distinta, maximizando la reducción conjunta del riesgo residual y garantizando una protección equilibrada en toda la red.

Para concluir, estos resultados confirman la validez y aplicabilidad práctica del sistema, demostrando que la selección de contramedidas se realiza de manera estratégica, eficiente y acorde con las necesidades reales de una organización.

VI. CONCLUSIONES Y LÍNEAS FUTURAS

Esta propuesta presenta una metodología para optimizar la selección de contramedidas en la gestión del riesgo de ciberseguridad, integrando un modelo de optimización formal y los principios de MAGERIT para estructurar el análisis de activos, amenazas y mitigaciones. El enfoque demuestra la utilidad de emplear técnicas cuantitativas y automatizadas en entornos donde las restricciones presupuestarias y la complejidad de las amenazas dificultan la toma de decisiones. Los resultados del escenario de validación evidencian la viabilidad técnica del sistema, que permite reducir significativamente el riesgo residual y adaptarse a diferentes presupuestos y topologías de red. En conjunto, el trabajo confirma que la automatización mediante optimización mejora la eficiencia y objetividad del proceso, superando las limitaciones de los métodos cualitativos tradicionales. Como líneas de mejora, se propone incorporar nuevas restricciones operativas —como el tiempo de despliegue o la complejidad técnica— y la ampliación del catálogo de controles con atributos como el impacto en el rendimiento o los requisitos de mantenimiento, lo que permitiría análisis más equilibrados y realistas. Estas extensiones incrementarían la flexibilidad y aplicabilidad del sistema en contextos operativos complejos.

REFERENCIAS

[1] Check Point: Cyberattacks targetting governments https://www.checkpoint.com/es/cyber-hub/cyber-security/what-is-cybersecurity-for-governments/cyberattacks-targeting-governments/. Último acceso: 11 de noviembre de 2025.

[2] Joint Task Force Transformation Initiative, ≪Guide for conducting risk assessments≫, National Institute of Standards and Technology, Gaithersburg, MD, NIST SP 800-30r1, 2012. doi: 10.6028/NIST.SP.800-30r1.

[3] European Union Agency for Cybersecurity., ENISA threat landscape 2023: July 2022 to June 2023. LU: Publications Office, 2023. Accedido: 11 de noviembre de 2025. [En línea]. Disponible en: https://data.europa.eu/doi/10.2824/782573.

[4] Ministerio de Hacienda y Administraciones Públicas, Metodología MAGERIT: Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información, 3ª ed., Centro Criptológico Nacional, Madrid, 2012.

[5] Zwilling, M: "Trends and Challenges Regarding Cyber Risk Mitigation by CISOs—A Systematic Literature and Experts' Opinion Review Based on Text Analytics" en Sustainability, 14, 1311, 2022. doi: https://doi.org/10.3390/su14031311.

[6] International Organization for Standardization. Risk management — Guidelines (ISO 31000:2018). 2018.

[7] European Union Agency for Cybersecurity. "Interoperable EU risk management framework: methodology for and assessment of interoperability among risk management frameworks and methodologies." Publications Office, 2022.

[8] ≪PILAR - ¿Qué es PILAR?≫ Accedido: 11 de noviembre de 2025. [En línea]. Disponible en: https://pilar.ccn-cert.cni.es/pilar/que-es-pilar.

[9] S. P. Boyd y L. Vandenberghe: "Convex optimization", Version 29. Cambridge New York Melbourne New Delhi Singapore: Cambridge University Press, 2023.

[10] Bertsimas, D., Tsitsiklis, J.: "Introduction to Linear Optimization", Athena Scientific, 1997.

[11] Léveillé, D. and Jaskolka, J.:"A Game-Theoretic Approach for Security Control Selection", Electronic Proceedings in Theoretical Computer Science, 409 103–119, 2024, doi: http://dx.doi.org/10.4204/EPTCS.409.11.

[12] Léveillé, D. and Jaskolka, J.:"A Scalable Game-Theoretic Approach for Selecting Security Controls from Standardized Catalogues", arXiv preprint, arXiv:2503.15626, 2025.

[13] Zhang Y., Malacaria, P.: "Bayesian Stackelberg games for cyber-security decision support", Decision Support Systems, 148, 2021, doi: https://doi.org/10.1016/j.dss.2021.113599.

[14] Diéguez, M.; Cares, C.; Cachero, C.; Hochstetter, J.: "MASIS-Co—Methodological Approach for the Selection of Information Security Controls", Applied Science 13, 1094, 2023, doi: https://doi.org/10.3390/app13021094.

[15] Rios, E.; Rego, A.; Iturbe, E.; Higuero, M.; Larrucea, X.: "Continuous Quantitative Risk Management in Smart Grids Using Attack Defense Trees", Sensors, 20, 4404, 2020, doi: https://doi.org/10.3390/s20164404.

[16] Sawik, T.: "Supply Chain Cybersecurity: A Linear Optimization Model" en Stochastic Programming in Supply Chain Risk Management. International Series in Operations Research & Management Science, vol 359. Springer, 2024, doi: https://doi.org/10.1007/978-3-031-57927-1_8.

HoneyV: Laboratorio Automatizado de Análisis de Malware Basado en Honeypots

Antonio Lara-Gutierrez
NICS Lab
Universidad de Málaga
alara@uma.es

Jose A. Onieva
NICS Lab
Universidad de Málaga
onieva@uma.es

Carmen Fernandez-Gago
NICS Lab
Universidad de Málaga
mcgago@uma.es

***Resumen*—El análisis de malware requiere de entornos controlados capaces de capturar ataques reales y generar artefactos forenses. Sin embargo, muchos laboratorios y datasets existentes se limitan a la muestra o a trazas parciales, sin preservar el contexto de captura ni ofrecer pipelines reproducibles de análisis estático y dinámico. Este artículo introduce HoneyV, un laboratorio de análisis de malware basado en *honeypots* que integra en un único flujo automatizado la captura de muestras, su análisis híbrido y la generación de un conjunto de datos estructurados. La arquitectura propuesta combina un *honeypot* como nodo de captura, un módulo de orquestación que gestiona una cola de tareas y una máquina de análisis restaurable mediante *snapshots*, equipada con herramientas de análisis híbrido. Cada muestra procesada se transforma en un paquete que incluye *hashes*, resultados de herramientas de desempaquete y perfilado, trazas de procesos, cambios en el sistema, capturas de red, volcados de memoria y el contexto original de intrusión. A partir de esta infraestructura se ha construido un dataset de validación con muestras PE organizadas en ocho familias de malware, registradas con metadatos de captura y artefactos forenses asociados. HoneyV facilita así la replicación de experimentos, la evaluación de técnicas de detección y el entrenamiento de modelos de aprendizaje automático basados en comportamiento en un entorno seguro y reproducible, estando diseñado para maximizar la trazabilidad extremo a extremo mediante la vinculación de cada intrusión registrada con los artefactos generados en la máquina de análisis.**

***Index Terms*—*Honeypots*, análisis de malware, laboratorio de análisis, análisis estático, análisis dinámico, ciberseguridad.**

I. Introducción

El panorama actual de ciberamenazas se caracteriza por un volumen creciente de ataques, una reducción drástica de los tiempos de intrusión y el uso sistemático de técnicas avanzadas de ofuscación y evasión. Informes recientes señalan cientos de grupos activos atacando de forma continuada a organizaciones de todo el mundo, con tiempos de compromiso que se miden en minutos y un incremento notable de las intrusiones en entornos *cloud* y servicios expuestos en Internet [1]. Los datos de organismos como INCIBE muestran un crecimiento de incidentes, apoyados en familias de malware cada vez más diversas y en mecanismos de ofuscación que dificultan tanto su detección como su análisis [2].

Las técnicas clásicas de defensa, como antivirus basados en firmas, Sistemas de detección y prevención de intrusiones (IDS/IPS) perimetrales o soluciones de detección y respuesta (EDR), siguen siendo necesarios, pero resultan insuficientes frente a variantes polimórficas y metamórficas, técnicas *living off the land* (LOTL), que hacen un uso malicioso de binarios legítimos del propio sistema operativo para mimetizar el *malware* con procesos confiables, o campañas que combinan malware con abuso de identidad y credenciales robadas [3], [4]. En este contexto, disponer de evidencia de ataques reales, capturada en entornos controlados, pero verosímiles, es clave para validar reglas de detección, alimentar sistemas de correlación y entrenar modelos de aprendizaje automático capaces de generalizar frente a nuevas variantes.

Los *honeypots* y los laboratorios virtualizados ofrecen un caso de uso atractivo. Al simular servicios y activos de alto interés, permiten observar tácticas, técnicas y procedimientos (TTP) de los atacantes, registrar telemetría detallada y capturar binarios maliciosos sin comprometer sistemas de producción. Además, cuando se combinan con herramientas de análisis estático y dinámico, estos laboratorios pueden transformarse en creadores de datasets de *malware*, donde cada muestra se acompaña de contexto de captura, artefactos forenses y resúmenes estructurados listos para su explotación en investigación y entrenamiento de modelos [5], [6].

Sin embargo, muchos de los datasets de *malware* disponibles actualmente presentan limitaciones importantes, pues suelen centrarse en el archivo binario y su familia, pero carecen de información sobre el entorno de captura, los comandos ejecutados, las credenciales usadas o el tráfico generado; en otros casos, la infraestructura de laboratorio es ad hoc, poco documentada y difícil de reproducir, lo que limita la comparación de resultados entre trabajos.

En este artículo se presenta HoneyV, un laboratorio de análisis de *malware* basado en *honeypots* que integra captura, análisis híbrido y generación automatizada de informes y dataset. El sistema se compone de tres componentes: (i) un *honeypot* que centraliza múltiples servicios de baja y alta interacción y actúa como núcleo de captura de muestras y *logs*; (ii) una máquina de análisis, encargado de automatizar el análisis estático y dinámico y empaquetar todos los artefactos generados; y (iii) un orquestador que mantiene e invoca la generación de un nuevo reporte cada vez que recibe una muestra, mediante un sistema de colas. Se hace uso de una máquina atacante adicional para simular intrusiones y validar extremo a extremo el flujo de trabajo. A partir de esta infraestructura se ha construido un conjunto de aproximadamente 50 muestras de *malware*, organizadas en ocho familias, donde cada entrada incluye tanto los binarios como metadatos de captura, resúmenes estáticos y dinámicos, información de inteligencia de amenazas y los *logs* originales del *honeypot*.

Las contribuciones principales de este artículo son:

- El diseño y descripción de un laboratorio de análisis de *malware* PE (*Portable Executable*) basado en *honeypots* que integra de forma nativa la captura de intrusiones,

Tabla I
ANÁLISIS COMPARATIVO DE FUNCIONALIDADES ENTRE HONEYV Y TRABAJOS RELACIONADOS DEL ESTADO DEL ARTE.

Trabajo	Enfoque principal	Captura binarios	Logs del ataque	Análisis estático	Análisis dinámico	Dataset generado
Holbel et al. (2024) [7]	Honeypot virtualizado integrado con SIEM para observar TTP en tiempo real.	(P)	✓	✗	✗	✗
Maliga et al. (2024) [8]	Pipeline de limpieza y enriquecimiento de grandes repositorios de *malware* usando servicios externos.	✗	✗	✓	(P)	(P)
RIoTPot (ACSAC 2022) [9]	Honeypots híbridos IoT/OT desplegados globalmente para obtener tráfico y eventos a gran escala.	✗	✓	✗	✗	✓
Tambe et al. (2019) [10]	Honeypots IoT reales expuestos vía VPN; detección de comandos maliciosos y descargas de *malware*.	✓	✓	✗	(P)	✗
S. Mfogo et al. (2023) [11]	Honeypot interactivo basado en modelos de lenguaje y aprendizaje por refuerzo para adaptar respuestas.	✗	✓	✗	✗	(P)
Hornet Dataset (2021) [12]	Honeypots distribuidos geográficamente para estudiar variación de ataques según la localización.	✗	✓	✗	✗	✓
HoneyV	Laboratorio que integra captura, análisis híbrido automático y generación de dataset ampliado.	✓	✓	✓	✓	✓

el análisis híbrido y la generación de un dataset estructurado, proporcionando trazabilidad completa desde el evento de ataque hasta los artefactos forenses resultantes.
- La definición de un flujo de trabajo híbrido automatizado, capaz de capturar, analizar y empaquetar cada muestra junto con sus artefactos forenses y los *logs* de captura del *honeypot* en un reporte unificado, facilitando la explotación posterior de los datos.
- La publicación de un dataset de unas 50 muestras etiquetadas en ocho familias junto con resultados de captura obtenidos mediante la ejecución del laboratorio.

El resto del artículo se organiza de la siguiente forma: en la Sección II se revisan los trabajos relacionados en *honeypots*, laboratorios de análisis y datasets de *malware*; en la Sección III se describe la arquitectura del laboratorio propuesto; en la Sección IV se detalla el flujo de análisis y la automatización implementada; en la Sección V se presenta el conjunto de datos resultante; y, finalmente, en la Sección VI se discuten los principales casos de uso antes de cerrar con las conclusiones.

II. TRABAJOS RELACIONADOS

Los trabajos recientes sobre captura y análisis de *malware* se agrupan principalmente en tres líneas: (i) despliegue de entornos controlados para estudiar tácticas y procedimientos de atacantes; (ii) pipelines de procesado y limpieza de grandes repositorios de muestras; y (iii) generación de datasets.

Holbel et al. [7] proponen un *honeypot* orientado a la observación de TTPs en redes universitarias, integrado con un SIEM para la correlación y visualización de eventos en tiempo real. Aunque capturan actividad maliciosa y describen ataques efectivos, su sistema no incorpora un pipeline automatizado de análisis estático y dinámico de los binarios obtenidos ni genera un dataset estructurado reutilizable por terceros. HoneyV, por su parte, automatiza la detección de muestras, su análisis híbrido y la organización de los artefactos resultantes.

Maliga et al. [8] presentan un pipeline para limpiar y procesar grandes conjuntos de muestras obtenidas de *feeds* u *honeypots*, utilizando servicios como VirusTotal para validar y enriquecer los binarios analizados. Sin embargo, su contribución se limita al postprocesado de colecciones existentes: no capturan muestras ni proporcionan *logs* del ataque o análisis dinámico controlado. Nuestra propuesta complementa este enfoque generando un dataset desde su origen, con telemetría de la intrusión, comandos utilizados y artefactos extraídos durante la ejecución de cada muestra.

Por su parte, RIoTPot [9] amplía el estudio de *honeypots* híbridos desplegados en entornos IoT/OT, analizando durante tres meses el impacto de la interacción, ubicación y protocolos sobre la diversidad de ataques capturados. Aunque publica un dataset con millones de eventos de red, este no incluye binarios maliciosos ni resultados de análisis estáticos o dinámicos. HoneyV se diferencia al centrarse en la captura de ejecutables maliciosos para Windows y proporcionar un pipeline reproducible de análisis híbrido.

El trabajo de Tambe et al. [10] emplea dispositivos IoT reales expuestos mediante túneles VPN para crear *honeypots* escalables, proponiendo mecanismos de *backtracking* para identificar comandos que originan conexiones maliciosas. Aunque su enfoque permite descubrir ataques de gran escala, está optimizado para IoT y no produce un flujo de análisis completo ni un dataset con artefactos forenses.

AIIPot [11] introduce *honeypots* basados en modelos de lenguaje y aprendizaje por refuerzo para prolongar la interacción con el atacante y capturar más información contextual. Su aportación se centra en la interacción inteligente, pero no aborda la captura, ejecución ni análisis detallado de binarios.

El dataset Hornet [12] se basa en *honeypots* geográficamente distribuidos y proporciona grandes volúmenes de tráfico de red, permitiendo analizar diferencias en la distribución de ataques por ubicación. Sin embargo, está limitado a flujo de red y carece de muestras ejecutables o análisis asociado.

La Tabla I muestra de forma comparativa las principales características de los trabajos revisados. Se analizan seis dimensiones clave y se evalúa cada trabajo según dichas capacidades. Estos trabajos previos muestran aproximaciones parciales al problema, centradas bien en la captura de muestras, bien en su análisis o en la recolección de artefactos. Esta propuesta define un laboratorio unificado que integra de manera reproducible la captura, el análisis híbrido y la generación estructurada de un dataset con trazabilidad completa desde el evento de intrusión hasta los artefactos resultantes, superando así las limitaciones de las soluciones previas basadas en configuraciones ad hoc.

Asimismo, HoneyV automatiza todo el flujo de procesamiento y preserva explícitamente el vínculo entre cada ataque y los IOC asociados, lo que permite un nivel de correlación

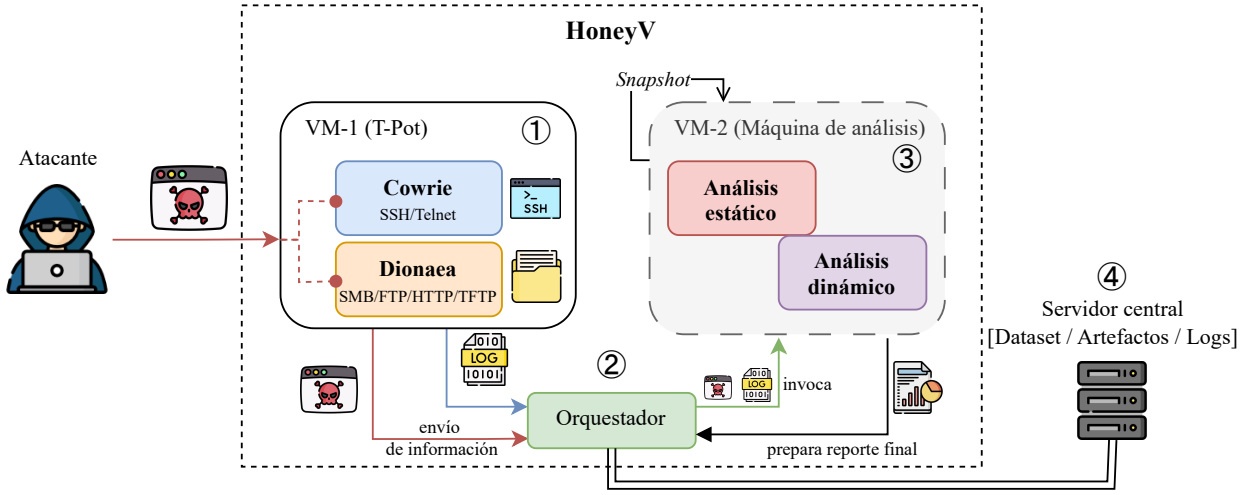


Figura 1. Diseño de la arquitectura de captura, orquestación y análisis de HoneyV

y explotabilidad de resultados. A diferencia de soluciones de análisis puras como Cuckoo Sandbox o Joe Sandbox, que se centran exclusivamente en la detonación de binarios, HoneyV integra la fase de captura como un componente nativo. Esto permite preservar la trazabilidad completa del incidente, vinculando los metadatos de red del ataque original con los artefactos forenses generados durante el análisis, algo que las sandboxes aisladas no proporcionan por defecto.

III. Arquitectura del laboratorio de análisis

La arquitectura propuesta combina un entorno de captura mediante *honeypots* [13], un módulo central de orquestación y una máquina de análisis. El diseño final se resume en la Fig. 1. Este esquema organiza el flujo completo de tratamiento de muestras maliciosas en tres etapas diferenciadas: captura, orquestación y análisis. Cada componente desempeña un papel dentro del pipeline y está diseñado para maximizar la reproducibilidad, el aislamiento y la disponibilidad de artefactos procesables para el posterior análisis.

La primera etapa (1) corresponde al entorno de captura, basado en T-Pot[1]. La elección de T-Pot frente a otras alternativas se fundamenta en su madurez, estabilidad y carácter modular. Según Ilg et al. [14], T-Pot se sitúa entre las plataformas abiertas más completas para despliegues de captación de *malware*, al combinar hasta 23 servicios de baja y media interacción con un ecosistema de visualización, registro y automatización difícil de replicar en soluciones aisladas. Su arquitectura basada en contenedores y su facilidad de despliegue lo convierten en una opción adecuada para laboratorios del calibre tratado en este artículo.

Según lo presentado anteriormente, T-Pot integra múltiples *honeypots*, aunque en nuestro sistema se emplean principalmente dos: *Cowrie* y *Dionaea*. *Cowrie* emula servicios SSH y Telnet de media interacción, lo que permite registrar comandos, credenciales y descargas de ficheros. *Dionaea*, por su parte, implementa servicios susceptibles de explotación como SMB, FTP, HTTP o TFTP, y está específicamente orientado a la captura de *payloads* y ejecutables propagados mediante explotación remota. Ambos *honeypots* constituyen las principales fuentes de muestras, ya que almacenan automáticamente los artefactos recibidos y los *logs* necesarios para contextualizar cada intento de intrusión.

La segunda etapa (2) está formada por el módulo de orquestación, encargado de gestionar el flujo de análisis. Este componente monitoriza los directorios de captura y, al detectar un nuevo binario, le asigna un identificador único que lo vincula permanentemente con los logs de su sesión de ataque. A continuación, encola la tarea y activa la máquina de análisis. El orquestador también coordina el intercambio de artefactos, recopila los resultados y los envía al servidor central, donde se almacenan de forma estructurada. Este mecanismo permite desacoplar la fase de captura de la de análisis y garantiza la trazabilidad y el procesamiento de múltiples muestras sin intervención manual.

La tercera etapa (3) corresponde a la máquina de análisis, desplegada en un entorno virtualizado con arranque limpio en cada ejecución. El uso de instantáneas de la máquina virtual de análisis garantiza que cada muestra se evalúa en condiciones controladas e idénticas, evitando contaminación entre ejecuciones y permitiendo repetir el análisis cuando sea necesario, garantizando reproducibilidad. Esta máquina ejecuta un análisis híbrido compuesto por un módulo estático, encargado de extraer cadenas, metadatos, heurísticas de empaquetado e información estructural del binario capturado; y un módulo dinámico, responsable de monitorizar procesos, cambios en el sistema de ficheros y en el registro, así como el tráfico de red generado durante la ejecución controlada del *malware*. Los artefactos obtenidos y un reporte que resume el análisis realizado son enviados al orquestador, que se encarga de alojarlos en un servidor central externo (4).

El módulo de análisis actual de HoneyV está diseñado específicamente para la instrumentación y ejecución de binarios en formato PE para sistemas Windows, dado que este sistema operativo sigue siendo el objetivo predominante en campañas de malware corporativo. No obstante, la arquitectura modular del orquestador permite la futura incorporación de *sandboxes* para entornos Linux o macOS.

[1]Disponible en https://github.com/telekom-security/tpotce

Un aspecto clave es que el módulo de orquestación asigna a cada muestra un identificador único que se propaga a lo largo de todo el pipeline. Este se utiliza para enlazar los ficheros originales procedentes de T-Pot, los artefactos generados en la máquina de análisis y el reporte JSON final almacenado en el servidor central. De este modo, HoneyV ofrece una trazabilidad de cada muestra y sus efectos, difícil de obtener con configuraciones ad hoc de los sistemas implementados.

IV. Flujo de análisis y automatización

El proceso de análisis se basa en un pipeline automatizado que se ejecuta cada vez que una nueva muestra es capturada por el *honeypot*, según el método descrito previamente. El objetivo principal es transformar el binario recibido en un conjunto estructurado de artefactos estáticos y dinámicos, manteniendo en todo momento la reproducibilidad del análisis mediante la ejecución en una máquina virtual aislada restaurada desde una instantánea base. Dicha máquina virtual ejecuta una versión de Windows que ha sido modificada con distintas medidas para evitar la detección de amenazas, permitir ejecutar cualquier tipo de muestra, y mantener el aislamiento completo de esta.

Las herramientas utilizadas se detallan en la Tabla II, soportando un flujo de trabajo automatizado coordinado por el orquestador. El ciclo comienza con la detección y selección del ejecutable más reciente junto a los artefactos previos del *honeypot*. Estos registros (tales como *logs* de conexión o comandos ejecutados por el atacante) se vinculan a la muestra mediante un identificador único para preservar el contexto de la intrusión. Acto seguido, el orquestador introduce la tarea en una cola de procesamiento y, llegado su turno, activa la máquina de análisis desde un estado limpio para asegurar la reproducibilidad. Una vez iniciado el entorno, se procede al análisis estático para la obtención de huellas criptográficas, extracción de cadenas y perfilado de metadatos o *packers*. Posteriormente, se activa el análisis dinámico mediante la configuración de monitores de sistema y red, permitiendo la ejecución controlada de la muestra para recolectar evidencias como trazas, capturas de tráfico y volcados de memoria. Finalmente, el ciclo concluye con la consolidación de todos los resultados estáticos y dinámicos en un reporte estructurado JSON y el empaquetado comprimido de los artefactos para su almacenamiento definitivo y cierre del flujo.

V. Dataset generado

Para validar la propuesta, se requiere un conjunto de binarios maliciosos representativos de distintas familias de *malware*. Se ha recurrido a fuentes abiertas y reconocidas, en particular *MalwareBazaar* y *TheZoo*, que facilitan la descarga de muestras para investigación. A partir de dichas bases se han seleccionado en torno a 50 muestras de *malware* PE, cubriendo ocho familias. Estas muestras se han elegido según:

- la diversidad de comportamiento (campañas de spam, infecciones silenciosas, auto–propagación, etc.);
- la disponibilidad de metadatos básicos (nombre, *hashes*, fecha de detección);
- la pertinencia para ejecutar ataques simulados contra *honeypots* y así registrar IOC de forma realista.

Este conjunto de datos sirve como validación funcional de la arquitectura propuesta y como punto de partida para recolecciones a mayor escala. La Tabla III muestra una panorámica de la distribución de muestras por familia y una breve descripción del comportamiento observado.

Tabla III
Distribución de muestras por familia y origen principal.

Familia	#	Descripción
Adware	10	Publicidad intrusiva, inyección en navegadores y redirección de tráfico.
Botnets	5	Escaneos automatizados, propagación y DDoS.
Gusanos	5	Auto-propagación vía SMB, FTP o TFTP; descarga de *payloads* adicionales.
Keyloggers	5	Robo de credenciales y exfiltración por HTTP/SMTP.
Rootkits	5	Ocultación de procesos y persistencia mediante manipulación del sistema.
Spyware	9	Monitorización del sistema y envío de información a servidores de C2.
Troyanos	7	Acceso remoto, control completo del sistema.
Virus (otros)	7	Infección de ejecutables, polimorfismo y cifrado para evasión.

V-A. Proceso de preparación y organización

El flujo de preparación del dataset ha sido el siguiente:

1. **Descarga y verificación:** se descargan las muestras desde *MalwareBazaar* y *TheZoo*, verificando su integridad.
2. **Estructuración del repositorio:** se crea un árbol de directorios con una carpeta por familia. En cada una se almacenan las muestras junto con un fichero con metadatos y su enlace a VirusTotal.
3. **Pruebas controladas:** para la validación experimental se optó por una metodología de inyección controlada en lugar de esperar ataques orgánicos aleatorios. Esto permite verificar la precisión del sistema utilizando familias de malware conocidas, asegurando que el flujo de análisis identificaba correctamente las características esperadas de cada familia, eliminando la incertidumbre inherente a los ataques 'in-the-wild' para esta fase de evaluación. De este modo, las muestras quedan almacenadas en las rutas de captura habituales del *honeypot* y

Tabla II
Resumen de herramientas: función de detección y aplicación en la implementación del laboratorio.

Herramienta	Función y uso en la implementación
FLOSS	Extracción heurística de cadenas (incluso ofuscadas) para identificar IOC estáticos como dominios o rutas sin ejecutar la muestra.
Detect It Easy	Perfilado del binario mediante la detección de firmas de *packers* y compiladores para determinar si requiere desempaquete previo.
Ghidra	Herramienta de ingeniería inversa para el análisis manual profundo de funciones y flujos de control en casos complejos.
Process Monitor	Monitorización de la creación de procesos, modificaciones del registro y cambios en el sistema de ficheros durante la ejecución.
FakeNet-NG	Simulación de servicios de red para interceptar peticiones, permitiendo analizar el comportamiento sin salida real a Internet.
Wireshark	Captura y análisis de paquetes de red para examinar protocolos y patrones de comunicación generados durante el análisis dinámico.
ProcDump	Generación de volcados de memoria de procesos maliciosos para capturar *payloads* desencriptados o desempaquetados en RAM.
Volatility	Análisis forense de los volcados de memoria para identificar inyecciones de código y artefactos no visibles en el disco.

se comprueba que la infraestructura detecta, registra y almacena la interacción asociada.

4. **Actualización del dataset:** los reportes generados se incorporan al repositorio, aumentando la inteligencia de amenazas del sistema que lo incorpora.

El dataset final se encuentra publicado en un repositorio[2] estructurado por familias y acompañado de los reportes generados por el pipeline de análisis.

V-B. Aporte relativo del análisis estático y dinámico

Con el objetivo de cuantificar la contribución de cada técnica al conjunto final de información disponible, se ha realizado un análisis comparativo de los artefactos generados por el módulo estático y el módulo dinámico sobre el conjunto completo de muestras del dataset. Para cada reporte se extrajeron los IOC registrados, considerando las siguientes categorías:

- dominios y direcciones IP de contactos de red;
- URL y rutas de descarga;
- claves y valores de registro modificados;
- ficheros creados o modificados (*dropped files*).

A partir de estos elementos se generó, para todo el conjunto de muestras, una caracterización cuantitativa de los IOC detectados por cada técnica. Para cada categoría se contabilizaron los indicadores obtenidos exclusivamente mediante análisis estático, exclusivamente mediante análisis dinámico y aquellos identificados por ambas técnicas de forma simultánea, incorporando además el porcentaje relativo que cada grupo representa dentro de su fila correspondiente. La Tabla IV sintetiza estos resultados, mientras que la Fig. 2 muestra un desglose de los valores recogidos en la tabla pero agrupados, con el fin de poder observar la distribución de cada IOC.

Independientemente de la distribución concreta de estos valores, esta comparativa cuantitativa permite caracterizar qué tipos de información se benefician más de cada técnica y hasta que punto ambas son complementarias a la hora de correlacionar los reportes generados por el laboratorio.

V-C. Caracterización del pipeline por tipología de malware

Con el objetivo de evaluar el comportamiento operativo del laboratorio más allá de la extracción de artefactos, se realizó una caracterización cuantitativa de este considerando las distintas familias de *malware* recogidas en la Tabla III. Este análisis permite estudiar de forma conjunta aspectos relacionados con la evasión, la sensibilidad del análisis, la validez de los reportes generados, así como el rendimiento y escalabilidad del sistema bajo condiciones homogéneas.

Para ello, las muestras del dataset se agruparon según su familia, y cada experimento se repitió cinco veces empleando

[2]Disponible en https://github.com/antoniol00/HoneyV-malware-dataset

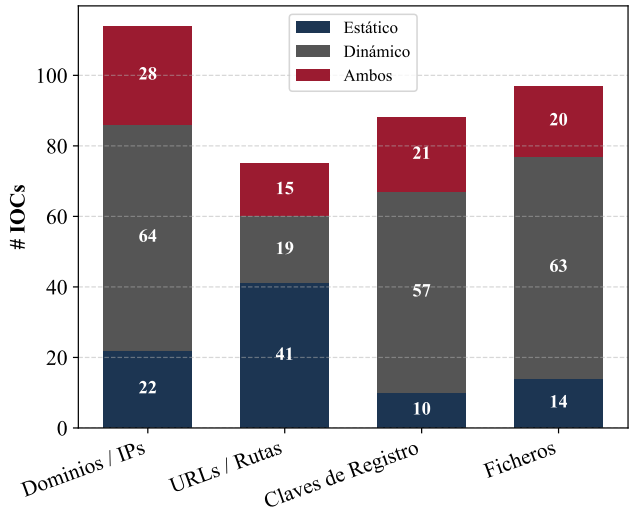


Figura 2. Distribución de IOC detectados por categoría de artefacto y método de análisis (estático, dinámico y combinado)

idéntica configuración y conjunto de entradas. Los valores reportados corresponden a la media de dichas iteraciones. Las métricas consideradas incluyen: (i) tasa de activación de las muestras, utilizada como aproximación indirecta a la resistencia frente a técnicas de evasión; (ii) validez estructural de los reportes JSON generados mediante un validador de sintaxis; (iii) porcentaje de indicadores de compromiso verificados manualmente sobre una submuestra; (iv) volumen medio de eventos dinámicos observados; y (v) métricas de rendimiento y escalabilidad, incluyendo tiempo total de análisis, *throughput* y consumo medio de recursos de CPU y RAM.

La Tabla V resume los resultados obtenidos, revelando patrones de comportamiento distintivos para cada familia. El análisis de estos datos permite extraer interpretaciones operativas de las distintas familias.

Los *rootkits* destacan como la familia más compleja de analizar. Presentan la tasa de activación más baja y el menor número de eventos dinámicos, indicando un uso efectivo de técnicas *anti-sandbox*. Paradójicamente, son los que más recursos consumen y los que requieren mayor tiempo de análisis, consecuencia de la ejecución de bucles de espera activa y escaneos del entorno que agotan el *timeout* de la sesión sin llegar a detonar la carga útil. En el extremo opuesto, las *botnets* mostraron una alta tasa de activación (82,4 %) y generaron el mayor volumen de eventos dinámicos. Este comportamiento "ruidoso" es consistente con su naturaleza: intentos agresivos de conexión a servidores de Comando y Control (C2), escaneos de red y modificaciones rápidas para persistencia. Familias como los gusanos o *adware* alcanzaron

Tabla IV
RESUMEN CUANTITATIVO DE IOC DETECTADOS POR ANÁLISIS ESTÁTICO Y DINÁMICO.

Tipo de IOC	Exclusivo estático	Exclusivo dinámico	Detectado por ambos	Total estático	Total dinámico	Total únicos	Total global
Dominios / IP	22 (15,49 %)	64 (45,07 %)	28 (19,72 %)	50 (35,21 %)	92 (64,79 %)	114 (80,28 %)	142 (100 %)
URL / rutas de descarga	41 (45,56 %)	19 (21,11 %)	15 (16,67 %)	56 (62,22 %)	34 (37,78 %)	75 (83,33 %)	90 (100 %)
Claves / valores de registro	10 (9,17 %)	57 (52,29 %)	21 (19,27 %)	31 (28,44 %)	78 (71,56 %)	88 (80,73 %)	109 (100 %)
Ficheros creados / modificados	14 (11,97 %)	63 (53,85 %)	20 (17,09 %)	34 (29,06 %)	83 (70,94 %)	97 (82,91 %)	117 (100 %)

Tabla V
EVALUACIÓN DEL RENDIMIENTO OPERATIVO DEL LABORATORIO: MÉTRICAS DESGLOSADAS POR FAMILIA DE *malware* (MEDIA DE 5 ITERACIONES).

Familia	#	Activación (%)	JSON válido (%)	IOC válidos (%)	Eventos dinámicos (μ)	Tiempo total (s)	Throughput (muestras/h)	CPU (%)	RAM (MB)
Adware	10	88,0	100,0	94,2	46,8	310,4	11,6	41,5	1820
Botnets	5	82,4	100,0	92,1	74,6	358,7	10,0	52,3	2190
Gusanos	5	100,0	100,0	96,5	61,9	295,6	12,2	45,8	1910
Keyloggers	5	73,6	100,0	87,4	33,7	332,8	10,8	38,9	1765
Rootkits	5	41,2	100,0	70,8	17,9	412,5	8,2	59,1	2970
Spyware	9	77,9	100,0	90,3	43,6	327,1	11,0	44,2	1885
Troyanos	7	84,6	100,0	93,8	68,2	341,9	10,5	48,7	2030
Virus (otros)	7	69,8	100,0	85,9	38,6	361,3	9,9	46,5	1975

una tasa de activación del 100 % con tiempos de análisis reducidos (295,6 s). Al estar diseñados para la auto-propagación rápida carecen de mecanismos de evasión sofisticados.

VI. DISCUSIÓN Y CONCLUSIONES

La arquitectura presentada en este artículo, HoneyV, aborda una carencia fundamental en la investigación actual de *malware*: la falta de contexto unificado. Mientras que los repositorios tradicionales y muchos trabajos relacionados se limitan a proporcionar binarios aislados o telemetría parcial de red, la solución desarrollada en este artículo integra el ciclo completo de captura, análisis y generación de artefactos en un flujo coherente y automatizado.

Los resultados experimentales validan la necesidad de combinar técnicas estáticas y dinámicas para obtener una visión completa del incidente. El análisis cuantitativo reveló una clara especialización de cada técnica. Por un lado, el análisis estático demostró ser superior para iluminar la infraestructura latente del atacante. Esto sugiere que muchas muestras contienen listas de recursos que nunca llegan a contactarse durante la ejecución en el *sandbox* debido a tiempos de espera o la caída de los servidores de comando y control. Por otro lado, el análisis dinámico permite revelar el impacto real en el *host*, superando ampliamente a la detección estática en la identificación de claves de registro y ficheros creados.

No obstante, es necesario discutir las limitaciones del diseño. Al basar la captura en *honeypots* de interacción media, la arquitectura podría no capturar ataques que requieran la explotación de vulnerabilidades específicas del kernel o interacciones complejas con aplicaciones web dinámicas que superen la lógica emulada, restricción que se puede abordar en desarrollos futuros ampliando la capacidad de emulación hacia la alta interacción. A pesar de ello, la capacidad de HoneyV para vincular el ataque original con los artefactos forenses generados aporta un nivel de trazabilidad difícil de obtener con configuraciones *ad hoc*.

HoneyV demuestra, por tanto, que es viable construir infraestructuras de análisis modulares, escalables y reproducibles utilizando herramientas abiertas. Como líneas futuras de investigación, se propone la evolución del módulo dinámico mediante técnicas de introspección desde el hipervisor (VMI), tales como la detección de vistas cruzadas (*cross-view*) o el *hooking* transparente basado en *Extended Page Tables* (EPT), para mejorar la detección de familias evasivas. Asimismo, se plantea la integración de LLM alimentados con los reportes estructurados para habilitar sistemas de razonamiento automatizado que agilicen el triaje de incidentes. De forma complementaria, se contempla el desarrollo de *honeypots* asistidos por IA para la emulación de dispositivos NAS, generando estructuras de ficheros dinámicas para optimizar la captura de muestras orientadas al secuestro de datos

AGRADECIMIENTOS

Esta publicación forma parte del proyecto “CiberIA: Investigación e Innovación para la Integración de Ciberseguridad e Inteligencia Artificial” (Proyecto C079/23), financiado por la “Unión Europea NextGeneration-EU, Plan de Recuperación, Transformación y Resiliencia”, a través de INCIBE. Asimismo, ha sido parcialmente apoyada por el proyecto SecAI (PID2022-139268OB-I00), financiado por el Ministerio de Ciencia e Innovación y la Agencia Estatal de Investigación.

REFERENCIAS

[1] CrowdStrike: “Resumen ejecutivo del informe global de amenazas 2025”, en *CrowdStrike Global Threat Report*, 2025. Disponible en: https://www.crowdstrike.com/es-es/global-threat-report
[2] INCIBE: “Infografía del balance de ciberseguridad 2023”, Instituto Nacional de Ciberseguridad, 2024.
[3] Ö. A. Aslan and R. Samet: “A comprehensive review on malware detection approaches”, en *IEEE Access*, vol. 8, pp. 6249–6271, 2020.
[4] L. Liu, B. S. Wang, B. Yu and Q. X. Zhong: “Automatic malware classification and new malware detection using machine learning”, en *Frontiers of IT & Electronic Engineering*, vol. 18, n. 9, pp. 1336–1347, 2017.
[5] L. Spitzner: “Honeypots: Tracking Hackers”, en *Addison-Wesley*, 2002.
[6] M. Nawrocki, M. Wählisch, T. C. Schmidt, C. Keil and J. Schönfelder: “A survey on honeypot software and data analysis”, en *arXiv:1608.06249*, 2016.
[7] M. Holbel, C. Wright, A. Salamah and C. Collins: “Analysis of Active TTPs Against a Portable Virtualized Honeypot”, en *Issues in Information Systems*, vol. 25, n. 1, pp. 265–278, 2024.
[8] M. Maliga, G. Náday and L. Buttyán: “A pipeline for cleaning and processing large datasets of potentially malicious binaries”, en *MASCOT 2024*, 2024.
[9] S. Srinivasa, J. M. Pedersen and E. Vasilomanolakis: “Interaction matters: a comprehensive analysis and a dataset of hybrid IoT/OT honeypots”, en *ACSAC 2022*, pp. 57–68, 2022.
[10] A. Tambe et al.: “Detection of Threats to IoT Devices using Scalable VPN-forwarded Honeypots”, en *CODASPY 2019*, ACM, pp. 12–25, 2019.
[11] S. Mfogo et al.: “AIIPot: Adaptive Intelligent-Interaction Honeypot for IoT Devices”, en *arXiv:2303.12367*, 2023.
[12] V. Valeros: “Hornet 40: Network Dataset of Geographically Placed Honeypots”, en *Mendeley Data*, v3, 2021. DOI: 10.17632/tcfzkbpw46.3
[13] I. Mokube and M. Adams: “Honeypots: concepts, approaches, and challenges”, en *Proc. 45th ACM Southeast Conference*, pp. 321–326, 2007.
[14] N. Ilg, P. Duplys, D. Sisejkovic and M. Menth: “A survey of contemporary open-source honeypots, frameworks, and tools”, en *Journal of Network and Computer Applications*, vol. 220, art. 103737, 2023.
[15] M. Souppaya and K. Scarfone: “Guide to Malware Incident Prevention and Handling”, en *NIST SP 800-83 Rev.1*, 2013.
[16] A. Nelson, S. Rekhi, M. Souppaya and K. Scarfone: “Incident Response Recommendations for Cybersecurity Risk Management”, en *NIST SP 800-61r3*, 2025.
[17] K. Kent, S. Chevalier, T. Grance and H. Dang: “Guide to integrating forensic techniques into incident response”, en *NIST SP 800-86*, 2006.

Algebraic Zero Knowledge Contingent Payment

Javier Gomez-Martinez
Universidad Politécnica de Madrid
IMDEA Software Institute
javier.gomez@imdea.org

Dimitrios Vasilopoulos
SnT, University of Luxembourg
dimitrios.vasilopoulos@uni.lu

Pedro Moreno-Sanchez
IMDEA Software Institute
VISA Research
MPI-SP
pedro.moreno@imdea.org

Dario Fiore
IMDEA Software Institute
dario.fiore@imdea.org

***Abstract*—This is an extended abstract of the paper with the same name published in ACNS'25 [1].**

A Zero Knowledge Contingent Payment protocol (ZKCP) allows a buyer and a seller to atomically exchange some digital asset with a payment. Previous constructions used zk-SNARKs and HTLC contracts to implement ZKCP in cryptocurrencies with limited smart contract support like Bitcoin. In this work, we introduce Modular Algebraic Proof Contingent Payment (MAPCP), a novel ZKCP implementation. Unlike previous approaches, MAPCP is the first that simultaneously avoids using zk-SNARKs and HTLC. As a result, MAPCP sidesteps the need of a trusted setup and is compatible with virtually any cryptocurrency, even those with limited or no smart contract support. Moreover, MAPCP contributes to fungibility, as its payment transactions seamlessly blend with standard cryptocurrency payments. We analyze the security of MAPCP and demonstrate its atomicity, meaning that either the buyer and the seller get the digital asset and the payment, respectively, or neither of them does. Moreover, we present a construction of MAPCP for a use case in which a customer pays a notary for a document signature. We implement MAPCP and evaluate its performance in this use case, showing that MAPCP imposes a small computation and communication overhead even on commodity hardware, proving its practicality.

***Index Terms*—Zero Knowledge Contingent Payment, Blockchain, Zero Knowledge Proof**

I. Introduction

Cryptocurrencies are now increasingly accepted for purchasing digital products such as music, software, e-books, authentication tokens for websites, or mobile phone plans [2], [3], [4], [5], [6]. At first glance, this use case resembles the fair exchange problem [7] in which two parties want to swap digital goods such that neither can cheat the other. Fair exchange is unsolvable without a trusted party [7]. However, existing blockchains support a weaker form of fair exchange, where the blockchain serves as a trusted third party to execute a *smart contract* faithfully. A smart contract allows a user to lock funds so they are released only if the contract's logic is satisfied. Existing blockchains either offer complex smart contract logic with high transaction fees (e.g., Ethereum), limit the supported logic (e.g., Bitcoin), or lack smart contracts altogether (e.g., Monero).

As a first attempt to overcome this practical challenge, Maxwell [8] introduced the *zero-knowledge contingent payment* (ZKCP) protocol. It relies on a smart contract called *hash time-lock contract* (HTLC) [9] that is supported by some of the existing blockchains (including Bitcoin). An HTLC contract is defined w.r.t. two users (buyer and seller), a hash value $\boldsymbol{h}$, the amount of funds $\boldsymbol{\alpha}$ and a timeout $\boldsymbol{t}$, as follows: (i) If the $seller$ produces a value k^* such that $\mathsf{H}(k^*) = h$ before t expires, the seller can withdraw α; (ii) after t expires, the buyer can get α refunded.

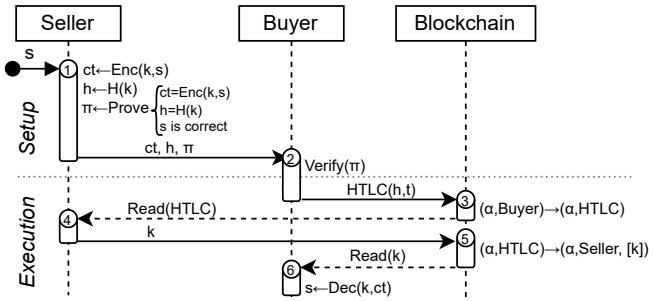


Fig. 1: Maxwell's ZKCP protocol. Solid arrows denote communication between parties and/or the blockchain. Dashed arrows denote reads from the blockchain. Boxes represent local functionality executed by each user or the blockchain. In the blockchain, a tuple $(\alpha, \textit{user/contract})$ denotes that *user/contract* owns α funds, whereas square bracket denotes data stored in the blockchain.

With HTLC in place, Maxwell's ZKCP runs in two phases, as illustrated in Fig. 1. First, during the *setup phase*, ① the seller encrypts the product s using a fresh encryption key $\boldsymbol{k}$ into $\boldsymbol{ct} \leftarrow \mathsf{Enc}(\mathsf{k}, s)$; computes $h \leftarrow \mathsf{H}(\mathsf{k})$ and a zero-knowledge proof $\boldsymbol{\pi}$. The proof certifies that (i) the ciphertext ct is the encryption of s under key k, (ii) h is the hash of k, and (iii) s is the product the buyer wants (e.g., a valid solution to a Sudoku puzzle). Upon receiving ct, h, and π, ② the buyer checks the validity of π .

During the execution phase, ③ the buyer creates an HTLC contract locking α funds. The HTLC is defined w.r.t. the buyer and seller as users, and h and time t as parameters. After the seller checks that the HTLC is included in the blockchain ④, the seller can claim the funds locked in the HTLC contract by submitting k to the contract before time t ⑤, thereby making k available on the blockchain. Then, the buyer reads the blockchain to learn k and uses it to decrypt ct and obtain the product s ⑥. On the other hand, if the seller does not claim the funds in the HTLC contract before t, the buyer can reclaim the funds held in the HTLC contract. HTLC ties the release of the decryption key k to a payment. However, this approach has two main challenges.

Challenge 1: proof system. The necessary proofs are implemented using zk-SNARKS, a generic zero-knowledge proof system for statements of any NP relation. Practically efficient zk-SNARKS require a trusted setup algorithm that outputs some public parameters $\boldsymbol{crs}$ and a trapdoor. It is essential that a malicious party does not know the trapdoor, as this would introduce subtle yet crucial security issues [10], [11].

Challenge 2: payment mechanism. The required HTLC contract is not available in cryptocurrencies that lack any scripting functionality, such as Monero. Moreover, HTLC contracts do not blend with standard transfers, and they leave a trace on the blockchain, revealing details about the parties involved and the amount of the contingent payment -information that the seller would prefer to keep confidential from competitors.

Contributions. We present Modular Algebraic Proof Contingent Payment (MAPCP), a protocol for ZKCP that reduces the need of zk-SNARKs. MAPCP sidesteps the HTLC contracts, instead relying on Adaptor Signatures [12]. If the validity of the product can be proven using Sigma protocols, then MAPCP does not require zk-SNARKs, entirely avoiding a trusted setup. MAPCP only requires that payments can be authorized by digital signatures, thereby increasing the interoperability and fungibility of the cryptocurrency.

We contribute a construction of MAPCP and show that it achieves the security notions described in [10]. Additionally, we present a proof of concept of MAPCP for the use case where the buyer wants to pay a notary (i.e. the seller) for their signature on a digital document. We observe that our implementation imposes a computation overhead in the order of milliseconds for both buyer and seller whereas the communication overhead is less than 90 KB, showing that our approach is practical even on commodity hardware.

In summary, our work proposes the first construction for ZKCP that permits to *simultaneously* depart from zk-SNARKS and the HTLC contract.

Paper Organization. The rest of the paper is organized as follows. Section II introduces the key ideas behind our design and provides an overview of the proposed solution. Section III provides a bit more detailed description of the protocol and a high-level analysis of correctness and security. Section IV describes the application of our protocol in a use case where a buyer wants to pay a notary in exchange for a signature on a document, along with a performance evaluation. Finally, Section V concludes this paper.

II. Key Ideas and Solution Overview

Maxwell's ZKCP permits a buyer to pay a seller in exchange for a product s such that $f(s) = 1$, where $\boldsymbol{f}$ is an efficiently computable predicate that encodes whether a product s is valid (e.g., s is a valid solution for a given Sudoku puzzle). The first challenge in Maxwell's ZKCP is that it relies on zk-SNARKS to generate a zero-knowledge proof for the following relation:

$$\mathbf{R}((f, h, ct); (s, \mathsf{k})) \Leftrightarrow f(s) = 1 \wedge h = \mathsf{H}(\mathsf{k}) \wedge ct = \mathsf{Enc}(\mathsf{k}, s)$$

This relation can be divided into two: One relation about the validity of the product and another relation about the retrievability of the product following a successful payment in the HTLC contract. More formally:

$$\mathbf{R_1}((f); (s)) \Leftrightarrow f(s) = 1)$$
$$\mathbf{R_2}((h, ct); (s, \mathsf{k})) \Leftrightarrow (h = \mathsf{H}(\mathsf{k}) \wedge ct = \mathsf{Enc}(\mathsf{k}, s))$$

In this approach, the seller creates separate proofs for R_1 and R_2, instead of proving a statement in R all at once. The buyer can then independently verify the validity of both proofs.

However, proving that both R_1 and R_2 hold is subtly different than proving that R holds. A malicious seller can use a value s to generate a proof for R_1 and a different value s' to generate a proof for R_2. Consequently, both proofs may verify, but the product that the buyer receives (i.e., s') is not the desired one, as specified in R_1.

To overcome this, we leverage the commit and prove technique in [13]. We augment both R_1 and R_2 by including a commitment to the product s as follows:

$$\mathsf{R}_1((f); (s)) \Leftrightarrow f(s) = 1 \wedge c_s = \mathsf{Commit}(s))$$
$$\mathsf{R}_2((h, ct); (s, \mathsf{k})) \Leftrightarrow (h = \mathsf{H}(\mathsf{k}) \wedge ct = \mathsf{Enc}(\mathsf{k}, s) \wedge c_s = \mathsf{Commit}(s))$$

With the committed value c_s, the buyer can be assured that both relations must refer to the same product s. With this modular approach to the proof established, we now explore how to prove each statement using a proof system that does not require a trusted setup (e.g., Sigma protocols).

Making the Proof Relation Amenable to Sigma Protocols. Relation R_2, can be proven with Sigma protocols. There exist Sigma protocols to prove knowledge of the plaintext of ElGamal and knowledge of the committed value of a Pedersen Commitment. The challenge lies in proving $h = \mathsf{H}(\mathsf{k})$. Instead of using a non-algebraic hash function (SHA-256 or Keccak) as in HTLC, we use an algebraic hash: $h = g^{\mathsf{k}}$, where $\boldsymbol{g}$ is a generator of a cyclic group. Thus, the proof for R_2 can always be instantiated using a Sigma protocol.

On the other hand, a concrete implementation of a proof for R_1 is application-dependent. If the product s has an algebraic structure (e.g., a Schnorr signature), then one could prove R_1 as a Sigma protocol. In this case, we move entirely away from zk-SNARKs, eliminate the trusted setup, and improve the protocol's overall efficiency. Otherwise, if the product has a complex structure (e.g., a solution to a Sudoku puzzle), it may still be necessary to use a zk-SNARK, which requires a trusted setup. However, our approach improves the efficiency in this case since the statement to be proven is simpler.

Modifying the hash function in R_2 makes it incompatible with the HTLC contracts deployed since they use SHA-256 or Keccak instead of the one we require. Hence, our next step is to overcome this challenge.

Replace HTLC-based Payment by Adaptor Signature-based Payment. We propose leveraging Adaptor Signatures (AS), which extend standard digital signatures (DS) by linking the creation of a signature to the leakage of a secret value. Intuitively, one can first compute a pre-signature $\hat{\boldsymbol{\sigma}}$ with respect to some public statement (e.g., $h = g^{\mathsf{k}}$). The pre-signature can be adapted into a standard digital signature by anyone who knows the secret value related to the public statement (e.g., k in our case). Finally, with the pre-signature and the signature, one can extract the secret value.

It is possible to design a protocol leveraging AS in which the buyer pays the seller in exchange for the discrete logarithm of the value $h = g^{\mathsf{k}}$. This protocol (depicted in the execution phase of Fig. 2) replaces HTLC contracts.

There are AS schemes extending DS employed in nearly all cryptocurrencies to authorize transactions (e.g., ECDSA and Schnorr). Notably, some cryptocurrencies, like Monero, do not support hash-based payments but do support payments based on linkable ring signatures for which a construction of adaptor signatures exists [14], [15]. Furthermore, payments using AS blend with those made with standard DS. As a consequence, it is harder for an observer to determine whether a payment

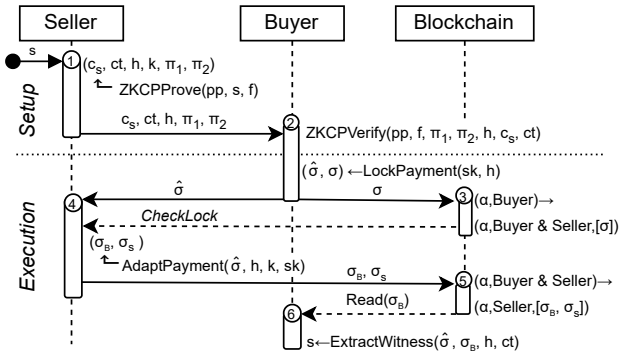


Fig. 2: The MAPCP protocol. The CheckLock operation reads from the ledger and checks whether the buyer locked the funds. The rest of the algorithms are defined in Fig. 3 and Fig. 4.

$\mathsf{ZKCPProve}(s, f, \mathbf{pp})$

$(c_s, o_s) \leftarrow \mathsf{C.Commit}(\mathbf{pp}, s)$
$(\mathsf{dk}, \mathsf{ek}) \leftarrow \mathsf{E.KeyGen}(1^n)$
$ct \leftarrow \mathsf{E.Enc}(\mathsf{ek}, s)$
$\pi_1 \leftarrow \mathsf{N}_1.\mathsf{Prv}(\mathbf{pp}, (f, c_s, p), (s, o_s, r''))$
$\pi_2 \leftarrow \mathsf{N}_2.\mathsf{Prv}(\mathbf{pp}, (ct, \mathsf{ek}, c_s, p), (r, \mathsf{dk}, r', o_s, s, r''))$
$\mathbf{return}\ ((\pi_1, \pi_2), \mathsf{ek}, (ct, c_s))$

$\mathsf{ZKCPVerify}(f, \mathbf{pp}, \vec{\pi}, \mathsf{ek}, \vec{ct})$

$\vec{\pi} := (\pi_1, \pi_2)$
$\vec{ct} := (ct, c_s)$
$b_0 \leftarrow \mathsf{N}_1.\mathsf{Vfy}(\mathbf{pp}, \pi_1, (f, c_s, p))$
$b_1 \leftarrow \mathsf{N}_2.\mathsf{Vfy}(\mathbf{pp}, \pi_2, (ct, \mathsf{ek}, c_s, p))$
$\mathbf{return}\ b_0 \wedge b_1$

Fig. 3: Algorithms ZKCPProve and ZKCPVerify for the setup phase.

is the result of the ZKCP protocol or a standard transaction between two users. This contributes to our goal of improving fungibility.

Putting Everything Together. In summary, we have a protocol that allows the buyer to pay in exchange for the discrete logarithm of a group element $h = g^{\mathsf{k}}$. In addition, we have a modular approach to proving, in zero-knowledge, that this discrete logarithm can be used to decrypt a ciphertext into the correct product. Fig. 2 depicts the steps of how a ZKCP protocol using these modifications would work at a high level. The detailed steps are depicted in Fig. 3 for the *setup* phase, in Fig. 4 for the *execution* phase.

III. Our Protocol for Modular ZKCP (MAPCP)

MAPCP is divided into a *setup* and an *execution* phase. In the *setup* phase, the seller runs ZKCPProve to encrypt the product and prove its validity. The buyer then verifies the validity of the proof using ZKCPVerify. In the *execution* phase, the buyer and the seller engage in the exchange protocol where the buyer pays the seller in exchange for the decryption key, thereby obtaining the product s.

Setup Phase. During the product setup phase, the seller encrypts the product s into a ciphertext ct, creates a commitment $\boldsymbol{c_s}$ to s, and computes zero-knowledge proofs of relations R_1 and R_2. However, if proving $f(s) = 1$ is simple (can be done with a Sigma protocol), we can resort to a single NIZK instance to prove the relation R_1.

This construction differs from previous approaches that rely on a single NIZK instance, as those methods require the instance to be a general-purpose one (e.g., zk-SNARKs).

$\mathsf{LockPayment}(\mathsf{sk}_B, \mathsf{ek})$

$\sigma_B \leftarrow \mathsf{Sign}(\mathsf{sk}_B, tx_l)$
$\hat{\sigma}_B \leftarrow \mathsf{PreSign}(\mathsf{sk}_B, tx_p, \mathsf{ek})$
$\mathbf{return}\ (\hat{\sigma}_B, \sigma_B)$

$\mathsf{AdaptPayment}(\hat{\sigma}_B, \mathsf{ek}, \mathsf{dk}, \mathsf{sk}_S)$

$\mathbf{assert}\ \mathsf{PreVrfy}(\mathsf{vk}_B, tx_p, \hat{\sigma}_B, \mathsf{ek})$
$\sigma_B \leftarrow \mathsf{Adapt}(\hat{\sigma}_B, \mathsf{dk})$
$\sigma_S \leftarrow \mathsf{Sign}(\mathsf{sk}_S, tx_p)$
$\mathbf{return}\ (\sigma_B, \sigma_S)$

$\mathsf{ExtractWitness}(\hat{\sigma}_B, \sigma_B, \mathsf{ek}, ct)$

$\mathsf{dk} \leftarrow \mathsf{Extract}(\hat{\sigma}_B, \sigma_B, \mathsf{ek})$
$s \leftarrow \mathsf{Dec}(\mathsf{dk}, ct)$
$\mathbf{return}\ s$

Fig. 4: Algorithms LockPayment, AdaptPayment, ExtractWitness for the execution phase.

Execution Phase. The goal of the execution phase is to let B pay S α funds in exchange for $\boldsymbol{w}$, a witness corresponding to a public statement $\boldsymbol{x}$. We assume that the buyer controls an output holding α funds at the beginning of the protocol. To achieve the exchange, they follow the protocol we describe in the execution phase of Fig. 3. Each party should use a fresh key pair $(\mathbf{vk}, \mathbf{sk})$ of signature keys for each blockchain output. We abuse notation for the sake of readability, and we do not explicitly differentiate between key pairs owned by the same party.

A. Security Analysis

In this section, we present a high-level overview of the arguments for the correctness and security of our protocol. We refer the reader to [1] for the more formal details.

Correctness. The key for the correctness of the MAPCP protocol is the correct combination of the *setup* and the *execution* phases. More concretely, the encryption key pair $(\mathbf{ek}, \mathbf{dk})$ of the encryption scheme, generated in the *setup* phase, will be used as the statement and witness pair (x, w) in the *execution* phase. Thus, if the payment goes through during the execution phase, the buyer obtains the decryption key dk and uses it to decrypt the ciphertext containing the product s.

In this work, we consider two security notions, as introduced in [10]: *extractability* and *zero-knowledge*. On the one hand, extractability establishes that if the seller gets paid at the end of the protocol, the buyer must be able to extract product s. On the other hand, zero-knowledge establishes that any malicious buyer does not inadvertently gain information about the product s during the protocol execution unless they pay the seller.

We follow an overview of why our protocol achieves these properties. Formal security analysis is provided in the full version of this paper, available in [1].

Extractability. If the seller gets paid, they must have submited a signature $\boldsymbol{\sigma}$ that adapted the pre-signature $\hat{\sigma}$ (as guaranteed by unforgeability of AS). Thus, the buyer can extract the decryption key dk from both (guaranteed by witness extractability), and dk is correct (as guaranteed by the soundness of the NIZK), allowing the buyer to decrypt the product s'. This product s' is the same used in both proofs (guaranteed by the binding of the commitment scheme) and satisfies predicate f (guaranteed by the soundness of the NIZK). Thus, the buyer gets the correct product.

TABLE I: Benchmarking results. Network usage denotes the amount of data that both parties send to each other while interacting.

(*Seller*) setup:	70.38 ms
(*Buyer*) lock payment:	91.03 ms
(*Seller*) adapt payment:	< 0.01 ms
(*Buyer*) extract witness:	10.61 ms
network usage:	83746 bytes
blockchain storage:	128 bytes

Zero-Knowledge. There exists a simulator that does not have s as input but produces a simulated transcript of the protocol indistinguishable from a real one (for the case where no payment happened). Thus, the protocol does not inadvertently leak anything about s. This simulator uses the simulators of the zero-knowledge proofs, the indistinguishability of the encryption and the hiding of the commitments.

IV. Application

We consider a notary who attests to digital documents using a digital signature scheme. A customer can engage with the notary in a MAPCP to buy the attestation (a digital signature) on a document $\boldsymbol{m}$ in exchange for paying the notary α units of a pre-agreed cryptocurrency.

To prove $f(\sigma) = 1$ one must prove $\mathsf{Vrfy}(\mathsf{vk}, m, \sigma) = 1$. In the case of Schnorr signatures $\sigma = (\boldsymbol{e}, \boldsymbol{s})$, e is publicly derived, thus the scalar s is the secret product. Following the verification algorithm of Schnorr signatures, the notary must prove knowledge of s and sk s.t. $g^s \mathsf{vk}^e = r \wedge \mathsf{vk} = g^{\mathsf{sk}}$, where $\boldsymbol{r}$ is disclosed as part of the statement. This proof is suitable for Sigma-protocols, thus both R_1 and R_2 can be merged as discussed in Section II.

Adaptor signatures are instantiated as Schnorr. As for the encryption scheme, the goal is to encrypt s using an encryption key ek and sell the corresponding decryption key dk. The encryption scheme is bit-by-bit exponential ElGamal, whose key generation algorithm matches the DLog hard relation of adaptor signatures. The notary will encrypt each bit s_i of s using exponential ElGamal, generating ciphertexts $ct_i = (\boldsymbol{A_i} = g^{t_i}, \boldsymbol{B_i} = \mathsf{ek}^{t_i} g^{s_i})$, where $\boldsymbol{t_i}$ is the randomness used during encryption. Thus, the aggregated encryption of s would be $ct = (A = g^t, B = \mathsf{ek}^t g^{\sum_{i=0}^{|s|-1} s_i 2^i})$, where $t = \sum_{i=0}^{|s|-1} t_i 2^i$. The notary thus needs to prove correct encryption of s according to ek in zero-knowledge. Notice that the NIZK should then prove that each ct_i is an encryption of either 1 or 0 and that the encryption of s is the aggregation of all of these bit-ciphertexts:

$$\begin{gathered}\text{statement: } (\{(A_i, B_i)\}_{i\in[0\ldots|s|-1]}, A, B, \mathsf{ek}, g^s)\\ \text{witness: } (\{(t_i, s_i)\}_{i\in[0\ldots|s|-1]}, s, \mathsf{dk})\\ (\forall i \in [0\ldots|s|-1] : (A_i = g^{t_i}) \wedge (B_i = \mathsf{ek}^{t_i} \vee B_i = \mathsf{ek}^{t_i} g))\\ \wedge (A = g^{\sum_{i=0}^{|s|-1} t_i 2^i} \wedge B = \mathsf{ek}^{\sum_{i=0}^{|s|-1} t_i 2^i} g^s) \wedge \mathsf{ek} = g^{\mathsf{dk}}\end{gathered}$$

Evaluating an implementation of our proof of concept in Rust [16] yields the results of the benchmarking in Table I. The benchmarking ran the MAPCP 1000 times and computed average times. The experiments were concluded in a Linux machine with a 13th Gen Intel Core i5-1335U (10-core) and 16 GiB of Ram (6400 MHz).

V. Conclusion

We presented MAPCP, a protocol for obtaining digital goods in exchange for a cryptocurrency payment. Our protocol differs from previous works in that it is the first one that simultaneously departs from using zk-SNARKS for the zero-knowledge proofs and HTLC for the atomic exchange of coins and a secret. Given that, MAPCP does not require a trusted setup, sidestepping known limitations of prior work, and provides interoperability and fungibility, essential properties for the privacy of the users. We analyze the security of MAPCP and provide a proof of concept implementation where a customer pays a notary in exchange for the signature on a document. Our evaluation shows that MAPCP is practical even on commodity hardware.

Acknowledgments. This work is supported by the PICOCRYPT project that has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (Grant agreement No. 101001283), by the Excellence Seal María de Maeztu (grant CEX2024-001471-M) by the ESPADA project (grant PID2022-142290OB-I00), MCIN/AEI/10.13039/501100011033/ FEDER, UE, and by the PRODIGY project (grant ED2021-132464B-I00), MCIN/AEI/10.13039/501100011033/ and the European Union NextGenerationEU/ PRTR.

References

[1] J. Gomez-Martinez, D. Vasilopoulos, P. Moreno-Sanchez, and D. Fiore, "Algebraic zero knowledge contingent payment," in *ACNS 2025*.
[2] PriceWaterhouseCoopers, "El salvador's law: a meaningful test for bitcoin," PWC webpage, 2021, https://www.pwc.com/gx/en/financial-services/pdf/el-salvadors-law-a-meaningful-test-for-bitcoin.pdf.
[3] KPMG, "Frontiers in finance: Innovating through platforms and ecosystems," KPMG, 2022, https://assets.kpmg.com/content/dam/kpmg/xx/pdf/2022/05/frontiers-in-finance.pdf.
[4] Shopify, "Shopify help center: cryptocurrencies," Shopify webpage, 2023, https://help.shopify.com/en/manual/payments/additional-payment-methods/cryptocurrency.
[5] AT&T, "At&t now accepts bitpay," AT&T webpage, 2019, https://about.att.com/story/2019/att_bitpay.html.
[6] Bitpay, "Buy from microsoft," Bitpay merchan directory, 2023, https://bitpay.com/directory/microsoft/.
[7] R. Cleve, "Limits on the security of coin flips when half the processors are faulty (extended abstract)," in *18th ACM STOC*, 1986.
[8] G. Maxwell, "Zero knowledge contingent payment," *Bitcoin Wiki*, 2015.
[9] T. B. Wiki, 2019. [Online]. Available: https://en.bitcoin.it/wiki/Hashlock
[10] M. Campanelli, R. Gennaro, S. Goldfeder, and L. Nizzardo, "Zero-knowledge contingent payments revisited: Attacks and payments for services," in *ACM CCS 2017*.
[11] G. Fuchsbauer, "WI is not enough: Zero-knowledge contingent (service) payments revisited," in *ACM CCS 2019*.
[12] L. Aumayr, O. Ersoy, A. Erwig, S. Faust, K. Hostáková, M. Maffei, P. Moreno-Sanchez, and S. Riahi, "Generalized channels from limited blockchain scripts and adaptor signatures," in *ASIACRYPT 2021*.
[13] M. Campanelli, D. Fiore, and A. Querol, "LegoSNARK: Modular design and composition of succinct zero-knowledge proofs," in *ACM CCS 2019*.
[14] S. A. K. Thyagarajan, G. Malavolta, F. Schmidt, and D. Schröder, "PayMo: Payment channels for monero," Cryptology ePrint Archive, Report 2020/1441. [Online]. Available: https://eprint.iacr.org/2020/1441
[15] P. Moreno-Sanchez, A. Blue, D. V. Le, S. Noether, B. Goodell, and A. Kate, "DLSAG: Non-interactive refund transactions for interoperable payment channels in monero," in *FC 2020*.
[16] J. Gomez-Martinez, "Repository for mapcp implementation," https://github.com/jgomezmartinez/mapcp_rust.

Diseño y desarrollo de un sistema predictivo de gestión de amenazas de seguridad utilizando CWE, CVE y CAPEC

Joaquín Sierra-Granados
Grupo de investigación GSyA
UCLM, Ciudad Real, España
joaquin.sierra@uclm.es

José L. Ruiz-Catalán
Grupo de investigación GSyA
UCLM, Ciudad Real, España
joseluis.ruiz@uclm.es

Ferney Martínez Ossa
Grupo de investigación GSyA - PRODIN
UCLM - COTECMAR, España-Colombia
ferney.martinez@alu.uclm.es

Ángel Suárez-Bárcena
Grupo de investigación GSyA
UCLM, Ciudad Real
Angel.SBarcena@uclm.es

Antonio Santos-Olmo
Grupo de investigación GSyA
UCLM, Ciudad Real
antonio.santos@uclm.es

Luis E. Sánchez
Grupo de investigación GSyA
UCLM, Ciudad Real
luise.sanchez@uclm.es

***Resumen*—Las organizaciones dependen cada vez más de plataformas digitales para respaldar sus operaciones, los procesos de toma de decisiones y la prestación de servicios críticos. Esta creciente dependencia ha ampliado su exposición a amenazas cibernéticas que ponen en riesgo la confidencialidad, integridad, disponibilidad y continuidad operativa de los sistemas de información. Este artículo presenta un sistema específicamente diseñado para apoyar la identificación y gestión de riesgos en infraestructuras tecnológicas. La solución propuesta recopila diariamente datos de vulnerabilidades de fuentes oficiales y los correlaciona con los activos de la organización, lo que permite priorizar los riesgos según su nivel de criticidad. Además, el sistema integra un módulo de predicción basado en técnicas de aprendizaje automático, capaz de estimar la evolución agregada del riesgo para el mes siguiente. Esta capacidad predictiva facilita la toma de decisiones preventivas y fortalece las estrategias proactivas de gestión de riesgos de ciberseguridad.**

***Index Terms*—ciberseguridad, CVE, CWE, CAPEC, CVSS, predicción de riesgo, machine learning**

I. Introducción

En los últimos años, la acelerada transformación digital en múltiples sectores ha incrementado significativamente la exposición de las organizaciones a amenazas cibernéticas [20]. Dominios críticos como la salud, la energía, las finanzas, el transporte y los servicios gubernamentales han experimentado un crecimiento sostenido tanto en la frecuencia como en la sofisticación de los ciberataques [15], [12], [13]. Investigaciones recientes destacan un aumento exponencial en el número de vulnerabilidades registradas, acompañado por la proliferación de campañas ofensivas [3], [19], [7]. Estos incidentes no solo comprometen la confidencialidad, integridad y disponibilidad de la información, sino que también interrumpen la continuidad operativa, provocan pérdidas financieras significativas y erosionan la confianza pública.

Se proyecta que el impacto económico del cibercrimen alcance billones de dólares anuales, situándolo entre las mayores economías globales cuando se mide en términos financieros [9], [4]. Al mismo tiempo, la superficie de ataque continúa expandiéndose debido a la adopción masiva de servicios en la nube, dispositivos del Internet de las Cosas (IoT) y sistemas de control industrial interconectados [2].

Esta creciente complejidad plantea serios desafíos a los mecanismos de defensa reactivos tradicionales, subrayando la necesidad urgente de enfoques proactivos y predictivos capaces de mitigar riesgos antes de que se materialicen [16], [5].

En este contexto, la gestión de vulnerabilidades y la predicción de riesgos se han convertido en prioridades estratégicas. Estándares y taxonomías internacionales como Common Vulnerabilities and Exposures (CVE), Common Weakness Enumeration (CWE), Common Attack Pattern Enumeration and Classification (CAPEC) y el Common Vulnerability Scoring System (CVSS) proporcionan marcos estructurados para describir y analizar debilidades y amenazas. Sin embargo, las herramientas actuales se centran principalmente en la detección y la remediación, ofreciendo capacidades limitadas para predecir la evolución del riesgo a lo largo del tiempo [21]. Superar esta limitación requiere la integración de modelos de aprendizaje automático con bases de datos de vulnerabilidades, lo que permite generar inteligencia accionable, anticipar amenazas potenciales, optimizar la asignación de recursos y fortalecer la ciberresiliencia organizacional [6].

Para abordar estas limitaciones, este trabajo propone el desarrollo de un sistema de gestión de vulnerabilidades y activos denominado VulnQ, implementado como una aplicación web diseñada para facilitar la evaluación y el monitoreo continuo de los riesgos de ciberseguridad.

Este artículo se organiza de la siguiente manera: la Sección 2 revisa trabajos relacionados y herramientas existentes en el campo de la gestión de vulnerabilidades y riesgos de ciberseguridad. La Sección 3 detalla el desarrollo del sistema propuesto, incluyendo su arquitectura, base de datos, módulos de priorización y predicción, así como los procesos de integración y visualización. La Sección 4 describe un caso de estudio práctico llevado a cabo en un entorno de Hogar Inteligente, demostrando la aplicabilidad y eficacia del sistema VulnQ en la detección, análisis y predicción de vulnerabilidades. Finalmente, la Sección 5 presenta las principales conclusiones y líneas de trabajo futuro.

II. Trabajo relacionado

La gestión del riesgo de ciberseguridad se ha convertido en un proceso continuo y complejo debido a la expansión de la superficie de ataque y al rápido incremento en la publicación de vulnerabilidades. Aunque la aparición de nuevos estándares y marcos regulatorios como ISO/IEC 27001 [11], NIST [18], CSF [17], NIS2 [8] y GDPR [24] ha proporcionado a las organizaciones respuestas estructuradas frente al creciente panorama de amenazas, la mayoría de estos estándares actuales siguen centrados en la gestión reactiva de incidentes, mientras que los enfoques predictivos aún se encuentran en una fase emergente [22], [10], [25].

La comunidad científica y técnica se apoya en taxonomías estandarizadas —CPE, CVE, CWE, CAPEC y CVSS— que permiten la interoperabilidad de los datos de seguridad y la vinculación de vulnerabilidades con inventarios de activos [27], [1]. Sin embargo, el uso de sistemas de puntuación estáticos como CVSS limita la consideración de factores dinámicos y contextuales. En respuesta, investigaciones recientes han propuesto métodos alternativos para mitigar estas limitaciones [14].

II-A. Herramientas existentes

Existen soluciones consolidadas que abordan parcialmente este problema. Wazuh [26], una plataforma de código abierto, ofrece capacidades destacadas en gestión de información y eventos de seguridad (SIEM), gestión de registros y cumplimiento normativo. También incluye detección de vulnerabilidades basada en el inventario de software; sin embargo, presenta limitaciones debido a la ausencia de escaneo activo y de mecanismos avanzados de priorización predictiva.

Por el contrario, Nessus [23], en su versión comercial, sobresale en el escaneo activo de sistemas y en la priorización del riesgo mediante métricas como CVSS, VPR (Vulnerability Priority Rating) y EPSS (Exploit Prediction Scoring System); no obstante, carece de una gestión integrada de activos y de modelado predictivo contextualizado.

La Tabla I resume esta comparación; a diferencia de Wazuh y Nessus, VulnQ unifica la agregación de riesgos basada en activos con capacidades predictivas orientadas al futuro dentro de un mismo flujo operativo. Cabe destacar que esta comparación es de carácter cualitativo y se basa en funcionalidades documentadas públicamente.

III. Desarrollo

La siguiente sección presenta el proceso de diseño e implementación del sistema propuesto. Este desarrollo abarca la definición de su arquitectura modular, la integración de bases de datos, los algoritmos de priorización y los modelos de predicción basados en aprendizaje automático. Además, detalla los principales componentes funcionales de la plataforma, así como las tecnologías utilizadas para su construcción y despliegue.

III-A. Base del sistema

El sistema propuesto se estructura como una arquitectura modular compuesta por tres capas principales: el servidor de aplicación, la interfaz de usuario y el motor de predicción.

La Fig. 1 presenta la arquitectura de la solución propuesta, desarrollada bajo el modelo Software como Servicio (SaaS).

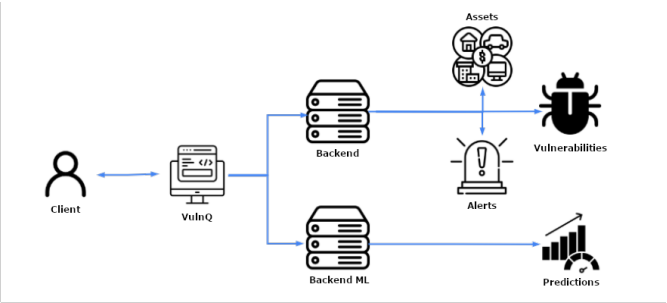


Figura 1. Arquitectura de la solución propuesta

El diseño está inicialmente orientado al sector Business-to-Business (B2B), dado que la naturaleza técnica de los datos requiere usuarios con conocimientos especializados en ciberseguridad. No obstante, se prevé una evolución progresiva hacia perfiles de usuario no especializados mediante mejoras en la accesibilidad y la simplificación de la interfaz.

En primer lugar, el servidor de aplicación actúa como intermediario entre el cliente y la base de datos, gestionando la comunicación, las peticiones y las respuestas (Fig. 1, componente Backend). Su función principal es coordinar el flujo de información, garantizar la integridad de los datos y aplicar las reglas de negocio definidas para la gestión de activos y vulnerabilidades.

En segundo lugar, la interfaz de usuario está implementada como un cliente web (Fig. 1, componente VulnQ) que permite la interacción directa con el sistema. Esta capa está diseñada para proporcionar una experiencia intuitiva y accesible mediante componentes dinámicos que facilitan consultas de vulnerabilidades, auditorías de activos y visualización de métricas de riesgo.

Finalmente, el motor de predicción corresponde al módulo encargado de procesar tanto datos históricos como actuales para estimar la evolución futura del riesgo (Fig. 1, componente BackendML). Este componente se basa en técnicas de aprendizaje automático, permitiéndole analizar patrones de vulnerabilidades y generar predicciones sobre los niveles de exposición de los activos en periodos de tiempo definidos. El motor expone servicios específicos que pueden ser consumidos por el servidor de aplicación, integrando así sus resultados en el flujo operativo general del sistema.

Esta arquitectura por capas garantiza una mayor escalabilidad, mantenibilidad y flexibilidad, permitiendo que cada componente evolucione de forma independiente y se adapte a nuevos requisitos funcionales o tecnológicos.

III-B. Base de datos y motor de búsqueda

La arquitectura de datos del sistema se basa en un repositorio relacional que integra múltiples catálogos de vulnerabilidades estandarizados. En concreto, se han definido esquemas dedicados para los diccionarios CVE, CWE, CAPEC y CPE. Estos esquemas permiten estructurar la información de forma consistente, garantizando la trazabilidad y la correcta vinculación entre vulnerabilidades, debilidades, patrones de ataque y las plataformas tecnológicas afectadas.

Para poblar la base de datos, se ha desarrollado un proceso de Extracción, Transformación y Carga (ETL) destinado a automatizar la recopilación de información desde las fuentes

Tabla I
COMPARATIVA ENTRE WAZUH, NESSUS Y VULNQ

Característica	Wazuh	Nessus	VulnQ
Enfoque y funcionalidad	SIEM, detección de intrusiones, gestión de vulnerabilidades	Escáner de vulnerabilidades	Gestión unificada de activos, vulnerabilidades y riesgos
Técnica de detección	Inventario de software y correlación con CVE	Escaneo activo de sistemas y servicios	Inventario de software y correlación con CVE
Clasificación de vulnerabilidades	Reglas e integración de eventos	CVSS, VPR y recomendaciones automáticas	Puntuación de riesgo personalizada basada en CVSS y KRI calculado
Modelo de distribución	Código abierto	Comercial, versión gratuita limitada	Comercial, versión gratuita limitada
Integraciones	SIEM y fuentes externas de seguridad	Soluciones de Tenable y otras plataformas	API REST

oficiales. Este proceso descarga los archivos publicados en formatos estructurados, los transforma según el modelo definido y los importa periódicamente al sistema. De este modo, el repositorio se mantiene actualizado y refleja fielmente el estado más reciente del ecosistema de amenazas.

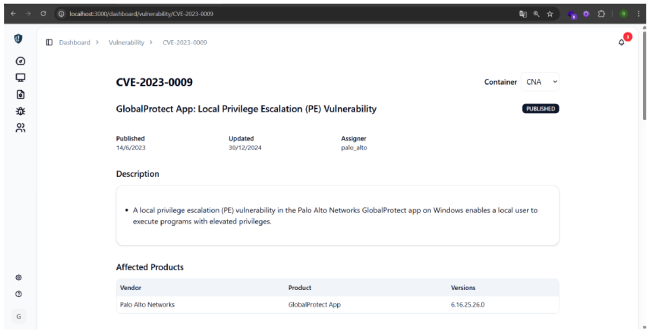


Figura 2. Visualización de vulnerabilidades en VulnQ

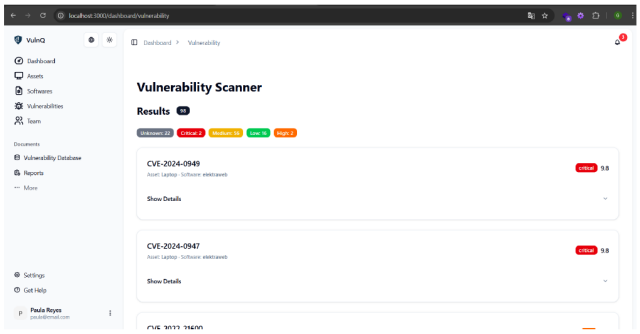


Figura 3. Escáner de vulnerabilidades de VulnQ

El módulo de consulta de vulnerabilidades (Fig. 2) permite realizar búsquedas basadas en identificadores estandarizados (p. ej., CVE). Este módulo facilita la interacción directa del usuario con la base de datos y constituye el núcleo del sistema de gestión, ya que proporciona un acceso rápido y estructurado a información crítica de seguridad.

Además, el sistema integra una interfaz de auditoría que muestra los resultados de los análisis realizados sobre los activos registrados (Fig. 3). Esta interfaz correlaciona las vulnerabilidades detectadas con los dispositivos y aplicaciones inventariados, ofreciendo una vista consolidada de los niveles de exposición y apoyando la priorización de medidas de mitigación.

III-C. Priorización y alertas

El sistema incorpora un mecanismo de priorización de vulnerabilidades basado en métricas estandarizadas de severidad. En particular, adopta como indicador principal la valoración de referencia de los sistemas de evaluación de vulnerabilidades (p. ej., la puntuación base de CVSS), debido a su amplio reconocimiento y capacidad para unificar criterios de evaluación.

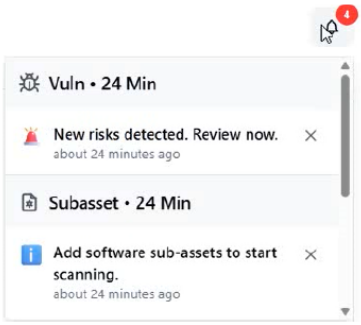


Figura 4. Alertas del sistema VulnQ

A partir de esta métrica, se implementa un algoritmo de priorización encargado de organizar y filtrar los resultados obtenidos en los procesos de análisis y auditoría. Este algoritmo aplica funciones auxiliares de ordenación y clasificación para estructurar la información según el nivel de riesgo, facilitando así la identificación de vulnerabilidades críticas que requieren atención inmediata.

En paralelo, se ha definido un modelo de alertas con el propósito de notificar automáticamente la ocurrencia de eventos relevantes del sistema. Estos eventos incluyen tanto la detección de vulnerabilidades asociadas a activos registrados como incidentes que puedan comprometer el correcto funcionamiento de la plataforma.

El mecanismo de alertas basado en eventos permite la notificación en tiempo real, asegurando que los usuarios reciban información oportuna sobre situaciones críticas (Fig. 4). Para su presentación, se ha desarrollado una interfaz de visualización de alertas diseñada conforme a patrones de arquitectura que separan la lógica de negocio de la representación gráfica. Este componente facilita la consulta y el seguimiento de las notificaciones, proporcionando una experiencia clara y estructurada.

III-D. Estimación de riesgo (ML)

El sistema incorpora un módulo de estimación predictiva del riesgo basado en Key Risk Indicators (KRI), diseñado

para anticipar el nivel de exposición de los activos tecnológicos. Este proceso se basa en atributos estandarizados de los catálogos CWE, CAPEC y CVE para calcular tres KRI diferenciados.

Para CWE, el valor KRI se asigna empleando una escala heurística determinada por la probabilidad de explotación: probabilidades altas reciben la puntuación máxima de 10, medias una de 6, bajas una de 3 y un valor por defecto de 5 en cualquier otro caso. Para CAPEC, el KRI se calcula multiplicando dos factores —probabilidad de ataque y severidad típica— ambos mapeados a valores normalizados (p. ej., 1.0 para "Alta" o "Muy Alta", 0.6 para "Media", 0.3 para "Baja", con valores por defecto en caso necesario). El resultado se escala posteriormente por un factor de 10 para obtener la puntuación final KRICAPEC.

Finalmente, la entrada CVE se calcula como una suma ponderada de cuatro variables: la puntuación base máxima de CVSS (60 %), el número de exploits (10 %), el KRICWE previamente calculado (10 %) y KRICAPEC (20 %). Esta métrica compuesta permite una estimación del riesgo dinámica y contextualizada que favorece una priorización más informada y estrategias proactivas de mitigación.

Estas métricas se almacenan en un conjunto de datos histórico utilizado para entrenar un modelo predictivo basado en regresión que estima la evolución mensual del riesgo de los activos. Este modelo mejora la gestión tradicional de vulnerabilidades al permitir estrategias proactivas y conscientes del contexto. La formulación actual de los KRI se basa en escalas heurísticas derivadas de conocimiento experto, lo que aporta interpretabilidad pero introduce cierto grado de subjetividad.

III-E. Entrenamiento y predicción

El módulo predictivo del sistema que está en sus primeras estapas se centra en construir un modelo capaz de anticipar la evolución del riesgo de los activos tecnológicos a partir de datos históricos (Fig. 1, componente BackendML).

En la fase de entrenamiento, se aplica un enfoque basado en regresión sobre el conjunto histórico de métricas KRI donde este se dividió en un 70 % para entrenamiento y un 30 % para validación. Las variables de entrada corresponden a los valores históricos mensuales de KRI_CVE, KRI_CWE y KRI_CAPEC, mientras que la variable objetivo es el nivel agregado de riesgo del activo para el mes siguiente. El uso de métricas históricas como entrada garantiza que el modelo incorpore tanto tendencias pasadas como la dinámica temporal de la exposición a amenazas.

Posteriormente, en la fase de predicción, el modelo genera estimaciones mensuales de riesgo por activo. Estas proyecciones se derivan de los resultados de las auditorías realizadas y de su correlación con los valores KRI calculados, permitiendo al sistema anticipar cambios a corto plazo en los niveles de exposición. Las predicciones se integran en el panel de control del sistema, proporcionando a los responsables de seguridad una herramienta de apoyo a la decisión para la acción preventiva y la planificación estratégica.

III-F. Integración y pruebas del modelo

El modelo de aprendizaje automático se integró en la arquitectura general del sistema a través de la capa de servicios, permitiendo la comunicación directa con los demás componentes funcionales. Esta integración siguió un enfoque estructurado que garantiza una clara separación entre la lógica de negocio y la lógica predictiva, manteniendo la coherencia con los principios de diseño modular.

Como parte de esta integración, se implementaron operaciones de gestión para permitir consultas, almacenamiento de resultados y actualización dinámica de predicciones. Estas operaciones proporcionan un marco de interacción estandarizado con el modelo, facilitando su incorporación en los flujos de trabajo del sistema y su reutilización en futuros escenarios analíticos. El modelo se reentrena periódicamente utilizando nuevos datos de KRI, permitiendo adaptarse a cambios en los patrones de amenazas y mitigar efectos de data drift.

Finalmente, se llevaron a cabo pruebas de validación extremo a extremo para verificar tanto la precisión de las predicciones como la correcta comunicación entre los distintos módulos. Estas pruebas confirmaron la capacidad del sistema para procesar datos históricos, generar estimaciones de riesgo y presentar los resultados de forma integrada dentro del panel de ciberseguridad.

III-G. Visualización

El sistema incorpora un módulo de visualización de datos diseñado para ofrecer una vista consolidada del estado de seguridad de los activos tecnológicos. Para ello, se han desarrollado gráficos de tipo resumen que representan de manera clara y concisa información crítica relacionada con vulnerabilidades, debilidades y niveles de riesgo.

La generación de estas vistas se basa en un proceso ETL que consolida datos provenientes de múltiples catálogos y resultados de auditorías. Este proceso garantiza la consistencia y coherencia de la información mostrada en la interfaz, permitiendo a los usuarios acceder a representaciones visuales actualizadas y confiables en todo momento.

De esta forma, en la Fig. 5, el sistema ofrece una herramienta de analítica visual que mejora la interpretación de indicadores clave y potencia la capacidad analítica al permitir la identificación rápida de tendencias y la priorización de acciones de ciberseguridad.



Figura 5. Panel de control del sistema VulnQ

IV. Caso de estudio

El caso de estudio se desarrolló en el laboratorio GSyA utilizando un entorno de pruebas SmartHome diseñado para simular un ecosistema digital heterogéneo e interconectado. Este entorno incluía dispositivos IoT y TI como cámaras IP,

Google Home, sensores Zigbee, enchufes inteligentes y un hub basado en Raspberry Pi con Home Assistant y un broker MQTT. Además, se integró un sitio web en WordPress alojado en un ordenador portátil dentro de una red segmentada y protegida mediante firewall.

El objetivo principal fue validar la funcionalidad, integración y capacidad predictiva de la plataforma de Gestión de Activos y Vulnerabilidades *VulnQ* en un entorno real y multidispositivo. Se evaluó su capacidad para identificar vulnerabilidades, priorizar acciones de mitigación y estimar la evolución del riesgo siguiendo los principios de confidencialidad, integridad y disponibilidad (CIA).

El proceso comenzó con el registro y categorización de todos los activos a través de la interfaz web de VulnQ, integrándose en su base de datos relacional con taxonomías estándar como CVE, CWE, CAPEC y CPE. Posteriormente, el proceso ETL sincronizó automáticamente la plataforma con los repositorios oficiales, garantizando información de vulnerabilidades actualizada.

Finalmente, el módulo de consulta permitió recuperar las vulnerabilidades asociadas a cada activo mediante identificadores como CVE, facilitando el mapeo de debilidades conocidas con los dispositivos del entorno SmartHome. Como ejemplo, se analizaron métricas detalladas de un activo representativo (portátil/WordPress).

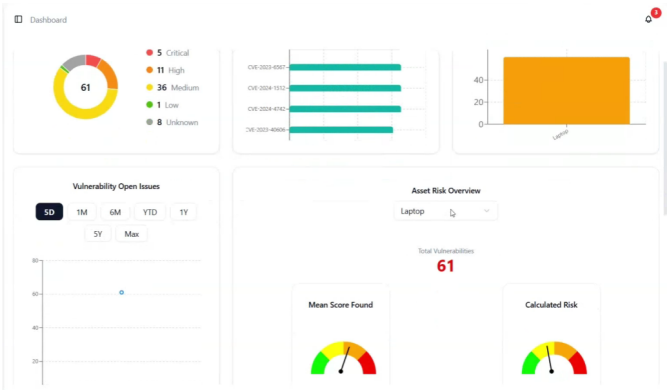


Figura 6. Muestra de vulnerabilidades



Figura 7. Métricas totales de vulnerabilidades

Tras el registro de los activos, la interfaz de auditoría permitió realizar evaluaciones automáticas del entorno, correlando las vulnerabilidades detectadas mediante CVE y CPE con el hardware y software registrados. El módulo de priorización clasificó estas vulnerabilidades utilizando la puntuación base de CVSS como referencia principal.

Los resultados se visualizaron en el panel de control, que consolida indicadores como severidad (CVSS), número de vulnerabilidades y nivel de riesgo. En este caso, se identificaron vulnerabilidades de alta severidad tanto en la plataforma WordPress como en su sistema operativo, lo que evidenció la necesidad de aplicar medidas de mitigación inmediatas. Las vulnerabilidades marcadas como *Unknown* fueron señaladas para verificación manual con el fin de evitar falsos positivos. Finalmente, las métricas globales del activo se resumen (Fig. 7).

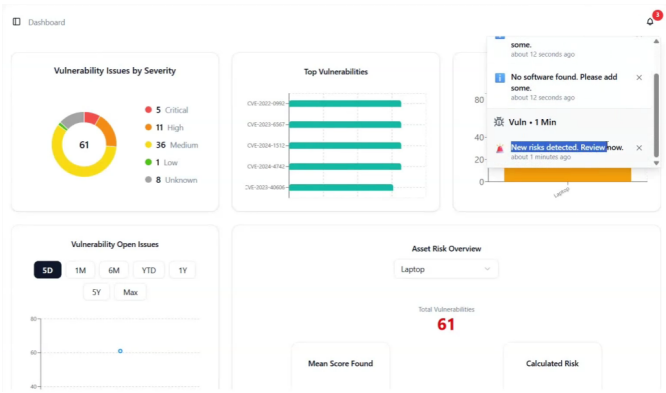


Figura 8. Notificación de nuevas vulnerabilidades detectadas

Se activó el subsistema de alertas para generar notificaciones en tiempo real sobre vulnerabilidades y anomalías críticas (Fig. 8), integradas en el panel web para soportar la monitorización continua.

Asimismo, se evaluó el módulo de estimación predictiva del riesgo, basado en un motor de aprendizaje automático que utiliza KRIs derivados de CVE, CWE y CAPEC. A partir de valores históricos y atributos contextuales, el sistema generó predicciones cuantitativas del riesgo mensual por activo, permitiendo anticipar incrementos de riesgo y priorizar acciones preventivas.

Una vez actualizado el panel, tal como muestra la Fig. 9, el número de vulnerabilidades identificadas aumentó.



Figura 9. Métricas totales tras la actualización de vulnerabilidades

El caso de estudio mostró que la herramienta *VulnQ* mejora notablemente la detección y gestión de vulnerabilidades en el entorno SmartHome del laboratorio GSyA, al combinar auditoría tradicional con análisis predictivo mediante aprendizaje automático. Esta aproximación permite una evaluación continua del riesgo, prioriza la mitigación de vulnerabilidades críticas y reduce los tiempos de respuesta. Asimismo, la actualización constante de su base de datos facilita una protección proactiva ante amenazas emergentes, fortaleciendo la seguridad del entorno inteligente.

V. Conclusiones

Se ha desarrollado un sistema de información para apoyar las necesidades de ciberseguridad y gestión de activos de las organizaciones. Mantiene un inventario jerárquico de activos con gestión completa CRUD e incorpora la recuperación automatizada de datos de vulnerabilidades oficiales. Un módulo de auditoría vincula los activos con sus vulnerabilidades asociadas, generando informes y alertas categorizadas. Además, el sistema proporciona un panel de control con métricas clave,

incluyendo tendencias de severidad de amenazas y pronósticos mensuales de riesgo derivados de un modelo de aprendizaje automático. Todas las funcionalidades se entregan a través de una interfaz web unificada, desplegada con procesos automatizados que equilibran eficiencia operativa y calidad de la información. *VulnQ* cumple con los objetivos propuestos; sin embargo, para competir con herramientas consolidadas del mercado, se han identificado varias líneas de desarrollo para mejorar su precisión, escalabilidad y aplicabilidad en diversos entornos.

Las principales limitaciones de este trabajo incluyen el carácter heurístico de los KRI, la escala reducida del entorno experimental, y el uso de un modelo de regresión base sin comparación empírica con modelos más complejos.

El trabajo futuro incluirá bucles de retroalimentación para reentrenamiento continuo, expansión de las funcionalidades de riesgo y la migración de la plataforma a una arquitectura de microservicios escalable con mensajería basada en eventos y mayor seguridad (p. ej., MFA). VulnQ se generalizará más allá de los Smart Homes hacia dominios de ciudades inteligentes, ICS y IoT en salud mediante una capa federada que permita inferencia en el borde, agregación segura de modelos y computación preservando la privacidad bajo conectividad intermitente. La validación cruzada en entornos multi-activo reportará métricas de escalabilidad, generalización y privacidad/seguridad. El modelo de regresión actual servirá como línea base interpretable para la comparación con modelos Random Forest, XGBoost y LSTM, bajo restricciones de despliegue en entornos *edge* y federados.

Agradecimientos

Esta publicación no habría sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MICIU/AEI/10.13039/501100011033 y por la Unión Europea (Programa Chist-Era); KOSMOS-UCLM (PID2024-155363OB-C44), financiado por MICIU/AEI/10.13039/501100011033 /FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la JCCM y FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447), financiado por FEDER; y la beca RED2024-154240-T, financiada por MICIU/AEI/10.13039/501100011033.

Referencias

[1] Ehsan Aghaei, Ehab Al-Shaer, Waseem Shadid, and Xi Niu. 2023. Automated CVE Analysis for Threat Prioritization and Impact Prediction. doi:10.48550/arXiv.2309.03040

[2] D. G. Rosado, A. Santos-Olmo, L. E. Sánchez, M. A. Serrano, C. Blanco, H. Mouratidis, and E. Fernández-Medina, "Managing cybersecurity risks of cyber-physical systems: The marisma-cps pattern," Computers in Industry, vol. 142, p. 103715, 2022. https://doi.org/10.1016/j.compind.2022.103715

[3] Seyed Ali Akhavani, Behzad Ousat, and Amin Kharraz. 2025. Open Source, Open Threats? Investigating Security Challenges in Open-Source Software. https://arxiv.org/abs/2506.12995

[4] Naeem Allahrakha. 2024. Impacts of Cybercrimes on the Digital Economy. Uzbek Journal of Law and Digital Policy 2 (08 2024), 29–36. doi:10.59022/ujldp.20

[5] Benjamin Ampel, Sagar Samtani, Hongyi Zhu, and Jay Nunamaker. 2024. Improving Threat Mitigation Through a Cybersecurity Risk Management Framework: A Computational Design Science Approach. Journal of Management Information Systems 41 (03 2024), 236–265. doi:10.1080/07421222.2023.2301178

[6] Khalid Bennouk, Dorra Mahouachi, Nawal Ait Aali, Younès El Bouzakri El Idrissi, Bechir Sebai, and Abou Zakaria Faroukhi. 2025. Dynamic Data Updates and Weight Optimization for Predicting Vulnerability Exploitability. IEEE Access PP(01 2025), 1–1. doi:10.1109/ACCESS.2025.3558990

[7] CyberPress. 2025. Over 40,000 CVEs published in 2024. Retrieved June 30, 2025 from https://cyberpress.org/over-40000-cves-published-in-2024/

[8] NIS2 Directive. 2025. NIS2 Directive - Official Information Portal. Retrieved June 23, 2025 from https://www.nis-2-directive.com/

[9] Martin Eling, Mauro Elvedi, and Greg Falco. 2022. The Economic Impact of Extreme Cyber Risk Scenarios. North American Actuarial Journal 27 (03 2022), 1–15. doi:10.1080/10920277.2022.2034507

[10] Adebola Folorunso, Viqaruddin Mohammed, Ifeoluwa Wada, and Bunmi Samuel. 2024. The impact of ISO security standards on enhancing cybersecurity posture in organizations. World Journal of Advanced Research and Reviews 24 (10 2024), 2582–2595. doi:10.30574/wjarr.2024.24.1.3169

[11] International Organization for Standardization (ISO). 2022. ISO/IEC 27001:2022 - Information Security, Cybersecurity and Privacy Protection. Retrieved June 23, 2025 from https://www.iso.org/obp/ui/en/#iso:std:iso-iec:27001:ed-3:v1:en

[12] Le Ha. 2022. Are digital business and digital public services a driver for better energy security? Evidence from a European sample. Environmental Science and Pollution Research 29 (01 2022), 27232–27256. doi:10.1007/s11356-021-17843-2

[13] Aleksy Kwilinski, Lyulyov Oleksii, and Tetyana Pimonenko. 2023. Environmental Sustainability within Attaining Sustainable Development Goals: The Role of Digitalization and the Transport Sector. Sustainability 15 (07 2023), 11282. doi:10.3390/su151411282

[14] Takashi Minohara, Masaya Shimakawa, and Sora Okada. 2025. Security Vulnerability Risk Growth Model based on CVSS 4.0. doi:10.1109/DSN-S65789.2025.00074

[15] Marc Mitchell and Lena Kan. 2019. Digital Technology and The Future of Health Systems. Health Systems & Reform 5 (02 2019). doi:10.1080/23288604.2019.1583040

[16] Henock Mulugeta. 2023. Context-Based and Adaptive Cybersecurity Risk Management Framework. Risks 11 (05 2023). doi:10.3390/risks11060101

[17] National Institute of Standards and Technology (NIST). 2018. Framework for Improving Critical Infrastructure Cybersecurity, Version 1.1. Retrieved September 26, 2025 from https://www.nist.gov/cyberframework

[18] National Institute of Standards and Technology (NIST). 2020. Zero Trust Architecture (NIST SP 800-207). Retrieved June 23, 2025 from https://nvlpubs.nist.gov/nistpubs/specialpublications/NIST.SP.800-207.pdf

[19] David Pecl, Yehor Safonov, Zdenek Martinasek, Matej Kacic, Lubomir Almer, and Lukas Malina. 2021. Manager Asks: Which Vulnerability Must be Eliminated First?. In Innovative Security Solutions for Information Technology and Communications. Springer International Publishing, Cham, 146–164.

[20] Saqib Saeed, Salha Altamimi, Norah Alkayyal, Ebtisam Alshehri, and Dina Alabbad. 2023. Digital Transformation and Cybersecurity Challenges for Businesses Resilience: Issues and Recommendations. Sensors 23 (07 2023). doi:10.3390/s23156666

[21] Kamran Shaukat, Suhuai Luo, Vijay Varadharajan, Ibrahim Hameed, Shan Chen, Dongxi Liu, and Jiaming Li. 2020. Performance Comparison and Current Challenges of Using Machine Learning Techniques in Cybersecurity. Energies 13 (052020), 2509. doi:10.3390/en13102509

[22] Elham Tabassi. 2023. Artificial Intelligence Risk Management Framework (AI RMF 1.0). doi:10.6028/NIST.AI.100-1

[23] Tenable. 2025. Nessus Vulnerability Scanner. Retrieved June 1, 2025 from https://www.tenable.com/products/nessus

[24] European Union. 2016. General Data Protection Regulation. Retrieved September 26, 2025 from https://eur-lex.europa.eu/eli/reg/2016/679/oj

[25] Apostol Vassilev, Alina Oprea, Alie Fordyce, and Hyrum Anderson. 2024. Adver-sarial Machine Learning: A Taxonomy and Terminology of Attacks and Mitigations. doi:10.6028/NIST.AI.100-2e2023

[26] Wazuh. 2025. Wazuh - Open Source Security Platform. Retrieved June 4, 2025 from https://documentation.wazuh.com/current/user-manual/capabilities/vulnerability-detection/index.html

[27] Ferda Özdemir Sönmez, Chris Hankin, and Pasquale Malacaria. 2022. Attack Dynamics: An Automatic Attack Graph Generation Framework Based on System Topology, CAPEC, CWE, and CVE Databases. Computers & Security 123 (10 2022). doi:10.1016/j.cose.2022.102938

Towards a European Data and AI Ecosystem: Convergences between the DGA, DA, AIA, and GDPR

Ignacio Miguel Rodríguez*, Jordi Castellà Roca*, Alexandre Viejo* M. Magdalena Payeras Capellà†
*Universitat Rovira i Virgili, Departament d'Enginyeria Informàtica i Matemàtiques, Tarragona, Spain
{ignacio.miguel, jordi.castella, alexandre.viejo}@urv.cat
†Universitat de les Illes Balears, Departament de Ciències Matemàtiques i Informàtica, Palma, Spain
mpayeras@uib.es

***Abstract*—The European Union has established a comprehensive and pioneering regulatory landscape to govern data protection, data sharing, and the deployment of artificial intelligence. Regulations such as the GDPR, DA, DGA, and AIA provide frameworks to safeguard individual rights, ensure data security and integrity, and promote trustworthy innovation. This paper presents a comparative overview of these regulatory instruments, highlighting their main objectives, actors, and responsibilities. Additionally, we analyze the opportunities they create for data-driven innovation and AI deployment, as well as the challenges associated with compliance, interoperability, transparency, and human oversight. The discussion emphasizes the EU's leading role in shaping global standards for ethical and responsible data and AI.**

***Index Terms*—GDPR, DGA, DA, AIA, Europe, Data, AI**

I. Introduction

In today's digital era, data has become the driving force behind most services and innovations. To ensure the security and privacy of individuals whose data is being collected and processed, it is crucial to establish clear rules and regulatory frameworks. In this regard, the European Union (EU) has taken a leading role in shaping global standards for data governance—starting with the General Data Protection Regulation (GDPR) in 2018 and continuing with more recent initiatives such as the Data Governance Act (DGA), the Data Act (DA), and most notably, the Artificial Intelligence Act (AIA), reflecting the rapid progress of AI technologies in recent years.

Together, these regulations seek to prevent misuse of data, mitigate biases and unethical practices in AI systems, and safeguard individual rights by ensuring transparency, fairness, and user control. In particular, where processing is based on consent, the GDPR guarantees that individuals may freely give or withdraw it at any time, and the other instruments must be applied without prejudice to this right.

In this regulatory context, it is precise to define what is considered as personal and non-personal data, as they are treated differently. According to [1], *personal data* refers to any information relating to an identified or identifiable natural person, meaning one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, or an online identifier. Conversely, *non-personal data* refers to any information that does not relate to an identified or identifiable person, or that has been anonymized in such a way that individuals are no longer identifiable. The distinction is fundamental, as the GDPR applies specifically to personal data, whereas non-personal data are primarily governed by other regulatory instruments such as the DGA and the DA, which establish frameworks for data sharing and reuse across sectors within the EU. However, both the DGA and the DA may also encompass personal data under certain conditions, insofar as their provisions are fully consistent with the GDPR. In practice, this means that the GDPR remains the primary and prevailing legal basis for the protection and lawful processing of personal data. The DGA, the DA and the AI Act complement it by defining mechanisms for secure access, sharing, and reuse of data, but must be applied without prejudice to the GDPR. Consequently, data-subject rights, such as the right to withdraw consent at any time (Art. 7(3)), cannot be restricted by these instruments.

This paper provides an overview of the EU's current data-related regulatory landscape, identifying its main actors, their duties, and responsibilities. Furthermore, it explores the interconnections among these frameworks and offers a comparative analysis of their objectives, scope, and implications.

II. Common actors

This section summarizes the common actors across all regulations, specifying their main obligations and responsibilities.

1) **Data Subjects**: Individuals whose personal data is processed and can be identified by any identifier directly or indirectly. This is a central concept in GDPR and relevant in other acts where personal data is involved.
2) **Data Holder**: An individual or entity that, while not being the data subject, possesses the authority to grant access to or share certain personal or non-personal data.
3) **Data Controllers**: Entities that determine the purposes and means of processing personal data. It is crucial under GDPR and may be relevant in other acts when data handling is involved.
4) **Data Processors**: Entities that process data on behalf of the data controller. Again, defined in GDPR but also relevant in the context of data processing in other acts.
5) **Users**: Individuals or entities that use digital services or products. This actor is relevant across all the mentioned legislations.
6) **Developers and Manufacturers (Providers)**: Companies involved in creating AI systems or digital products,

particularly relevant in the AIA and DA.

7) **Regulatory / Supervisory Authorities**: Bodies responsible for overseeing compliance with the legislation.

After conducting a comparative analysis of the four main European regulations (GDPR, AIA, DGA, and DA), we identified the key actors that appear consistently across these frameworks. Table I consolidates these common actors and summarizes their primary responsibilities and rights as defined by the respective regulations. The process involved mapping the provisions of each regulation to identify overlapping roles and extracting the most relevant obligations and entitlements for each actor. While this synthesis provides a unified perspective, it does not include some regulation-specific nuances or additional actors unique to individual frameworks. Note that throughout this document, references to Articles and Chapters are made directly, without explicitly restating the corresponding regulatory instrument.

III. Regulations

Depending on the specific use case of an application, one or more regulations may apply. Figure 1 illustrates a decision workflow to determine the applicability of key European regulatory frameworks through a series of guiding questions. These regulations are cumulative, meaning that multiple frameworks can apply simultaneously. For instance, AI-driven apps that analyze biometric data from wearables to provide personalized health recommendations must comply with GDPR for sensitive data, AIA, as it is classified as high-risk, and DA, as it is a connected product that generates data. To verify whether an AI system qualifies as low-risk under the EU AI Act, organizations can consult the official questionnaire[1] provided by the European Union. Next, we provide a summary of the key aspects of each regulation.

A. General Data Protection Regulation (GDPR)

The GDPR establishes a comprehensive regulatory framework, showcasing the European Union's pioneering role in setting global standards for personal data protection. According to [1], personal data refers to any information relating to an identifiable individual (data subject), including identification numbers, location data, genetic information, and similar attributes. The regulation empowers data subjects with control over how their data are collected, processed, and managed by service providers. Its main pillars include:

- **Right to access, rectification, and erasure (Right to be forgotten)** — Articles 15, 16, and 17.
- **Right to restriction of processing** — Article 18.
- **Right to data portability** — Article 20.
- **Right to object to processing** — Article 21.

These rights ensure that individuals maintain significant authority over their personal information, reinforcing transparency, accountability, and fairness in data processing activities.

B. Data Act (DA)

The DA establishes a harmonized framework to ensure fair access to and use of data across the European Union, fostering a competitive and innovative data economy. It complements the Data Governance Act by focusing on the practical mechanisms of data access and sharing across sectors. According to [2], the regulation seeks to ensure that users, businesses, and public sector bodies can access and utilize data generated by connected devices (IoT) and related services under fair, reasonable, and non-discriminatory conditions.

Its main pillars include:

- **Right of access to data generated by IoT devices** (Chapter II).
- **Fairness in business-to-business (B2B) data sharing**: It establishes obligations to prevent unfair contractual terms and promote equitable data access among enterprises (Chapter IV).
- **Obligation of business-to-government (B2G) data sharing for the public interest**: The regulation enables public sector bodies to request access to privately held data in cases of exceptional need, such as public emergencies (Chapter V).
- **Right of change between data processing services**: The DA promotes cloud switching and interoperability between data processing services, aiming to reduce vendor lock-in and foster competition (Chapter VI). This objective aligns with the GDPR's Article 20 on data portability.
- **International data transfers and safeguards**: It ensures that non-personal data leaving the EU remain protected under conditions equivalent to EU standards, thereby safeguarding data sovereignty (Chapter VII).
- **Data interoperability**: Establishes requirements for data interoperability, including technical standards and mechanisms for secure data sharing, as well as the use of smart contracts to facilitate automated and transparent exchanges (Chapter VIII).

These provisions aim to create a fair and trustworthy data economy in which users retain control over the data they generate, businesses can innovate responsibly, and public authorities can leverage data for societal benefit.

C. Data Governance Act (DGA)

The DGA focuses on promoting the reuse of data to enable data-driven solutions for both public and private entities. It also establishes the rules for data sharing and data altruism, introducing new certification schemes for enterprises that offer data intermediation services or engage in data altruism.

Data intermediation services (Chapter III) facilitate data sharing between multiple data subjects and data holders on one side and data users on the other side. These services build commercial relationships to share personal or non-personal data, while ensuring that data subjects can exercise their rights with respect to their personal data. These neutral figures are intended to foster trust in data sharing between individuals and enterprises. Currently, only a few companies are registered in the EU register of data intermediation services[2], with

[1]EU AI Risk Classifier and Compliance Checker: https://www.trail-ml.com/eu-ai-act-compliance-checker

[2]https://digital-strategy.ec.europa.eu/en/policies/data-intermediary-services Accessed in 2026

Table I
COMMON ACTORS ACROSS THE GDPR, AIA, DGA, AND DA: MAIN RESPONSIBILITIES AND RIGHTS

Actor	Responsibilities / Obligations	Rights
Data Subjects	Provide accurate personal data; exercise their data protection rights; understand how their data are processed and used; cooperate with data controllers when required.	Right of access, rectification, erasure (right to be forgotten), restriction of processing, portability, and objection; right to transparent and lawful processing of their data.
Data Holders	Maintain and manage data securely; ensure that sharing or granting access complies with applicable laws and agreements; preserve data quality, confidentiality, and integrity.	Authorize access to or sharing of personal and non-personal data under defined conditions; retain control over how data are reused or accessed by third parties.
Data Controllers	Determine purposes and means of data processing; ensure compliance with legal frameworks; implement data protection, accountability, and transparency mechanisms; maintain documentation and conduct risk assessments.	Process data under lawful bases (e.g., consent, contractual necessity, legitimate interest); delegate processing under contractual safeguards.
Data Processors	Process data only on behalf of and under the instructions of the data controller; ensure data security, confidentiality, and technical compliance; assist controllers in fulfilling data subject rights.	Operate under lawful contracts; process data within the limits of the controller's mandate.
Users	Use digital or AI-based systems responsibly and in accordance with terms of service and applicable regulations; report misuse or unlawful content where relevant.	Access secure, transparent, and reliable digital and AI services; ask for correction for unlawful or harmful practices.
Developers and Manufacturers (Providers)	Design and develop AI systems, digital platforms, or products in compliance with regulatory standards; ensure robustness, transparency, and risk management; provide technical documentation and conformity assessments.	Commercialize compliant AI or digital products and services within the EU market; benefit from legal certainty and harmonized conformity frameworks.
Regulatory / Supervisory Authorities	Oversee implementation and enforcement of relevant acts; conduct audits, risk evaluations, and conformity assessments; handle complaints and impose corrective measures or sanctions.	Issue guidance and authorizations; supervise notified bodies; enforce penalties and ensure harmonized application of EU data and AI laws.

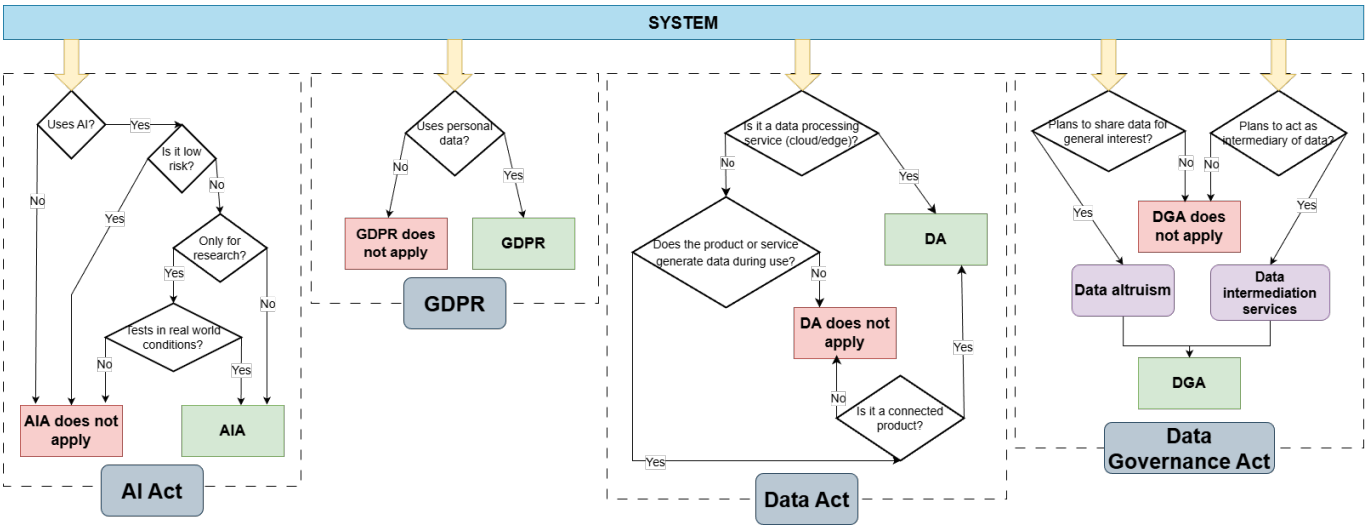


Figure 1. Workflow for assessing the applicability of European regulations.

registrations from countries including Austria, France and the Netherlands, among others. As of the beginning of 2026, 30 companies are listed, but this sector is expected to expand as more providers enter the market and as entities increasingly rely on the neutrality and practicality of these services.

The second main topic of DGA is **data altruism** (Chapter IV). Data altruism is understood as "*the voluntary sharing of data on the basis of the consent of data subjects to process personal data pertaining to them [...] or non-personal data without seeking or receiving a reward [...] for objectives of general interest [...] such as healthcare, combating climate change, improving mobility [...]*" [3]. This concept is central to advancing scientific research and innovation within the EU, as free access to high-quality and diverse data is crucial for generating reliable and impactful results.

D. Artificial Intelligence Act (AIA)

The European Union has taken significant steps in regulating the rapidly-growing field of artificial intelligence. This legislation aims to categorize AI systems based on the levels of risk they pose, ensuring safety and protecting human rights. It also establishes the regulations on general purpose AI systems such as ChatGPT [3].

This act establishes the pillars of AI systems' classification in several categories according to their risk (see Figure 2):

- **Unacceptable risk**: Systems that threaten safety, human lives, or human rights fall into this category and

[3]ChatGPT: https://chatgpt.com/

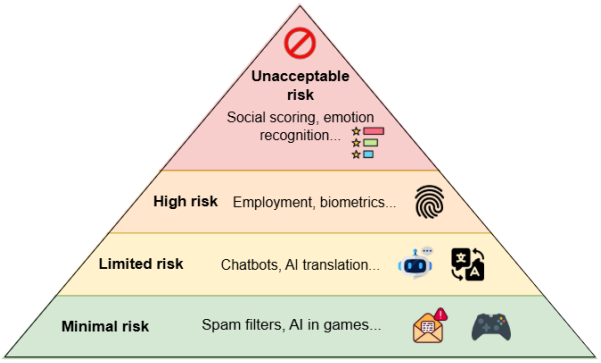


Figure 2. Categorization of AI systems according to their risk.

are strictly prohibited. Examples include social scoring, manipulation of individuals, and emotion recognition in sensitive environments like workplaces or schools.

- **High risk**: These systems are outlined in Annex III of the AIA [4] and are subject to ongoing evaluation. They require stringent oversight due to their potential impact on safety and fundamental rights. Examples include AI systems used in employment, education, access to essential services, and biometric identification.
- **Limited risk**: These applications require transparency to preserve trust (e.g., inform users that they are interacting with an AI system or tool) but do not pose any risk to the security or human rights. Examples include chatbots or translators.
- **Minimal risk**: Solutions that are not subject to any specific rules and cover most of the systems used in the EU, for instance spam filters or AI in video games.

In addition to system-level classification, the AIA regulates **general-purpose AI models (GPAI)**, including so-called *foundation models*, which can be adapted for a wide range of uses. Models that meet specific thresholds of computational power[4] or demonstrate capabilities with systemic impact are designated as **general-purpose AI models with systemic risk** (Article 51). These models are subject to enhanced obligations, including comprehensive risk assessments, evaluation of systemic impacts, documentation, and cybersecurity measures, as defined in Articles 53 and 55.

To encourage innovation, the AIA provides for the creation of **AI regulatory sandboxes** (Article 56). These controlled environments, supervised by national authorities, allow testing and validation of AI systems before market deployment. Additionally, the European Commission and the AI Office will promote the development of **codes of good practice** (Article 7), encouraging voluntary alignment with trustworthy and ethical AI principles beyond mere legal compliance.

The two principles of the AIA are:

- **Obligation and right of transparency**: The AIA explicitly regulates transparency obligations for both providers and users of AI systems, especially those considered high-risk or limited-risk. Article 13 establish that AI systems must be designed and developed to ensure that their operation is sufficiently transparent so that users can interpret and use them appropriately. Providers of high-risk AI systems must include clear instructions for use, describing the system's capabilities, limitations, performance metrics, and potential risks. In addition, limited-risk systems must inform users that they are interacting with an AI system and, when applicable, that content has been artificially generated or manipulated.
- **Obligation and right of reliability**: Although this term is not explicitly defined, the AIA addresses it through requirements for **robustness**, **accuracy**, **data quality**, and **cybersecurity**. Article 15 stipulates that high-risk AI systems must be technically robust, resilient to errors, and secure against manipulation or misuse. They must demonstrate consistent performance across their intended purposes and operate reliably under both expected and reasonably foreseeable conditions. Providers are further required to implement **risk management** (Article 9), **quality management** (Article 17), and **post-market monitoring** (Article 72) mechanisms to preserve reliability throughout the system's lifecycle.

[4]Floating-point operations greater than 10^{25}

IV. Discussion

The GDPR laid the groundwork for personal data processing within the European Union. However, the rapid evolution of AI technologies and the need to promote responsible data use and innovation made additional regulatory measures essential. New instruments such as the DGA, DA, and AIA build on the GDPR's principles. Although these frameworks share conceptual alignment, notable practical tensions emerge, as discussed below.

A. GDPR vs. Data-Centric Frameworks: Points of Divergence

While the GDPR provides a robust framework for personal data protection, the introduction of DA and DGA incorporates new principles aimed at enhancing data sharing and reusability. These objectives, though complementary in theory, often create friction with GDPR's strict consent and purpose limitation requirements. This section examines the key areas where these regulations diverge and the implications for compliance and interoperability.

*1) **GDPR - DA**:* The DA imposes obligations on Data Holders to share or make data available, whereas the GDPR embraces the principle of data minimization (Art 5(1)(c)). The incentive to provide complete and diverse datasets clashes with GDPR's "privacy by design" principle, creating tension between data sharing and minimization.

*2) **GDPR - DGA**:* The DGA promotes altruistic data sharing, intended to support research and other public-interest purposes. However, GDPR provisions, such as the right to withdraw consent (Art. 7(3)) and, where applicable, the right to erasure (Art. 17), limit unrestricted data reuse. While the GDPR prevails and these rights remain fully enforceable whenever personal data are involved, this creates a significant operational challenge for data-altruism ecosystems, particularly regarding effective consent withdrawal and downstream data deletion in multi-actor settings.

Moreover, the GDPR mandates that Data Subjects receive clear and sufficient information about processing activities. In contrast, the DGA ecosystem involves multiple actors and potential future uses of data that are not yet defined,

complicating transparency obligations. This creates a tension between individual transparency and complex, evolving ecosystems.

B. From Data to AI: Interactions and Complementarity Between GDPR and AIA

Next, we discuss how the GDPR and the AIA interact in practice and how their obligations largely converge when AI systems process personal data. Articles 13(2)(f), 14(2)(g), and 15(1)(h) of the GDPR require Data Controllers to provide Data Subjects with meaningful information about automated decision-making. The AIA complements this by imposing transparency and effective human-oversight obligations for high-risk systems (Art. 13–14), ensuring that AI outputs can be interpreted and supervised.

Likewise, the GDPR right not to be subject to a decision based solely on automated processing (Art. 22) is operationalized by the AIA through mandatory risk management and human control. Beyond transparency and oversight, both frameworks address a trade-off between data minimization and algorithmic fairness: larger and more diverse training datasets, including sensitive attributes when strictly necessary, can improve bias mitigation but also increase personal data processing, subject to strict GDPR safeguards. Notably, while the GDPR does not explicitly prohibit algorithmic discrimination, the AIA introduces concrete obligations to identify and mitigate discriminatory risks, illustrating their complementarity.

A remaining challenge concerns the practical enforcement of GDPR rights to erasure and consent withdrawal in AI training. Once personal data contribute to model learning, ensuring effective deletion or "unlearning" raises significant technical and economic difficulties, even though these rights remain fully applicable. Recent guidance from the EDPB (Opinion 28/2024)[5] further stresses the need to assess when AI-driven processing falls within the scope of GDPR Art. 22 and how newly generated AI data should be treated under data-protection law.

C. Opportunities and Compliance Burdens

The introduction of the aforementioned regulatory instruments creates both opportunities and significant compliance challenges:

- **Data intermediation services**: This emerging niche market aims to foster trust and enable secure data sharing among individuals, organizations, and businesses. By acting as neutral intermediaries, they promote broader and more reliable data exchange.
- **Anonymization of data** (*data protection vs data utility*): In terms of the AIA, achieving strong privacy guarantees and high-utility datasets remains an open challenge [14]. To solve these issues, privacy-preserving techniques can be used. Among them, anonymization is one of the most widely adopted [15]. However, its use comes with additional complexities. For example, adding noise to personal data does not always ensure privacy, as aggregation or reconstruction techniques can potentially reverse these modifications, exposing sensitive information. Furthermore, as anonymization becomes more aggressive (e.g., through generalization), model accuracy deteriorates due to information loss, distorting underlying patterns and leading to biased or unreliable outputs [5]. This underscores the technical and ethical challenge of balancing data utility, privacy, and GDPR compliance when training AI systems.
- **Data lifecycle** (*data provenance and privacy*): A secure and well-documented lifecycle for training datasets is essential to demonstrate lawful sourcing and authorization. Measures that limit identifiability—together with strong governance—help maintain trust and regulatory conformity.
- **Dataset quality**: Dataset bias and misrepresentation are also critical concerns addressed in the AIA (Art. 10). Several initiatives aim to mitigate bias in datasets used for AI training. For instance, [10] introduces an open-source Python library offering dataset representations and algorithms for bias detection, mitigation, and metric explainability. This tool, along with others such as FiftyOne[6], has been proposed to support compliance with data governance requirements [11].
- **Compliance complexity** (*compliance automation*) : Although the AIA aims to encourage innovation within defined boundaries, some experts argue that its stringent requirements may have the opposite effect. According to Nizza [8], the regulatory burdens imposed by the AIA risk stifling innovation, particularly for small and medium-sized enterprises (SMEs). This concern is confirmed by the APPRAISE governance framework [9], which highlights the persistent tension between fostering innovation and ensuring compliance. The authors propose a structured approach to help organizations navigate these complexities, emphasizing auditing as a mechanism to prevent "creative compliance" and promote consistency. Their empirical study of 34 organizations in the Netherlands revealed generally low compliance levels, with SMEs performing worst due to assigning lower priority to regulatory conformity. To mitigate these challenges, the AIA offers facilitating measures for SMEs, including free access to regulatory sandboxes, proportional conformity assessment fees, simplified documentation templates, tailored training, and dedicated guidance channels. Penalties and obligations are also applied proportionally to organizational size.
- **Explainability and transparency** (*human understanding and interpretability*): The AIA does not impose a general duty of explainability but promotes it through transparency and documentation obligations, requiring that high-risk AI systems be sufficiently interpretable to enable effective human oversight (Art. 13), even where full algorithmic interpretability is technically infeasible. Despite progress, significant challenges remain, notably the lack of standardized explainability evaluation methods and the need for robust lifecycle monitoring and auditing of high-risk AI systems to ensure safety, accountability, and transparency. Current XAI tools involve

[5]https://www.edpb.europa.eu/our-work-tools/our-documents/opinion-board-art-64/opinion-282024-certain-data-protection-aspects_en

[6]FiftyOne documentation: https://docs.voxel51.com/

trade-offs between fidelity, interpretability, and computational efficiency, and no universally accepted standard exists for evaluating explanation quality [6]. Moreover, AI reliability requires not only transparency but also data quality, bias mitigation, robustness, and stability [7]. Frameworks integrating these dimensions are therefore essential for trustworthy AI. Parlak's BAXDT approach [12] enhances transparency by recording explainable decision traces, but its reliance on a single technique may limit comprehensiveness; combining it with methods such as LIME [13] or other XAI strategies could support more holistic interpretability.

Ultimately, these challenges underscore the need for a practical compliance framework that is beneficial for all stakeholders—data subjects, enterprises, and regulators alike. Such a framework should harmonize GDPR principles with the operational requirements of the AIA, DA, and DGA, ensuring that privacy, transparency, and innovation can coexist without imposing disproportionate burdens.

V. Conclusions

This study analyzed the interplay between the GDPR, the AIA, the DA, and the DGA, focusing on their points of convergence and interaction. We conducted a comparative analysis of these frameworks, identifying areas where principles align and where significant practical tensions arise (such as the tension between data altruism and consent withdrawal, or between data-sharing obligations and privacy-by-design requirements). While these regulations mark a significant step toward trustworthy AI by promoting transparency, accountability, and societal benefit, their effective implementation remains challenging.

Moreover, the parallel application of these frameworks implies the involvement of multiple supervisory authorities (data protection, data governance, and AI regulators), raising risks of regulatory overlap, fragmentation, and competence gaps.

Key issues include the gap between regulatory principles and practical feasibility, the difficulty of balancing privacy with data utility and model accuracy, and the complexity of ensuring consistent compliance across overlapping frameworks. Additionally, overcoming the opacity of "black-box" models remains critical: despite progress in explainability techniques, the lack of standardized methodologies and practitioner training limits their effectiveness.

VI. Future Directions

Future work should focus on data provenance in AI training, effective consent management, and practical transparency and explainability mechanisms. It should also examine the interaction of the DA's international data-transfer regime and the AIA cybersecurity requirements with broader EU cyber-risk frameworks, including NIS2 and the Cyber Resilience Act, to ensure coherent and efficient compliance.

Acknowledgments and Disclaimer

This research is supported by Project HERMES, funded by the European Union NextGenerationEU/PRTR via INCIBE; by Projects PID2021-125962OB-C32 "SECURING/DATA" and PID2024-156914OB-C42 "SAFE/DATA", funded by MCIN/AEI/ 10.13039/501100011033; and by Grant SGR2021-00115, funded by AGAUR, Generalitat de Catalunya. Ignacio Miguel Rodríguez is supported by the Generalitat of Catalonia under a FI-STEP grant (ref. 2025 STEP 00378).

References

[1] European Parliament and Council, "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation)," *Official Journal of the European Union*, 2016. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng

[2] European Parliament and Council, "Regulation (EU) 2023/2854 of the European Parliament and of the Council of 13 December 2023 on harmonised rules on fair access to and use of data and amending Regulation (EU) 2017/2394 and Directive (EU) 2020/1828 (Data Act)", *Official Journal of the European Union*, 2023.https://eur-lex.europa.eu/eli/reg/2023/2854/oj/eng

[3] European Parliament and Council, "Regulation (EU) 2022/868 of the European Parliament and of the Council of 30 May 2022 on European data governance and amending Regulation (EU) 2018/1724 (Data Governance Act)", *Official Journal of the European Union*, 2022. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2022/868/oj/eng

[4] European Parliament and Council, "Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence," *Official Journal of the European Union*, 2024. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng

[5] H. Lee, S. Kim, J. W. Kim, and Y. D. Chung, "Utility-preserving anonymization for health data publishing," *BMC Medical Informatics and Decision Making*, vol. 17, no. 1, p. 104, Jul. 2017, doi: 10.1186/s12911-017-0499-0. [Online]. Available: https://doi.org/10.1186/s12911-017-0499-0

[6] H. I. Aysel, X. Cai, and A. Prugel-Bennett, "Explainable artificial intelligence: Advancements and limitations," *Applied Sciences*, vol. 15, no. 13, art. no. 7261, 2025, doi: 10.3390/app15137261. [Online]. Available: https://www.mdpi.com/2076-3417/15/13/7261

[7] S. T. H. Mortaji and M. E. Sadeghi, "Assessing the reliability of artificial intelligence systems: Challenges, metrics, and future directions," *IJIMES*, vol. 4, no. 2, pp. 1–13, Jun. 2024, doi: 10.59615/ijimes.4.2.1. [Online]. Available: https://doi.org/10.59615/ijimes.4.2.1

[8] U. Nizza, "Assessing the Impact of the European AI Act on Innovation Dynamics," *Transatlantic Technology Law Forum Working Paper*, no. 138, 2025. [Online]. Available: https://wwws.law.northwestern.edu/research-faculty/clbe/events/standardization/documents/nizza_assessing_impact_ai_act_innovation.pdf

[9] D. Dey and D. Bhaumik, "APPRAISE: a Governance Framework for Innovation with Artificial Intelligence Systems," *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, vol. 7, no. 1, pp. 328–340, Oct. 2024, doi: 10.1609/aies.v7i1.31640. [Online]. Available: https://ojs.aaai.org/index.php/AIES/article/view/31640

[10] R. K. E. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic, S. Nagar, K. N. Ramamurthy, J. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, and Y. Zhang, "AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias," *arXiv preprint*, Oct. 2018. [Online]. Available: https://arxiv.org/abs/1810.01943

[11] A. Obradov and B. Pavković, "Ensuring Compliance of High-Risk AI Systems with the EU AI Act Using Open Source Libraries," *2025 IEEE Zooming Innovation in Consumer Technologies Conference (ZINC)*, pp. 54–59, 2025, doi: 10.1109/ZINC65316.2025.11103517.

[12] İ. E. Parlak, "Blockchain-assisted explainable decision traces (BAXDT): An approach for transparency and accountability in artificial intelligence systems," *Knowledge-Based Systems*, vol. 329, pp. 114402, 2025, doi: 10.1016/j.knosys.2025.114402. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705125014418

[13] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You?: Explaining the Predictions of Any Classifier," arXiv preprint arXiv:1602.04938, 2016. Available: https://arxiv.org/abs/1602.04938.

[14] H. Zhong and K. Bu, *Privacy-Utility Trade-Off*, arXiv:2204.12057, 2022. Available at: https://arxiv.org/abs/2204.12057.

[15] A. Majeed, *Attribute-Centric and Synthetic Data Based Privacy Preserving Methods: A Systematic Review*, *Journal of Cybersecurity and Privacy*, vol. 3, pp. 638–661, 2023. Available at: https://doi.org/10.3390/jcp3030030.

Fusión Audio-Visual con Mecanismos de Atención para la Detección de Contenido Sensible en Vídeo

Daniel Povedano Álvarez, Sergio Muñoz Martín, Ana Lucila Sandoval Orozco, Luis Javier García Villalba*

Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {dpovedano, asandov, javierv}@ucm.es

***Resumen*—Detectar contenido sensible en vídeos en línea es fundamental para mantener entornos digitales seguros y garantizar una moderación efectiva. Este trabajo presenta *Multimodal Audio-Visual Cross-Attention* (MAV-XAtt), un modelo de aprendizaje profundo que integra información de audio y vídeo mediante mecanismos de atención para identificar contenido sensible. Esta propuesta constituye una continuación natural de una línea de investigación previa centrada en técnicas multimodales para la detección de contenido sensible. La principal contribución consiste en alinear características de ambas modalidades, capturando información complementaria y dependencias temporales entre los fotogramas mediante *cross-attention* . MAV-XAtt se evaluó con el conjunto LSPD, compuesto por 2 000 vídeos seguros y 2 000 sensibles. Utilizando una pérdida combinada que equilibra los objetivos de audio, visual y fusión, el modelo alcanzó un *F1-score* de 96,53 % en segmentos audio-visuales y 95,5 % en vídeos completos, demostrando su eficacia y eficiencia.**

***Index Terms*—Mecanismos de atención, Fusión audio-visual, Aprendizaje multimodal, Detección de pornografía.**

I. Introducción

En los últimos años, la proliferación de dispositivos móviles y redes sociales ha provocado un aumento significativo de contenido ilegal en Internet, intensificado por dinámicas derivadas de la pandemia. Esto incluye material sensible y generado por IA que pone en grave riesgo a menores [1], [2]. Paradójicamente, aunque la tecnología facilita la difusión de Material de Abuso Sexual Infantil (MASI), el uso de Inteligencia Artificial (IA) presenta un gran potencial para combatir este delito y mejorar la protección infantil. [3], [4].

En este contexto, la detección automatizada de contenido sensible (desnudez) y MASI es prioritaria para el análisis forense, aunque las restricciones legales exigen sistemas que operen sin acceso directo a los datos originales [5]. Si bien los métodos multimodales han mejorado el rendimiento, aún enfrentan dificultades cuando el contenido explícito no es visualmente evidente, generando falsos negativos [6]. Para mitigar esto, el análisis de audio ha demostrado ser complementario y eficaz en casos visualmente ambiguos [7], [8].

No obstante, muchos métodos actuales se limitan a estrategias de concatenación simple, desaprovechando la interacción entre flujos de audio y vídeo. La integración de mecanismos de *cross-attention* se perfila como una solución necesaria para modelar dinámicamente estas interacciones, cubriendo un vacío crítico en la literatura [9], [10].

Este trabajo contribuye al desarrollo de sistemas de detección más eficaces mediante un enfoque multimodal que integra características auditivas y visuales, mejorando la identificación de contenido sensible y reduciendo los falsos negativos.

El artículo se organiza en seis secciones. La Sección I introduce el estudio. La Sección II revisa el estado del arte, mientras que el conjunto de datos y metodología propuesta se detallan en las secciones III y IV respectivamente. Finalmente, la Sección V presenta los experimentos y resultados, cerrando con las conclusiones en la Sección VI.

II. Trabajos relacionados

Recientemente, los mecanismos de atención se han aplicado ampliamente en modelos multimodales, mejorando la integración de características visuales y de audio en tareas complejas. De manera similar, los avances en Aprendizaje Profundo han elevado la precisión y eficiencia en la detección de contenido sexual, combinando claves visuales y acústicas para identificar y clasificar MASI en línea, facilitando la prevención de material ilegal.

II-A. Características visuales

Esta sección resume los principales enfoques basados en características visuales mediante modelos de aprendizaje profundo. La mayoría de los trabajos emplean CNNs y arquitecturas avanzadas para extraer automáticamente patrones visuales relevantes, logrando mejoras sustanciales frente a métodos tradicionales.

Moustafa et al. [11] utilizaron GoogLeNet y AlexNet ajustados con datos pornográficos, alcanzando un 94,1 % de precisión en Pornography-800 [12]. Otros trabajos obtuvieron resultados similares o superiores: Nian et al. [13] lograron un 98.6 %, Pérez et al. [14] un 97,9 % y Silva y Marana [15] un 95,1 % en vídeo. Modelos recientes, como PPCensor [16], AttM-CNN [17] y la canalización basada en ResNet-18 [18], alcanzan entre un 97 % y un 99 % en Pornography-2k y Pornography-800.

Enfoques más nuevos exploran la transferencia de aprendizaje, como el método de Samal et al. [19] (98,50 % en NPDI y Pornography-2k), y modelos semisupervisados como DeepHSAR [20], que alcanza un 99,85 % en Pornography-2k. Estudios recientes también analizan comparativas amplias de clasificadores [21] y modelos multimodales como AM3 [9], que obtiene un 92,86 %. Finalmente, enfoques zero-shot basados en VLMs, como el propuesto por Phan et al. [22],

alcanzan un 91,6 % sobre el conjunto de datos LSPD sin necesidad de entrenamiento específico, reduciendo la dependencia de grandes conjuntos de datos anotados.

II-B. Características de audio

La detección de contenido sensible en archivos de audio sigue siendo un reto importante y requiere técnicas avanzadas de aprendizaje automático. Esta sección resume las principales contribuciones recientes basadas en características acústicas para la identificación de material sensible.

Lovenia et al. [23] emplearon CNNs entrenadas sobre espectrogramas log Mel para la detección pornográfica, obteniendo puntuaciones F1 de 94,89 % por segmento y 92,02 % a nivel de audio en Pornography-800. Zhou et al. [24] propusieron un modelo basado en CNN con mecanismos de atención, logrando una precisión del 97,19 % en un conjunto de datos de más de 498.000 muestras. Liu et al. [25] presentaron DPFTNet, una arquitectura que combina log Mel, MFCC y GFCC, con una precisión del 93,20 % y una F1 del 93,56 % en un conjunto de 7.942 audios.

II-C. Características multimodales

La integración de características visuales y de audio en marcos multimodales ha permitido procesar información más rica y compleja, mejorando la capacidad de los modelos para entender contextos desafiantes. Esta sección resume los principales avances en la detección de contenido sensible mediante enfoques que combinan múltiples modalidades.

Song et al. [26] exploraron características multimodales (audio, imagen, movimiento y flujo óptico), logrando precisiones del 88,3 % y 80 % para las modalidades de movimiento y audio. Wang et al. [27] abordaron la detección en transmisiones de vídeo en vivo con una precisión del 74,86 %. Song et al. [8] propusieron un esquema de apilamiento multimodal con una precisión del 92,33 %. Fu et al. [7] presentaron PornNet, que combina audio e imagen con mecanismos de atención, alcanzando un 93,4 %. Zhao et al. [28] desarrollaron Porn2Vec, logrando una F1 del 99,25 % en la detección de sitios web pornográficos. Estudios más recientes [29], [30], [31] emplean atención cruzada y enfoques basados en transformadores, mejorando la detección en escenarios ambiguos y fomentando estrategias de fusión transmodal que fortalecen la comprensión contextual. La Tabla II resume las técnicas y resultados más destacados de estudios recientes. La adopción de enfoques multimodales que integran información visual, auditiva y textual ha impulsado mejoras significativas en la detección, especialmente en escenarios difíciles con contenido ambiguo o poca presencia de desnudez. Siguiendo esta línea, los trabajos más actuales exploran arquitecturas basadas en transformadores y métodos de fusión transmodal que fortalecen la interpretación contextual entre diferentes modalidades.

III. Conjunto de Datos

La preparación del conjunto de datos es crucial para el entrenamiento supervisado, y en esta ocasión, no lo es menos. Se ha seleccionado el conjunto de datos *Large Scale Pornography Dataset* (LSPD) [32] debido a sus características únicas, que incluyen etiquetas a nivel de clase y máscaras poligonales para identificar objetos sexuales privados en imágenes y vídeos. Con un corpus de 500.000 imágenes y 4.000 vídeos, el conjunto es robusto para el entrenamiento y la evaluación, al estar compuesto por 2.000 vídeos seguros y 2.000 vídeos de contenido sexual. La Tabla I resume la cantidad de vídeos por clase y la disponibilidad de audio (* indica que un vídeo de la clase sexual no pudo extraerse por encontrarse dañado).

Tabla I
Estadísticas de los vídeos en el conjunto de datos LSPD.

Clase	Total de vídeos	vídeos con audio	vídeos sin audio
Seguro	2.000	641	1.359
Pornografía	1.999*	1.660	339
Total	3.999	2.301	1.698

Tabla II
Trabajos más relevantes relacionados con técnicas de Aprendizaje Profundo.

Ref.	Conjunto de datos	Características	Arquitectura	Métrica (%)
[8]	Pornography-2k	Multimodal	EMSS	Acc: 92,33
[7]	100.000 muestras	Multimodal	RANet + DCNet	Acc: 93,4
[17]	Pornography-2k	Visual	AttM-CNN	F2: 97,45
[23]	NPDI-800	Audio	CNN	F1: 92,02
[25]	7.942 muestras	Audio	DPFTNet	F1: 93,56
[19]	NPDI-800	Visual	TLMobDense	F1: 99,12
	Pornography-2k	Visual	TLMobDense	F1: 98,32
[20]	Pornography-2k	Visual	DeepHSAR	Acc: 99,85
[22]	LSPD	Visual	VLM	Acc: 96,1

IV. Método propuesto: MAV-XAtt

La arquitectura propuesta, MAV-XAtt, integra modalidades de audio y visual mediante un mecanismo explícito de *cross-attention*, que permite una fusión multimodal más efectiva. Cada rama incorpora *self-attention*: en audio, para modelar dependencias temporales y frecuenciales en el espectrograma, y en visual, para capturar la dinámica temporal a partir de *keyframes* secuenciales. Esta interacción cruzada entre modalidades mejora la discriminación de contenido multimedia sensible.

A diferencia de gran parte del estado del arte, evaluado sobre Pornography-2k, este trabajo emplea el conjunto LSPD, más diverso y desafiante, con aproximadamente el doble de vídeos y mayor variabilidad de escenas y duraciones, lo que permite una evaluación más realista y reduce sesgos asociados a conjuntos de datos homogéneos. Además, frente a enfoques basados en fusión temprana o tardía, MAV-XAtt modela directamente las interacciones multimodales mediante *cross-attention*.

La arquitectura opera de extremo a extremo (*end-to-end*), optimizando conjuntamente las representaciones de audio y visual mediante *backpropagation*.

El procesamiento comienza con la etapa *Media Unit Segment* (Figura 1), donde los vídeos se dividen en *keyframes* (segmentos visuales) y segmentos de audio de 20, 60 o 90 segundos. Para reducir la redundancia visual, se aplicó una técnica de reducción de redundancia mediante el algoritmo *perceptual hashing* (pHash) durante el preprocesamiento, filtrando *frames* casi duplicados y asegurando un conjunto más diverso y representativo de *keyframes*. Paralelamente, se realizó una alineación temporal entre los segmentos de audio y visual para asegurar entradas sincronizadas, esencial para una fusión multimodal efectiva. Finalmente, se estableció una

función de pérdida compuesta que optimiza conjuntamente las ramas de audio, visual y *cross-attention*, promoviendo contribuciones equilibradas de cada modalidad.

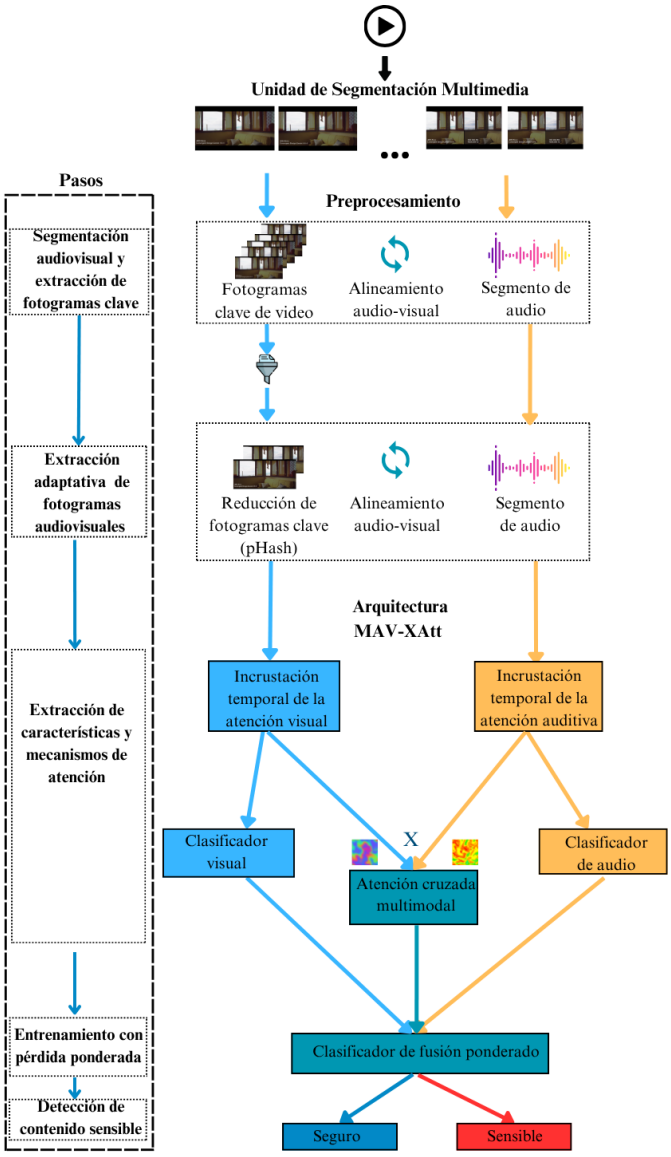


Figura 1. Descripción de la arquitectura y el flujo de trabajo de MAV-XAtt.

La arquitectura cuenta con los siguientes elementos:

1. **Preprocesamiento**: Se implementó un *pipeline* de preprocesamiento para explotar las características audiovisuales del conjunto de datos, preparando los vídeos para la extracción de características y análisis posterior. Se incluyeron limpieza, reducción de ruido, normalización y sincronización de audio y visuales, así como técnicas de aumento de datos como recorte y volteo aleatorio.
   a) Segmentación temporal: los vídeos se dividieron en clips de 20 y 60 segundos, siguiendo prácticas de referencia [23] (se probaron también segmentos de 90 segundos para explorar contextos temporales más largos), extrayendo segmentos de silencio cuando los vídeos no contenían pista de audio,de esta manera, el modelo puede aprender la representación visual de los vídeos que no cuenten con canal auditivo.
   b) Extracción de *keyframes*: se obtuvieron los *keyframes* (I-Frames) de cada segmento, capturando el contenido visual más relevante.
   c) Extracción de características: se extrajeron características de audio y visuales de los clips de vídeo (los elementos visuales se alinearon con cada segmento de audio para garantizar sincronización temporal entre modalidades):
   - Audio: procesado para análisis auditivo.
   - Visual: *keyframes* analizados para patrones espaciales.
2. **Audio a espectrogramas**: El audio se digitalizó mediante muestreo a 16–44.1 kHz, obteniendo una señal discreta adecuada para análisis. Esta señal se transformó en un espectrograma log Mel para capturar patrones de frecuencia en una representación 2D, permitiendo su integración en la arquitectura visual.
3. **Extracción de características**: El modelo ConvNeXt V2 -femto-, usado como *backbone*, combina las ventajas de Vision Transformers y CNNs, modernizando ResNet con *layer normalization* y convoluciones eficientes. Este modelo ofrece resultados de vanguardia y es escalable para tareas multimodales. La entrada de la rama visual (*keyframes*) se redimensionó a 224×224, mientras que la rama auditiva consistirá en extraer el espectrograma log-Mel de cada audio. La atención (*self-attention*) refinará los patrones temporales y semánticos, mientras que la codificación posicional (*positional encoding*) preservará el orden temporal de las características visuales.
   - Las características de audio codifican patrones temporales y espectrales auditivos mediante codificadores basados en convoluciones o transformadores.
   - Las características visuales representan información espacial y semántica extraída de los *keyframes*.
4. **Atención unimodal**: Cada modalidad aplica *self-attention* para enfatizar sus regiones más informativas, como partes del cuerpo u objetos en visual, y patrones de frecuencia significativos en audio.
5. **Cross-attention**: Para modelar interacciones entre diferentes modalidades, los *embeddings* de audio sirvieron como consultas (Q) y los visuales como claves y valores (K, V), donde d_k es la dimensión de la clave. Esto permite que cada modalidad atienda selectivamente señales relevantes de la otra. La atención cruzada (*cross-attention*) combina por tanto elementos del canal auditivo y visual según relevancia, y clasificadores independientes generarán las predicciones intermedias que se fusionan mediante un clasificador ponderado por pérdida. Para concluir, se utilizaron operaciones de *pooling* para reducir la dimensionalidad manteniendo la información esencial.
6. **Fusión y optimización**: Se combinan las salidas de:
   - La rama de audio,
   - La rama visual,
   - La representación fusionada audio-visual.

   Cada rama contribuye a la predicción final, guiada por la función de pérdida conjunta.

Finalmente, un *MultiLayer Perceptron* (*MLP*) genera la

salida binaria a partir de la fusión de las tres ramas, reflejando la integración de la información audio-visual.

V. Experimentos

V-A. Protocolo de evaluación

Se empleó un esquema de validación cruzada de 5 *k-folds* para evaluar el desempeño de cada modelo, garantizando una evaluación exhaustiva. Los hiperparámetros y configuraciones de los algoritmos se detallan en esta sección. Tras la selección del mejor modelo, las predicciones de vídeos se realizaron mediante votación mayoritaria sobre cada segmento audio-visual (audio y fotogramas.)

Los experimentos se llevaron a cabo sobre un conjunto de datos anotados y preprocesados de 3 999 vídeos, dividido en subconjuntos de entrenamiento y prueba con una proporción 80/20, asegurando balance y evitando filtraciones de datos entre conjuntos. Los segmentos audio-visuales fueron seleccionados según los pasos de preprocesamiento, y los experimentos se repitieron en los cinco *folds*, reportando los resultados promedio.

Se emplearon las métricas de exactitud (*accuracy*), precisión (*precision*), sensibilidad (*recall*) y F1-score para evaluar los modelos:

V-B. Detalles de la implementación

En el entrenamiento se utilizó el optimizador *AdamW* con tasa de aprendizaje 1×10^{-4}, decaimiento 1×10^{-3}, mini-batch de 32, 50 épocas y *early stopping* con paciencia de 5 para evitar el *overfitting*.

Los *keyframes* se redimensionaron a 224×224, se normalizaron con los parámetros de ImageNet y se emplearon para el ajuste fino del modelo preentrenado *ConvNeXt-femto*. El audio se convirtió a espectrogramas log-Mel (16 kHz, $N_{\text{FFT}} = 1024$, $N_{\text{MELS}} = 224$, *hop length*=512) y se normalizó de forma consistente. Para mejorar la robustez, se aplicaron técnicas de *data augmentation*. En imágenes se utilizaron transformaciones geométricas y de degradación visual, mientras que en audio se incorporaron ruido aditivo, cambios de tono y filtros pasa-altos y p

V-C. Experimentos preliminares sobre la duración óptima de los segmentos

Se realizaron experimentos preliminares sin mecanismos de atención, sin *data augmentation* y sin *fine-tuning*, con el fin de evaluar el efecto de la duración del segmento y establecer métricas de referencia. Se emplearon segmentos audio-visuales de 20, 60 segundos, siguiendo la motivación de [23] y ampliando el rango para cubrir contextos más largos. La Tabla III recoge las métricas obtenidas mediante validación cruzada.

Los resultados mostraron que los modelos unimodales visuales superan claramente a los de audio (p. ej., 91,03 % frente a 80,8 % en 20 s), indicando que la rama visual aporta información más relevante. No obstante, el audio sigue siendo complementario: la fusión mejora sistemáticamente cada modalidad por separado.

En los modelos multimodales, las configuraciones con ponderación explícita de pérdidas ofrecen el mejor rendimiento. Asignar pesos $0{,}1$ (audio), $0{,}3$ (imagen) y $0{,}5$ (fusión) produce los F1 más altos tanto en 20 s como en 60 s, superando en torno a un punto la mejor modalidad unimodal.

En cuanto a la duración de los segmentos, con la configuración óptima se obtuvieron 92,28 % (20 s) y 92,43 % (60 s). Aunque ventanas más largas pueden aportar contexto adicional, también incrementan el coste computacional y la posibilidad de añadir ruido al modelo; por ello, esta configuración se reutilizó posteriormente para evaluar segmentos de 90 s sin reajustar hiperparámetros.

A continuación se presentan los experimentos realizados para refinar las configuraciones *baseline* para segmentos de 20, 60 y 90 segundos. A diferencia de la fase previa —con codificadores congelados y sin aumento de datos ni atención—, aquí se analizó el impacto de ajustar selectivamente capas del codificador de audio e imagen, incorporando además estrategias de aumento y mecanismos de *self-attention* y *cross-attention*.

Los experimentos, resumidos en la Tabla IV, varían en tres ejes: (i) número de capas entrenables en el codificador de audio, (ii) número de capas entrenables en el codificador de imagen y (iii) uso de aumento de datos (estiramiento temporal y ruido en audio; recortes aleatorios y *colour jitter* en imagen).

El experimento inicial, con ambos codificadores congelados, ya mejora el *baseline* previo (logrando un 92,87 %). Al descongelar más capas del codificador visual que del auditivo (audio 15, imagen 10), el rendimiento aumenta, alcanzando su mejor variante con la configuración (audio 10, imagen 15; 94,21 %), lo que evidencia la mayor aportación del ajuste fino visual en segmentos cortos.

Para clips largos (60 segundos), el rendimiento con codificadores congelados asciende a 94,90 %, reflejando el mayor valor del contexto temporal. La configuración —diez capas de audio y quince de imagen, con aumento de datos— logró el mejor rendimiento alcanzando un 96,23 %, indicando que el ajuste fino se beneficia de segmentos de un minuto.

La variante de 90 segundos, análoga a la mejor configuración de 60s, obtuvo un 94,96 %, superando todas las de 20 segundos pero sin alcanzar el máximo logrado en 60. Esto sugiere que las ganancias por aumentar la duración se reducen tras el *fine-tuning*.

En general, el aumento de datos mejoró la estabilidad y descongelar capas del codificador visual sigue siendo más beneficioso que hacerlo en el acústico. Una cantidad moderada de *fine-tuning*, combinada con aumento de datos, ofrece mejoras sustanciales manteniendo una carga computacional razonable. Finalmente, aunque los segmentos largos aportaron mayor coherencia semántica bajo *weak labels*, los 60 segundos representan el mejor equilibrio global.

V-D. Evaluación del modelo de mejor desempeño

Para evaluar la mejor configuración de MAV-XAtt, se utilizaron segmentos audio-visuales de 60 segundos, que proporcionan más *keyframes* y enriqueciendo las características del modelo. Inicialmente, se realizaron cuatro experimentos sin mecanismos de atención ni aumento de datos para establecer métricas de referencia y comprender la contribución de audio, imagen y la fusión de ambas, guiando la configuración de la función de pérdida combinada para el *fine-tuning* posterior.

La evaluación en el conjunto de prueba confirma la robustez de la configuración de 60 segundos seleccionada en el

Tabla III
RESUMEN DE LOS EXPERIMENTOS *baseline* Y SUS VALORES MEDIOS DE *F1-score* EN VALIDACIÓN CRUZADA.

Segmento (s)	Experimento	Peso audio	Peso imagen	Peso fusión	*F1-score (%)*
20	Solo audio	-	-	-	80,8
	Solo imagen	-	-	-	91,03
	Audio + Imagen + Fusión (pérdida de fusión)	-	-	-	92,18
	Audio + Imagen + Fusión (pérdida combinada)	1	1	1	92,04
	Audio + Imagen + Fusión (pérdida combinada)	0,1	0,2	0,4	92.26
	Audio + Imagen + Fusión (pérdida combinada)	**0,1**	**0,3**	**0,5**	**92,28**
60	Solo audio	-	-	-	80,91
	Solo imagen	-	-	-	91,26
	Audio + Imagen + Fusión (pérdida de fusión)	-	-	-	92,23
	Audio + Imagen + Fusión (pérdida combinada)	**0.1**	**0,3**	**0,5**	**92,43**

Tabla IV
Fine-tuning DEL MEJOR MODELO: IMPACTO DE LAS CAPAS ENTRENABLES Y DEL AUMENTO DE DATOS.

Experiment	Segmento	Capas entrenables audio	Capas entrenables imagen	Aumento de datos	*F1-score (%)*
1	20	0	0	No	92,87
2	20	15	10	Sí	93,86
3	20	10	15	Sí	94,21
4	60	0	0	No	94,90
5	**60**	**10**	**15**	**Sí**	**96.23**
6	90	10	15	Sí	94,96

Experimento 5. Como se muestra en la Tabla V, el modelo alcanza una precisión global de 96,53 % y un F1-score macro de 96,49 %, demostrando un rendimiento equilibrado a pesar del ligero desbalance entre clases (sensible: 2.072; seguro: 1.621).

Tabla V
INFORME DE CLASIFICACIÓN CON PRECISIÓN, *recall*, *F1-score* Y SOPORTE.

Clase	Precisión	Sensibilidad	*F1-score*	Soporte
Seguro	94,8	97,41	96.1	1621
Sensible	97,9	95,85	96,9	2072
Exactitud (*Acc.*)			96.53	3693
Promedio Macro	96,4	96,65	96,49	3693
Promedio Ponderado	96,66	96,65	96,49	3693

V-E. Rendimiento de clasificación de vídeos basado en segmentos audio-visuales de 60 segundos

Para predicciones a nivel de vídeo, el modelo entrenado con segmentos de 60 segundos mostró el mejor desempeño frente a los entrenados con segmentos de 20 y 90 segundos (Tabla VI). La precisión alcanzada fue de 94,82 % para la clase segura y 96,16 % para la clase sensible, con sensibilidad de 96,25 % y 94,24 % respectivamente. Los *F1-scores* fueron 95,53 % para seguro y 95,48 % para sensible, reflejando un comportamiento de clasificación equilibrado y fiable. Estos resultados indican que los segmentos de mayor duración permiten capturar patrones temporales más amplios, incluyendo transiciones y elementos contextuales importantes para casos ambiguos. La configuración de 60 segundos ofrece el mejor compromiso entre contexto y complejidad, superando a segmentos de 20 y 90 segundos. La matriz de confusión (Figura 2) presenta una marcada dominancia diagonal y una reducción de errores, confirmando la robustez del modelo en escenarios de vídeo reales con información distribuida temporalmente.

¡

VI. CONCLUSIONES

Este estudio demuestra que la combinación de modalidades de audio y visual mediante mecanismos de atención cruzada mejora significativamente la clasificación de contenido

Tabla VI
INFORME DE CLASIFICACIÓN DE VÍDEOS CON PRECISIÓN, *recall*, *F1-score* Y SOPORTE USANDO SEGMENTOS DE 60 SEGUNDOS.

Clase	Precisión	Sensibilidad	*F1-score*	Soporte
Safe	94,82	96,25	95,53	400
Sensitive	96,16	94,24	95,48	399
Exactitud			95,5	799
Promedio Macro	95,5	95,5	95,5	799
Promedio Ponderado	95,5	95,5	95,5	799

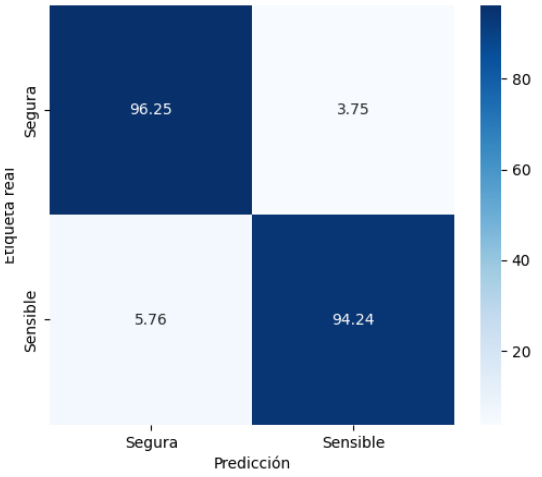


Figura 2. Matriz de confusión de vídeos usando el mejor modelo con segmentos de 60 segundos sobre el conjunto de prueba.

sensible en vídeo. MAV-XAtt, entrenado *end-to-end* sobre secuencias de 60 segundos, alcanza un *F1-score* macro del 96,49 % y una exactitud del 96,53 % sobre LSPD, un conjunto más diverso y exigente que Pornography-2k. Estos resultados, comparables o superiores a los del estado del arte, evidencian la capacidad de generalización del modelo y su idoneidad para escenarios realistas de moderación automática de contenido.

Se identificaron limitaciones importantes: la dependencia de ambas modalidades puede afectar la exactitud en vídeos silenciosos o de baja resolución, y la interpretabilidad del modelo sigue siendo un desafío. Para abordar estas cuestiones, se proponen futuras líneas de trabajo:

(a) adaptar visualizaciones estilo Grad-CAM a espectrogra-

mas de audio,

(b) inspeccionar las matrices de pesos de *cross-attention* para identificar los pasos temporales más relevantes,

(c) explorar modelos de tipo *Visual Language Model* (VLM) y otros modelos multimodales basados en LLMs.

Otras mejoras potenciales incluyen optimizar estrategias de fusión multimodal, explorar *backbones* de audio alternativos, aplicar ponderación dinámica de pérdidas y entrenar con muestras de audio intercambiado para aumentar la resiliencia frente a desajustes audio-visuales.

AGRADECIMIENTOS

Este trabajo se ha realizado con los fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra "Ciberseguridad para la Innovación y la Protección Digital" INCIBE-UCM. Además, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, proyecto CIRMA-CM (TEC-2024/COM-404). El contenido de este artículo no refleja la opinión oficial de la Unión Europea. Los autores son los únicos responsables de la información y las opiniones expresadas en él.

REFERENCIAS

[1] S. Ramaswamy and S. Seshadri, "Children on the brink: Risks for child protection, sexual abuse, and related mental health problems in the covid-19 pandemic," *Indian Journal of Psychiatry*, vol. 62, no. Suppl 3, p. S404, 2020.

[2] E. Kokolaki and P. Fragopoulou, "Unveiling ai's threats to child protection: Regulatory efforts to criminalize ai-generated csam and emerging children's rights violations," 2025. [Online]. Available: https://arxiv.org/abs/2503.00433

[3] H. Wolbers, T. Cubitt, and M. Cahill, "Artificial intelligence and child sexual abuse: A rapid evidence assessment," Australian Institute of Criminology, Tech. Rep. 711, 2025. [Online]. Available: https://www.aic.gov.au/publications/tandi/tandi711

[4] Europol, "AI and policing: The benefits and challenges of artificial intelligence for law enforcement," Publications Office of the European Union, Tech. Rep., 2023. [Online]. Available: https://www.europol.europa.eu/cms/sites/default/files/documents/Europol_AI_and_Policing_Report_2023.pdf

[5] J. Cifuentes, A. L. Sandoval Orozco, and L. J. García Villalba, "A survey of artificial intelligence strategies for automatic detection of sexually explicit videos," *Multimedia Tools and Applications*, vol. 81, no. 3, pp. 3205–3222, 2022.

[6] D. Povedano Álvarez, A. L. Sandoval Orozco, J. P. García-Miguel, and L. J. García Villalba, "Learning strategies for sensitive content detection," *Electronics*, vol. 12, no. 11, p. 2496, 2023.

[7] Z. Fu, J. Li, G. Chen, T. Yu, and T. Deng, "Pornnet: A unified deep architecture for pornographic video recognition," *Applied Sciences*, vol. 11, no. 7, p. 3066, 2021.

[8] K. Song and Y. S. Kim, "An enhanced multimodal stacking scheme for online pornographic content detection," *Applied Sciences (Switzerland)*, vol. 10, p. 2943, 2020.

[9] J. Yuan, Y. Yu, G. Mittal, M. Hall, S. Sajeev, and M. Chen, "Rethinking multimodal content moderation from an asymmetric angle with mixed-modality," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 8532–8542.

[10] N. Sun, J. Jin, L. Meng, W. Lin, H. Wang, L. Liu, and H. Zhang, "Mfcqa: Multi-range feature cross-attention mechanism for no-reference image quality assessment," *Knowledge-Based Systems*, p. 113027, 2025.

[11] M. N. Moustafa, "Applying deep learning to classify pornographic images and videos," *CoRR*, vol. abs/1511.08899, 2015. [Online]. Available: http://arxiv.org/abs/1511.08899

[12] S. Avila, N. Thome, M. Cord, E. Valle, and A. De A. Araújo, "Pooling in image representation: The visual codeword point of view," *Computer Vision and Image Understanding*, vol. 117, no. 5, pp. 453–465, 2013.

[13] F. Nian, T. Li, Y. Wang, M. Xu, and J. Wu, "Pornographic image detection utilizing deep convolutional neural networks," *Neurocomputing*, vol. 210, pp. 283–293, 2016.

[14] M. Perez, S. Avila, D. Moreira, D. Moraes, V. Testoni, E. Valle, S. Goldenstein, and A. Rocha, "Video pornography detection through deep learning techniques and motion information," *Neurocomputing*, vol. 230, no. March, pp. 279–293, 2017.

[15] M. V. da Silva and A. N. Marana, "Spatiotemporal cnns for pornography detection in videos," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11401 LNCS, pp. 547–555, 2019.

[16] J. Mallmann, A. O. Santin, E. K. Viegas, R. R. dos Santos, and J. Geremias, "Ppcensor: Architecture for real-time pornography detection in video streaming," *Future Generation Computer Systems*, vol. 112, pp. 945–955, 2020.

[17] A. Gangwar, V. González-Castro, E. Alegre, and E. Fidalgo, "Attm-cnn: Attention and metric learning based cnn for pornography, age and child sexual abuse (csa) detection in images," *Neurocomputing*, vol. 445, pp. 81–104, 2021.

[18] N. Gautam and D. K. Vishwakarma, "Obscenity detection in videos through a sequential convnet pipeline classifier," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 8920, no. c, pp. 1–10, 2022.

[19] S. Samal, R. Nayak, S. Jena, and B. K. Balabantaray, "Obscene Image Detection Using Transfer Learning and Feature Fusion," *Multimedia Tools and Applications*, vol. 82, no. 19, pp. 28 739–28 767, 2023.

[20] A. Gangwar, V. González-Castro, E. Alegre, E. Fidalgo, and A. Martínez-Mendoza, "Deephsar: Semi-supervised fine-grained learning for multi-label human sexual activity recognition," *Information Processing and Management*, vol. 61, 9 2024.

[21] F. C. Akyon and A. Temizel, "State-of-the-art in nudity classification: A comparative analysis," in *2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*. IEEE, 2023, pp. 1–5.

[22] D.-D. Phan, H.-L. Tran, T.-T. Nguyen, and D.-L. Vu, "Zero-shot pornographic detection using vision-language models," in *2024 13th International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 2024, pp. 1–6.

[23] H. Lovenia, D. P. Lestari, and R. Frieske, "What did i just hear? detecting pornographic sounds in adult videos using neural networks," in *Proceedings of the 17th International Audio Mostly Conference*. Association for Computing Machinery, 9 2022, pp. 92–95.

[24] L. Zhou, K. Wei, Y. Li, Y. Hao, W. Yang, and H. Zhu, "Acoustic pornography recognition using convolutional neural networks and bag of refinements," in *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2022, pp. 840–845.

[25] S. Liu, R. Li, Q. Li, and J. Zhao, "Porn streamer audio recognition based on deep learning and random forest," *Applied Intelligence*, vol. 53, no. 15, p. 18857–18867, feb 2023. [Online]. Available: https://doi.org/10.1007/s10489-023-04491-x

[26] K. H. Song and Y. S. Kim, "Pornographic video detection scheme using multimodal features," pp. 1174–1182, 2018.

[27] L. Wang, J. Zhang, Q. Tian, C. Li, and L. Zhuo, "Porn streamer recognition in live video streaming via attention-gated multimodal deep features," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 12, pp. 4876–4886, 2020.

[28] J. Zhao, M. Shao, H. Peng, H. Wang, B. Li, and X. Liu, "Porn2vec: A robust framework for detecting pornographic websites based on contrastive learning," *Knowledge-Based Systems*, vol. 228, p. 107296, 9 2021.

[29] B. Mocanu, R. Tapu, and T. Zaharia, "Multimodal emotion recognition using cross modal audio-video fusion with attention and deep metric learning," *Image and Vision Computing*, vol. 133, p. 104676, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0262885623000501

[30] M. Khan, W. Gueaieb, A. El Saddik, and S. Kwon, "Mser: Multimodal speech emotion recognition using cross-attention with deep fusion," *Expert Systems with Applications*, vol. 245, p. 122946, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423034486

[31] M. Brousmiche, J. Rouat, and S. Dupont, "Multimodal attentive fusion network for audio-visual event recognition," *Information Fusion*, vol. 85, pp. 52–59, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253522000264

[32] D. D. Phan, T. T. Nguyen, K. N. K. Nguyen, Q. H. Nguyen, D. L. Vu, and H. L. Tran, "Lspd: A large-scale pornographic dataset for detection and classification," *International Journal of Intelligent Engineering and Systems*, vol. 15, pp. 198–213, 2022.

Microagregación con Privacidad Diferencial: Ruido Adaptativo a Nivel de Clúster

Jordi Soria-Comas
y David Megías
Universitat Oberta de Catalunya,
Est. d'Inf, Mult i Telecom,
UOC-TECH Research Centre,
CYBERCAT,
Rambla del Poblenou, 154
08018, Barcelona, Spain
{jordi_sc,dmegias}@uoc.edu

Jordi Castellà-Roca
y Carles Anglés-Tafalla
Universitat Rovira i Virgili,
Dept. d'Eng Inf i Mat,
CYBERCAT,
Av. Països Catalans 26,
E-43007 Tarragona, Spain
{jordi.castella, carles.angles}@urv.cat

Guillermo Navarro-Arribas
y Julián Salas
Universitat Autonoma de Barcelona,
Dept. d'Eng. de la Inf i les Com,
CYBERCAT,
08193 Bellaterra, Catalonia, Spain
{guillermo.navarro, julian.salas}@uab.cat

***Resumen*—La microagregación es una técnica establecida para la anonimización de datos que reemplaza los registros originales por centroides de clústeres, preservando la utilidad estadística, pero sin ofrecer garantías teóricas de privacidad. Para mejorar las garantías, se puede combinar con Privacidad Diferencial (DP), añadiendo ruido a los centroides. Sin embargo, los enfoques actuales aplican una cantidad uniforme de ruido a todos los centroides. Esto tiene un impacto en la utilidad de los datos en regiones densas del dominio, donde los clústeres son inherentemente más compactos y menos susceptibles a la reidentificación. En este trabajo, proponemos una nueva metodología que integra la microagregación con la Privacidad Diferencial Individual (iDP), con el objetivo de optimizar el equilibrio entre privacidad y utilidad. El núcleo de nuestra propuesta reside en ajustar dinámicamente la cantidad de ruido introducido basándose en la sensibilidad local de cada clúster individual.**

***Palabras clave*—Microagregación, privacidad diferencial, utilidad**

I. Introducción

La microagregación es una técnica de anonimización de datos utilizada para proteger la privacidad de los individuos, a la vez que se mantiene la utilidad de los datos para el análisis. Consiste en agrupar puntos de datos individuales en pequeños clústeres y luego reemplazar los datos de cada individuo dentro de ese grupo por un valor central (conocido como centroide) que es representativo de todo el clúster. Es habitual calcular el centroide como la media de los valores del clúster. El objetivo es suprimir la información sensible que podría identificar a los individuos, a la vez mantener la utilidad estadística de los datos.

Aunque la microagregación se utiliza ampliamente para anonimizar datos, no proporciona garantías de privacidad teóricas como da la Privacidad Diferencial (DP). En ciertas situaciones, los datos microaggregados pueden revelar información personal:

- Revelaciones por tamaño de grupo insuficiente o falta de diversidad estadística. Cuando los grupos son demasiado pequeños o tienen poca variabilidad interna (por ejemplo, todos los individuos tienen valores muy similares para los atributos agregados), la media u otros valores agregados no introducen suficiente ambigüedad o ruido para proteger la privacidad individual.
- Revelaciones por conocimiento externo. Si un atacante tiene acceso a conjuntos de datos externos con información sobre los individuos (como características demográficas o registros públicos), podría vincular esos datos con centroides concretos. Al hacerlo, el atacante reduce efectivamente el tamaño del grupo.
- Revelaciones debidas a valores atípicos (outliers). Si hay valores atípicos (por ejemplo, un individuo con un valor significativamente diferente a los demás individuos del clúster, como un ingreso muy alto en un grupo), el centroide se verá fuertemente influenciado por ese valor, comprometiendo la anonimidad.

Una forma efectiva de abordar los problemas de privacidad inherentes a la microagregación es combinarla con la privacidad diferencial [1]. El objetivo es mejorar la protección de la privacidad sin que esto comporte una gran perdida en la utilidad de los datos. En términos de privacidad, al añadir ruido DP a los valores agregados después de la microagregación, reducimos significativamente el riesgo de revelar los datos de cualquier individuo. Incluso si un atacante solo desconoce un registro en un grupo, el ruido añadido por DP asegura que el valor del registro restante no pueda inferirse del valor del centroide. En términos de utilidad, dado que la microagregación ya ofusca a los individuos al transformar los datos individuales en un valor representativo del grupo, la cantidad de ruido necesaria para la privacidad diferencial es menor que si añadiéramos ruido DP a nivel de datos individuales.

La combinación de microagregación y DP ha sido analizada en varios trabajos [6], [7], [4]. Sin embargo, estos enfoques se enfrentan a un desafío significativo que impacta negativamente en la utilidad de los datos: aplican la misma cantidad de ruido a todos los centroides, independientemente de las características del clúster. La razón es que la protección debe garantizarse con independencia del conjunto de datos.

Para minimizar el impacto del ruido, sería conveniente que el ruido se ajustara en función de la extensión del clúster, de modo que las regiones con mayor densidad de datos (valores más comunes) reciban menos ruido, mientras que las regiones más dispersas (valores poco frecuentes), que son más propensas a la reidentificación, reciban más ruido. Esto ayudaría a equilibrar la protección de la privacidad con una

mejor utilidad de los datos.

Para abordar este problema, recurrimos a la Privacidad Diferencial Individual (iDP), que, al igual que la DP, tiene como objetivo hacer que la contribución de cualquier individuo sea indistinguible (la probabilidad de obtener la misma respuesta a una consulta debe ser similar independientemente de la presencia o ausencia de ese individuo en el conjunto de datos). Sin embargo, a diferencia de la DP, la iDP solo busca ofrecer protección para el conjunto de datos real (no sobre cualquier conjunto de datos posible). Para ello basta con exigir indistinguibilidad entre el conjunto de datos real y aquellos que difieren de él en un único registro. La idea es que a un individuo no le importa la protección que obtendría si se consultara un conjunto de datos arbitrario, solo le importa la protección sobre el conjunto de datos real.

El resto del artículo se organiza de la siguiente manera. La Sección II recuerda los conceptos centrales de Privacidad Diferencial (DP) y microagregación involucrados en nuestra propuesta. La Sección III describe el uso de la Privacidad Diferencial Individual (iDP) para enmascarar los centroides obtenidos mediante microagregación. La Sección IV presenta los resultados empíricos en una variedad de escenarios de datos para demostrar las mejoras de nuestra propuesta e identificar los contextos en los que se obtiene el mayor beneficio. La Sección final ofrece las conclusiones y enumera las líneas de investigación futura.

II. Marco Teórico

II-A. Microagregación Univariada

La microagregación es una técnica utilizada en el contexto de la privacidad de datos para proteger la información sensible a la vez que se preservan sus propiedades estadísticas. Consta de dos pasos principales:

- Agrupación (*Clustering*) de los datos: los datos se dividen en grupos o clústeres no superpuestos, cada uno de los cuales contiene k o más registros.
- Agregación de valores dentro de cada grupo: cada clúster de registros se reemplaza por un valor representativo, el centroide (por ejemplo, la media).

Para minimizar la pérdida de información en la microagregación, los clústeres deben ser lo más homogéneos posible, lo que significa que los valores dentro de cada grupo deben ser lo más similares entre sí.

En este artículo, nos centramos en la microagregación univariada de datos numéricos. Los siguientes pasos resumen el proceso de microagregación en dichos conjuntos de datos:

1. **Entrada:**
   - Un conjunto de datos numéricos $D = \{x_1, x_2, \ldots, x_n\}$
   - Un tamaño de grupo especificado k, donde k es el número de puntos de datos en cada grupo.
2. **Clasificación de los datos:**
   - Se ordena el conjunto de datos D en orden no decreciente de los valores.
   - El conjunto de datos ordenado pasa a ser $D_{\text{sorted}} = \text{sort}(D)$.
3. **División de los datos en grupos:**
   - Se divide el conjunto de datos ordenado D_{sorted} en grupos no superpuestos de tamaño k.
   - Si el tamaño del conjunto de datos n no es divisible por k, los registros restantes se distribuyen en los clústeres.
4. **Cálculo del valor central para cada grupo:**
   - Para cada grupo, se calcula la media de los valores dentro del grupo:
   $$\text{media}_i = \frac{\sum_{j=1}^{k} \text{grupo}_i[j]}{k}.$$
5. **Sustitución de los valores de cada grupo por el valor central:**
   - Todos los valores dentro de cada grupo se reemplazan por el valor central calculado.

II-B. Privacidad Diferencial

La DP asume la presencia de un tercero de confianza que: (i) posee el conjunto de datos, (ii) recibe las consultas enviadas por los usuarios de los datos, y (iii) responde a ellas de una manera que garantiza la privacidad. La noción estándar de DP se formaliza en términos de indistinguibilidad de la respuesta a las consultas entre conjuntos de datos que difieren en un registro: la probabilidad de obtener una misma respuesta debe ser similar en ambos casos.

Definición 1 (ε-Privacidad Diferencial)**.** Una función aleatorizada κ proporciona ε-Privacidad Diferencial (o ε-DP) si, para todos los conjuntos de datos D_1 y D_2 que difieren en un registro (también conocidos como conjuntos de datos vecinos), y para todo $S \subset \text{Rango}(\kappa)$, se tiene

$$\Pr(\kappa(D_1) \in S) \leq \exp(\varepsilon) \cdot \Pr(\kappa(D_2) \in S).$$

Dada una consulta f, el objetivo de la DP es encontrar una κ_f que satisfaga los requisitos de privacidad establecidos en la Definición 1 y se aproxime a f lo más posible. El valor de $\kappa_f(D)$ se devuelve entonces como el resultado de la consulta, en lugar del valor real $f(D)$. El valor de ε nos indica el nivel de privacidad perdida, cuanto más grande sea menos garantías de privacidad.

Una propiedad interesante de la privacidad diferencial es que podemos hacer un seguimiento de la privacidad perdida debido a la realización de varias consultas; esto se conoce como composición. En este trabajo solo tratamos con composición secuencial [2].

Teorema 2 (Composición secuencial)**.** *Sea κ_1 una función aleatorizada que proporciona ε_1-DP y κ_2 una función aleatorizada que proporciona ε_2-DP. Entonces, cualquier función determinista de (κ_1, κ_2) proporciona $(\varepsilon_1 + \varepsilon_2)$-DP.*

Una forma común de lograr DP para consultas numéricas es añadir un ruido cuya variabilidad se ajusta al cambio máximo entre conjuntos de datos vecinos (también conocida como sensibilidad global).

Definición 3 (Sensibilidad global)**.** La sensibilidad global de una función $f : \mathcal{D} \to \mathbb{R}^k$ es

$$\Delta f = \max_{D_1, D_2 \text{ vecinos}} \|f(D_1) - f(D_2)\|_1 .$$

Un ruido distribuido por Laplace es una alternativa común para lograr DP.

Proposición 4. *Sea f una función de consulta con valores en $\mathbb{R}^k$. El mecanismo $\kappa_f(D) = f(D) + (N_1, \ldots, N_k)$, donde N_i son ruidos aleatorios independientes e idénticamente distribuidos extraídos de una distribución* $\mathrm{Laplace}(0, \varepsilon/\Delta f)$*, proporciona ε-DP.*

II-C. Microagregación con DP

Sea f la función que evaluada en un conjunto de datos, retorna los clústeres de la microagregación. Para obtener DP, tenemos que enmascararlos añadiendo ruido.

En el caso de microaggregación univariada, el cambio más grande en los centroides se da cuando un registro que estava en un extremo del dominio pasa al otro extremo [4]. En este caso, si ΔD es la dimensión del dominio, la suma de los cambios en los centroides es $\Delta D/k$, donde el cociente proviene del hecho que al hacer la media el cambio en cada centroide queda dividido por k. Por tanto, la función $\kappa_f(D) = (c_1(D), \ldots, c_{n/k}(D)) + (L_1, \ldots, L_{n/k})$, donde $L_i \sim L(0, {}^{\Delta D}/_{k\varepsilon})$ independientes, da ε-DP.

II-D. Privacidad Diferencial Individual

Antes de formalizar la DP, en el artículo [1] se propone la siguiente intuición:

> *Cualquier revelación dada será, dentro de un pequeño factor multiplicativo, igual de probable participe o no el individuo en la base de datos. Como consecuencia, hay un riesgo marginalmente mayor para el individuo que participa y solo una ganancia marginal para el individuo que oculta o tergiversa sus datos.*

Sin embargo, la formalización de la DP (véase la Definición 1) es más rigurosa de lo que requiere la intuición anterior. De acuerdo con la intuición previa, es suficiente decirle al individuo que las respuestas a las consultas basadas en el conjunto de datos no permitirán a nadie aprender algo que no fuera ya aprendible sin su presencia. En otras palabras, si D es el conjunto de datos recopilado, las garantías de privacidad objetivo pueden lograrse simplemente exigiendo la indistinguibilidad de las respuestas entre D y sus conjuntos de datos vecinos. No es necesario pedir la indistinguibilidad entre cualquier par de conjuntos de datos vecinos.

Definición 5 (ε-Privacidad Diferencial Individual [5])**.** Dado un conjunto de datos D, un mecanismo de respuesta $\kappa(\cdot)$ satisface ε-Privacidad Diferencial Individual (o ε-iDP) si, para cualquier conjunto de datos D' que sea vecino de D, y cualquier $S \subset \mathrm{Rango}(\kappa)$ se tiene

$$\exp(-\varepsilon) \cdot \Pr(\kappa(D') \in S) \leq \Pr(\kappa(D) \in S)$$
$$\Pr(\kappa(D) \in S) \leq \exp(\varepsilon) \cdot \Pr(\kappa(D') \in S).$$

Como el papel de los conjuntos de datos D y D' no es intercambiable, necesitamos solicitar ambas desigualdades.

Los teoremas de composición para la DP descritos en la Sección II-B también son trivialmente satisfechos por la privacidad diferencial individual. El mecanismo de adición de ruido Laplace descrito en la Sección II-B también conduce a ε-iDP. Sin embargo, podemos ajustar el ruido a la sensibilidad local, en lugar de a la sensibilidad global.

Definición 6 (Sensibilidad local)**.** Sea f una función que se evalúa en conjuntos de datos y devuelve valores en $\mathbb{R}^k$. La sensibilidad local de f en D es

$$\mathrm{LS}_f(D) = \max_{D', D \text{ vecinos}} \|f(D') - f(D)\|_1 .$$

Proposición 7. *Sea f una función de consulta que toma valores en $\mathbb{R}^k$. El mecanismo $\kappa(x) = f(x) + (N_1, \ldots, N_k)$, donde N_i son ruidos aleatorios independientes e idénticamente distribuidos* $\mathrm{Laplace}(0, LS_f(D)/\varepsilon)$*, proporciona ε-iDP.*

III. Microagregación con iDP

Al microagregar un conjunto de datos en clústeres de tamaño k, la densidad de los registros en cada región del dominio de datos tiene un impacto significativo en la medida de los clústeres. Los clústeres formados en áreas más densas del dominio contienen registros que están más próximos entre sí y reemplazarlos por el centroide provoca una pérdida de información menor. Por el contrario, los clústeres formados en áreas más dispersas contienen registros que están distantes entre sí y reemplazarlos por el centroide conduce a una mayor pérdida de información. Esto es coherente con los enfoques tradicionales de privacidad de datos, donde la cantidad de enmascaramiento que se realiza a los registros depende de cuán único sea un registro.

Cuando aplicamos DP sobre los centroides, como explicamos en la Sección II-C, cada uno de ellos se enmascara con la misma cantidad de ruido: un ruido laplaciano con escala $\Delta D/k\varepsilon$. Así, el impacto es más notable en los clústeres en regiones más densas. En las regiones dispersas, el centroide era una aproximación burda y, por lo tanto, proporcionalmente el nuevo ruido es menor.

El objetivo es ajustar la cantidad de ruido añadido a la densidad de la región en la que se encuentra cada clúster. Al hacerlo, podremos ofrecer resultados más precisos en regiones más densas (que probablemente representan un mayor porcentaje del total de registros) a expensas de introducir más ruido en regiones más dispersas. En DP, la sensibilidad global de cada centroide es $\Delta D/k$, ya que necesitamos considerar cualquier par posible de conjuntos de datos vecinos, incluso aquellos en los que uno de los registros del clúster cambia de un extremo del dominio de datos al otro. Por el contrario, si empleamos iDP como noción de privacidad, es posible calcular una sensibilidad mucho más precisa para cada uno de los centroides.

Antes de calcular la sensibilidad de cada centroide, fijemos algunas notaciones. Sea $D = \{x_1, x_2, \ldots, x_n\}$ el conjunto de datos real. Sin pérdida de generalidad, asumimos que $x_1 \leq x_2 \leq \ldots \leq x_n$. Sea $D' = \{x'_1, x'_2, \ldots, x'_n\}$ un conjunto de datos vecino. Al igual que antes, asumimos que los registros en D' están ordenados de forma ascendente. Sea k el tamaño del clúster y, por simplicidad, asumiremos que n es múltiplo de k. Es decir, que podemos generar n/k clústeres de tamaño k sin que quede ningún registro.

Como se describe en la Sección II-A, los clústeres de microagregación en D y D' contienen cada uno k registros contiguos. Sea $C_D^i = \{x_{ik+1}, \ldots, x_{(i+1)k}\}$ el i-ésimo clúster sobre el conjunto de datos D. De manera similar, $C_{D'}^i$ es el i-ésimo clúster sobre el conjunto de datos D'. Como D y D' difieren como máximo en un registro, los clústeres C_D^i y $C_{D'}^i$ también difieren como máximo en un registro. En particular,

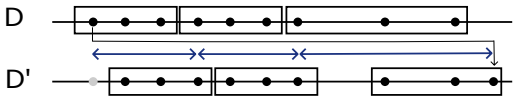


Figura 1. Ilustración de cómo el cambio de un único registro afecta a los clústeres en conjuntos de datos ordenados D y D'

si los conjuntos de datos D y D' difieren en los valores de los registros x_r y x'_s, entonces los clústeres que contienen los registros entre x_r y x'_s difieren en un registro, mientras que todos los demás clústeres permanecen sin modificar (véase la Figura 1).

Así, la sensibilidad local de un centroide es el máximo cambio en el centroide debido al cambio de uno de los registros del clúster. El cambio máximo ocurre cuando:

- Se elimina el registro más pequeño del clúster y se añade el registro siguiente al clúster.
- Se elimina el registro más grande del clúster y se añade el registro anterior al clúster.

Cuando se trata del primer y el último clúster, el nuevo registro puede ir hasta el límite inferior o superior del dominio de datos (si el dominio está acotado). Para tener esto en cuenta, añadimos dos registros adicionales:

- x_0, que se establece en el límite inferior del dominio de datos.
- x_{n+1}, que se establece en el límite superior del dominio de datos.

Estos registros no participan en el cálculo del centroide (ya que no son registros propiamente dichos), pero permiten evitar tener que lidiar con diferentes casos al calcular la sensibilidad local.

Por lo tanto, la sensibilidad del centroide c_i es

$$\Delta c_i = \text{máx} \left\{ \frac{x_{(i+1)k+1} - x_{ik+1}}{k}, \frac{x_{(i+1)k} - x_{ik}}{k} \right\}.$$

Por composición secuencial, si consultamos cada centroide y obtenemos una respuesta ε_i-iDP, el conjunto de centroides es $\sum \varepsilon_i$-iDP. Para obtener la misma protección para cada centroide, hacemos que todos los ε_i sean iguales. Tomamos $\varepsilon_i = \varepsilon/(n/k)$. Por lo tanto, enmascaramos cada centroide añadiendo un ruido Laplace:

$$c_i + \text{Laplace}\left(0, \Delta c_i \times (n/k)/\varepsilon\right).$$

Como el ruido que se introduce es proporcional a la longitud del clúster, estamos añadiendo, efectivamente, menos ruido a los clústeres en áreas densamente pobladas, mientras que aumentamos el ruido añadido a las áreas escasamente pobladas. Para ilustrar esto, analicemos el caso de datos distribuidos uniformemente en todo el dominio de datos. En tal caso, es probable que cada clúster abarque la misma proporción del dominio de datos y se espera que la sensibilidad del centroide sea:

$$\Delta c_i = {}^{\Delta D/(n/k)}/_{k}.$$

Para esta sensibilidad, el ruido que tenemos que añadir es $\text{Laplace}(0, \Delta D/(k\varepsilon))$, el mismo ruido que añadimos al usar DP simple. A partir de esta base, es fácil ver que si la distribución no es uniforme, el ruido añadido en áreas densamente pobladas se vuelve menor (ya que el Δc_i en esas áreas se reduce), y el ruido en áreas escasamente pobladas se vuelve mayor (ya que el Δc_i en esas áreas aumenta).

IV. Resultados experimentales

Esta sección presenta la evaluación empírica sobre conjuntos de datos sintéticos. Los conjuntos de datos generados contienen 1.000 registros con un único atributo numérico, cuyo valor sigue una distribución $N(0,1)$ truncada en el intervalo $[-5,5]$.

Para evaluar la pérdida de información del conjunto de datos (es decir, la diferencia entre los datos originales y los enmascarados), utilizamos el Error Absoluto Medio (MAE) y el Error Cuadrático Medio (MSE), que son medidas estándar de pérdida de información en microagregación. Estas métricas contabilizan la distancia euclidiana promedio y la distancia euclidiana al cuadrado entre los registros originales y los enmascarados:

$$\text{MAE} = \frac{1}{|D|} \sum_{r \in D} \|r - r'\|_2,$$

$$\text{MSE} = \frac{1}{|D|} \sum_{r \in D} \|r - r'\|_2^2,$$

donde r son los registros del conjunto de datos, y r' son las versiones enmascaradas de cada r.

Si bien las medidas globales de pérdida de información nos dan una visión general, no son suficientes para evaluar nuestra propuesta, ya que esperamos que el ruido no esté distribuido de manera uniforme. Por esta razón, también evaluaremos el MAE y el MSE sobre cada uno de los clústeres.

Para la privacidad diferencial (DP), utilizamos $\varepsilon = 1$, que es un valor común. Para la microagregación, hemos utilizado tres tamaños de clústeres diferentes $k \in \{10, 50, 100\}$, lo que abarca un amplio rango de tamaños de clústeres.

Para dar contexto a nuestra propuesta, la comparamos con la microagregación simple (que muestra el límite inferior de la pérdida de información) y la microagregación combinada con DP, *microagregación+DP*. Para cada método y parametrización, se han realizado 10 rondas y se proporciona el promedio de todas ellas. Para hacer la comparación más consistente, todos los valores detallados se calcularon sobre los mismos 10 conjuntos de datos.

En cuanto a las medidas globales de MAE y MSE, no observamos diferencias significativas entre *microagregación+DP* y *microagregación+iDP*. En algunos casos, uno es ligeramente mejor que el otro, y viceversa. Por supuesto, ambos muestran una pérdida de información significativamente mayor que la microagregación simple, lo que proviene de las fuertes garantías de privacidad que incorporan. En la Tabla I se muestran los valores de MAE i MSE globales para diferentes niveles de mmicroagregación.

La información que obtenemos al analizar la pérdida de información a nivel de clúster confirma la redistribución del ruido entre clústers. En la Figura 2, observamos que, independientemente del tamaño del clúster, *microagregación+iDP* produce mejores resultados en una gran parte de los clústeres. Aproximadamente el 95 % de los clústeres presentan mejores resultados con *microagregación+iDP*.

Para analizar la distribución del ruido debido a iDP entre las regiones densas y dispersas, se representa el MSE de

Tabla I
MAE I MSE PARA LOS MÉTODOS EVALUADOS Y DIFERENTES NIVELES DE MICROAGREGACIÓN (K).

	$k = 10$		$k = 50$		$k = 100$	
	MAE	MSE	MAE	MSE	MAE	MSE
Microagregación	0,0174	0,00279	0,0688	0,0152	0,126	0,041
Microagregación + DP	1,0	1,9	0,225	0,096	0,179	0,0635
Microagregación + iDP	0,562	1,25	0,242	0,0423	0,167	0,0881

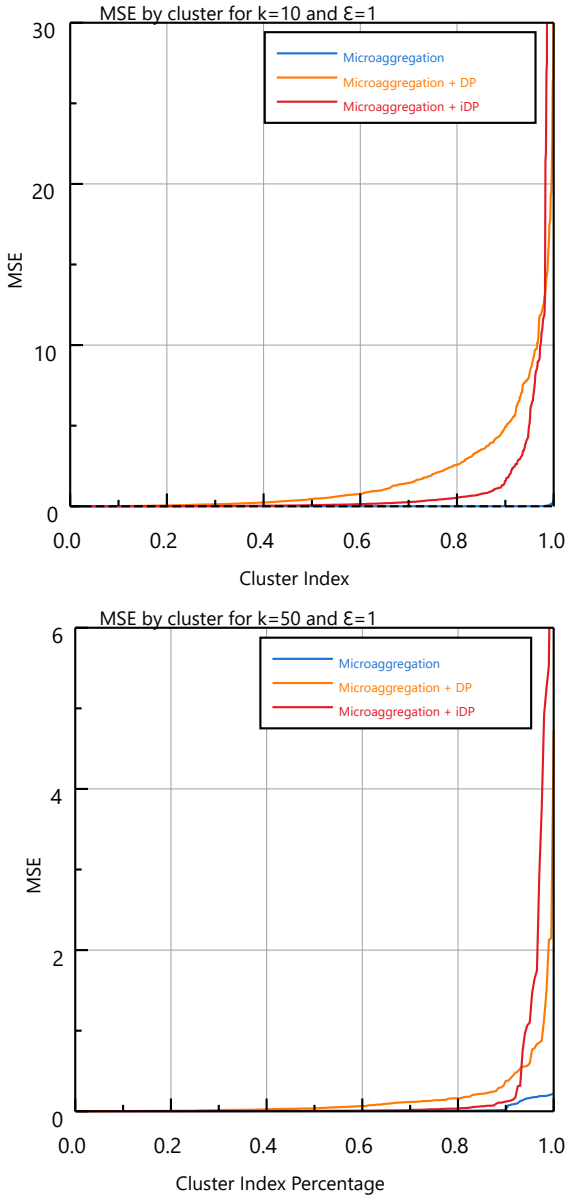


Figura 2. Distribución de MSE por clúster ordenada de menor a mayor MSE, para $k = 10$ y $k = 50$.

los clústeres sin reordenarlos. De esta manera, los valores al principio y al final del eje x corresponden a los clústeres en las regiones cercanas a –5 y 5 (que son dispersas), mientras que los valores en el centro del eje x corresponden a los clústeres cercanos a 0 (que son más densos). Hay que tener en cuenta que el eje x representa el índice del clúster y no la distribución real de los valores en el rango $[-5, 5]$ (que sigue una distribución $N(0, 1)$). En la Figura 3, se representa el $\log_{10}$ del MSE de los clústeres para los tres métodos analizados. Utilizamos escala logarítmica para separar los puntos que corresponden a diferentes métodos de forma más clara. Esto nos permite observar mejor las tendencias en el MSE. Los puntos correspondientes a la microagregación tienen un MSE más bajo y es más pequeño en el centro (región densa) que en los extremos (regiones dispersas). Por el contrario, la *microagregación+DP* muestra un MSE que es bastante uniforme en todo el dominio de los datos. La razón es que el ruido debido a DP es uniforme en todos los clústeres y domina sobre el ruido debido a la microagregación. Finalmente, los puntos relacionados con la *microagregación+iDP*, aunque muestran valores de MSE más grandes, podemos observar que siguen una tendencia similar a la de la microagregación (con valores más grandes en los extremos y más pequeños en el centro).

V. CONCLUSIONES

Si bien la microagregación es una técnica comúnmente utilizada para el control de divulgación estadística, carece de las garantías de privacidad que ofrece la DP. Por ejemplo, al vincular los registros de datos con clústeres particulares, los intrusos pueden reducir el nivel de protección. Para evitar esto, se puede aplicar DP a los centroides de los clústeres. Dado que cada centroide depende de un grupo de k o más registros, la sensibilidad se reduce por un factor k, lo que reduce la cantidad de ruido que es necesario agregar.

Sin embargo, en la DP estándar, cada centroide se enmascara con la misma cantidad de ruido. Agregar la misma cantidad de ruido a cada clúster tiene un impacto mayor en los clústeres de las regiones más densas, ya que son más pequeños. Para los clústeres en áreas más dispersas, como el centroide ya es una estimación aproximada, el ruido adicional es menos notorio. Este tratamiento uniforme de los clústeres tiene un profundo impacto en la pérdida de información.

Para resolver este problema, proponemos el uso de iDP, que nos permite adaptar la cantidad de ruido a cada clúster. A los clústeres en áreas más densas se les agrega menos ruido, mientras que a los clústeres en áreas más dispersas se les agrega más ruido. Esta adaptación nos permite reducir significativamente la pérdida de información en las áreas más densas, a costa de agregar significativamente más ruido en las áreas más dispersas. Sin embargo, dado que la precisión de los centroides en las áreas dispersas ya era baja, el impacto de agregar más ruido es menos notorio.

Hemos evaluado nuestra propuesta en un conjunto de datos sintético generado a partir de una distribución normal. Los resultados muestran que, aunque las medidas globales de pérdida de información tienen valores similares entre DP e iDP, al evaluar a nivel de clúster iDP preserva mejor la utilidad de los clústers en regiones densas.

En este trabajo, la microagregación se usa únicamente para reducir la cantidad de ruido introducido por la DP.

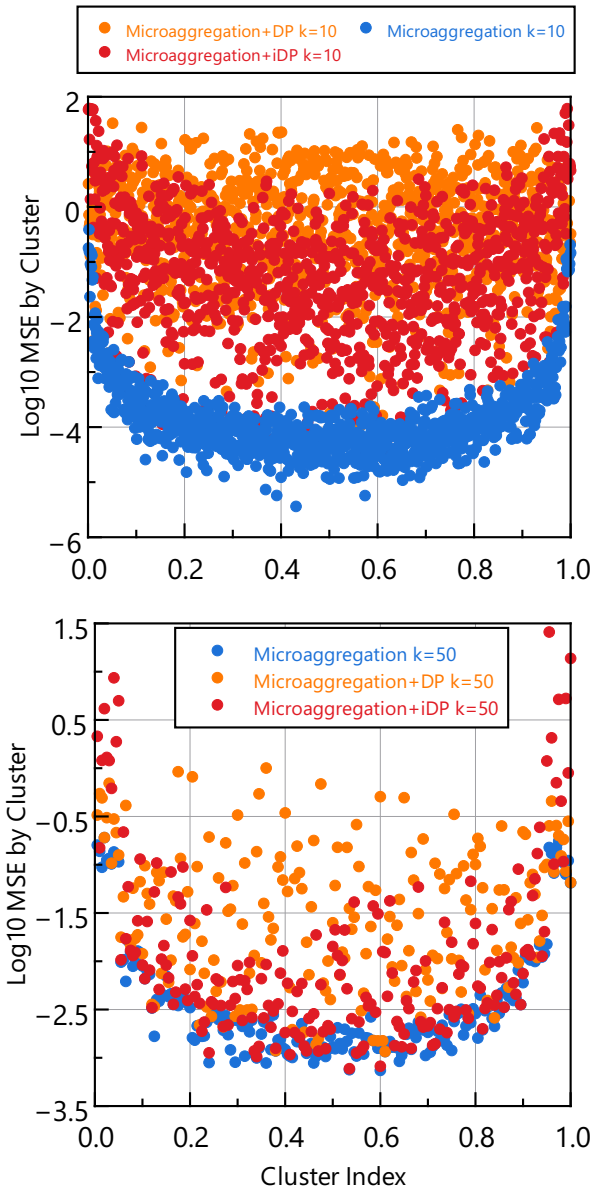


Figura 3. Log10 del MSE de los clústeres sin reordenar para $k = 10$ y $k = 50$.

Como trabajo futuro queremos determinar que medida tiene que tener cada clúster para que se minimice la perdida de información total que resulta de agregar la procedente del ruido de DP a la de la microagregación, teniendo en cuenta que el nivel óptimo de microagregación puede variar en función de la densidad del área.

Agradecimientos

Este trabajo ha sido apoyado por los proyectos estratégicos en ciberseguridad HERMES y DANGER (C062-23), y la cátedra en ciberseguridad ARTEMISA (C057/23) financiados por INCIBE a través de la Unión Europea – NextGenerationEU y el Plan de Recuperación, Transformación y Resiliencia; por los proyectos PROVTOPIA (PID2023-150098OB-I00), SECURING/CYBER (PID2021-125962OB-C31), SECURING/DATA (PID2021-125962OB-C32), SECURING/NET (PID2021-125962OB-C33), SAFE/DATA (PID2024-156914OB-C42), SAFE/BLOCKCHAIN (PID2024-156914OB-C43), SAFE/CYBER (PID2024-156914OB-C41) del Ministerio de Ciencia e Innovación, Agencia Estatal de Investigación, y *European Regional Development Fund* (ERDF). Los proyectos SGR2021-00643, 2021SGR 00115 de la Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) de Catalunya.

Referencias

[1] C. Dwork. Differential privacy. *Automata, languages and programming*, page 1–12, 2006.

[2] Frank D. McSherry. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data (SIGMOD '09)*. Association for Computing Machinery, New York, NY, USA, 19–30.

[3] J. Domingo-Ferrer and J. M. Mateo-Sanz, Practical data-oriented microaggregation for statistical disclosure control. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):189–201, Feb 2002.

[4] J. Soria-Comas, D. Sánchez, J. Domingo-Ferrer, S. Martínez, L. Del Vasto-Terrientes. Conciliating privacy and utility in data releases via individual differential privacy and microaggregation. *Transactions on Data Privacy*, 18:29–50, Jan 2025.

[5] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and D. Megias. Individual differential privacy: A utility-preserving formulation of differential privacy guarantees. *IEEE Transactions on Information Forensics and Security*, 12:1418–1429, Jun 2017.

[6] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez and S. Martínez. Enhancing data utility in differential privacy via microaggregation-based k-anonymity. *The VLDB Journal*, 23(5):771–794, Sep 2014.

[7] X. Ye, Y. Zhu, M. Zhang and H. Deng, Differential Privacy Data Release Scheme Using Microaggregation With Conditional Feature Selection. *IEEE Internet of Things Journal,* 10(20):18302–18314, Oct 2023,

Generación de Código Malicioso con IA Generativa y Modelos de Lenguaje

Luis Alberto Martínez Hernández[1], Sandra Pérez Arteaga[1], Ana Lucila Sandoval Orozco[1], Luis Javier García Villalba[1]

[1]Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {luisal11, sandrp08, asandov, javierv}@ucm.es

***Resumen*—La creciente dependencia para el análisis y generación de código, ha impulsado el uso de modelos de aprendizaje automático para la detección de vulnerabilidades. Aunque estos modelos han demostrado alcanzar un rendimiento notable, su robustez frente a entradas adversarias sigue siendo un aspecto poco explorado. Por lo anterior, resulta fundamental evaluar si los modelos generativos permiten la creación de código vulnerable capaz de evadir los detectores basados en IA. En este trabajo se aborda este vacío analizando la capacidad de los modelos generativos para producir código vulnerable en C/C++ diseñado específicamente para evadir los sistemas de detección de vulnerabilidades. Se estudian diversas arquitecturas secuenciales como LSTM, GRU, DoubleLSTM, DoubleGRU y modelos de lenguaje como GPT-2 y CodeBERT. Para evaluar la eficacia de los modelos generadores, se desarrollan clasificadores binarios basados en arquitecturas LSTM bidireccionales para discriminar si el código vulnerable generado es vulnerable o no. Los resultados muestran que los modelos basados en transformadores superan significativamente a las redes recurrentes. El trabajo menciona limitaciones importantes relacionadas con la detección de código vulnerable generado, subrayando la necesidad de mejorar la robustez de los detectores y abriendo líneas futuras de investigación en generación adversaria y tokenización avanzada.**

***Index Terms*—Ciberseguridad, Código vulnerable, Detección de vulnerabilidades, Inteligencia artificial generativa, Modelos generativos, Transformers**

I. Introducción

En la actualidad, la inteligencia artificial es uno de los campos emergentes y de mayor interés de investigación científica y tecnológica. La aplicación de técnicas de inteligencia artificial en el ámbito de la ciberseguridad ha permitido desarrollar sistemas avanzados de detección y prevención de amenazas mejorando la respuesta a incidentes de seguridad. Sin embargo, su fiabilidad se ve comprometida cuando se enfrentan a escenarios no previstos durante su entrenamiento. Si bien se han dedicado esfuerzos tanto académicos como industriales para perfeccionar la capacidad de detección de estos sistemas, la evaluación de la robustez frente a ataques adversarios, principalmente los generados por la inteligencia artificial siguen siendo un área poco explorada, además de que, si bien hay grandes avances en el ámbito científico, carecen de pruebas o aplicaciones en el entorno real. Este trabajo aborda este vacío analizando el potencial ofensivo de los modelos generativos modernos. Se plantea la factibilidad de emplear arquitecturas de aprendizaje automático, potencialmente accesibles para atacantes malintencionados, con el fin de generar código vulnerable estructurado de manera que pueda evadir sistemas de detección de vulnerabilidades basados en inteligencia artificial. No solo se trata de generar código, sino de evaluar la capacidad de estos modelos para producir *"Falsos Negativos"* que pasen desapercibidos para los sistemas de detección. Se ha diseñado un entorno experimental adversario que contrasta el rendimiento de dos paradigmas de generación: las redes neuronales recurrentes, implementando variantes de LSTM y GRU, frente a los Modelos de Lenguaje Grande (LLM) basados en la arquitectura Transformer, específicamente GPT-2 y CodeBERT. El presente estudio no solo mide la coherencia sintáctica del código resultante, sino que cuantifica la "tasa de engaño" de cada arquitectura. A diferencia de investigaciones previas que se limitan a la detección estática, este enfoque pone de manifiesto las limitaciones de los detectores actuales ante la complejidad semántica que aportan los modelos basados en atención y paralelismo. Los hallazgos presentados a continuación sugieren que la sofisticación de los LLM representa un desafío inmediato para los sistemas de seguridad vigentes, obligando a replantear las estrategias de defensa ante la inminente proliferación de malware y exploits generados de forma sintética. El resto del trabajo se organiza como sigue: En la Sección II se realiza una síntesis de los principales trabajos relacionados con el tema de investigación. En la Sección III se describe la Metodología seguida para el desarrollo de la propuesta. En la Sección IV se realizan los experimentos realizados y se presentan los resultados obtenidos. Finalmente, las conclusiones del trabajo se incluyen en la Sección V.

II. Trabajos relacionados

Diversos estudios han explorado el uso de modelos de aprendizaje automático para identificar patrones asociados a vulnerabilidades conocidas, mejorando la capacidad de respuesta ante amenazas emergentes [1]. Sin embargo, un área poco explorada es la generación de código vulnerable mediante modelos generativos y NLP.

En la actualidad ha habido avances significativos en las técnicas de generación automática de código además de abordar aplicaciones y desafíos. Se han desarrollado métodos meta-aprendidos como Metacoder [2], capaces de transferir conocimiento desde lenguajes generales de dominio específico

mediante MAML y clustering no supervisado. Paralelamente, en [3] se centran en la calidad del código producido por LLMs mostrando que, aunque ChatGPT-4o genera soluciones funcionales, persisten carencias en documentación, manejo de excepciones y robustez, así como las diferencias de rendimiento entre lenguajes. Otros trabajos comparativos revelan que la eficacia general de modelos generativos varía ampliamente y que herramientas como Codex y GPT-4.0 destacan en tasas de éxito y evaluaciones de sostenibilidad [4]. Para abordar la complejidad de proyectos extensos de Deep Learning, se proponen enfoques guiados por planificación como DLCodeGen, que integra predicción estructurada, RAG dual y aprendizaje comparativo, superando a modelos generalistas en exactitud y evaluación humana [5].

En dominios complejos como la simulación científica, en [6] se observa los LLMs requieren supervisión humana y tienen limitaciones por knowledge cutoff, inconsistencias y falta de contexto. Desde la perspectiva ética y forense, se ha demostrado la necesidad de marcos de auditoría como CODEFORENSIC [7] para evaluar memorización, detección y atribución de código neuronal, revelando que la atribución unitaria es teóricamente imposible y debe reemplazarse por enfoques multi-fragmento o clasificatorios. Otros enfoques como el de [8] se orientan a la traducción automática de artefactos de diseño a código mediante visión por computadora, CNNs y MDA, mostrando la capacidad de generar código alineado con modelos UML y DSL especializados. Finalmente, investigaciones aplicadas al desarrollo web y a la seguridad del código generado por IA destacan tanto la utilidad de sistemas basados en la API de OpenAI para acelerar el prototipado [9] como la persistencia de vulnerabilidades severas como inyecciones y prácticas criptográficas inseguras dando su importancia al código generado por humanos como referencia de seguridad, exigiendo estrategias de mitigación como fine-tuning seguro y RLHF [10].

III. Metodología

Para evaluar la viabilidad de la generación de código vulnerable capaz de evadir los sistemas de detección de vulnerabilidades basados en IA, se diseñó un entorno experimental compuesto por tres modelos principales: el preprocesamiento de datos, un modelo discriminador que actúa como defensa y un conjunto de modelos generativos que actúan como atacantes variando la complejidad y el enfoque.

III-A. Selección y Preprocesamiento del dataset

El desarrollo de modelos de aprendizaje automático requiere una gran cantidad de datos de calidad para entrenar los modelos de manera óptima. Para este trabajo se emplearon datasets que contienen fragmentos de código en C/C++ etiquetados como vulnerables o no vulnerables. Los generadores del modelo deberán capturar la distribución de los datos vulnerables para generar código que pueda evadir la detección, y el discriminador deberá aprender a distinguir entre código vulnerable y no vulnerable. Entre los datasets existentes, resumidos en la Tabla I, destacan BigVul y Draper VDISC por ofrecer muestras a nivel de función, a diferencia de otros conjuntos que agrupan scripts, proyectos o archivos completos. Para esta primera aproximación, seleccionamos BigVul, que contiene 188.637 muestras de código, de las cuales 10.900 corresponden a funciones vulnerables y 177.737 a funciones no vulnerables, principalmente en C/C++. Cabe destacar que, aunque en la Tabla se indica el número de proyectos de origen, el total de muestras de BigVul corresponde a funciones individuales extraídas de dichos proyectos de código abierto alojados en GitHub recopiladas manualmente a partir de repositorios públicos, incorporando información adicional como la fecha de publicación, el mensaje de commit y el identificador CVE asociado. Este dataset ofrece un equilibrio adecuado entre tamaño y calidad de los datos, permitiendo un entrenamiento inicial robusto y dejando abierta la posibilidad de extender el estudio a Draper VDISC en fases posteriores.

Tabla I
Características Sobre los Datasets Existentes

Dataset	Granularidad	Casos	# Vulns
Big-Vul	Funciones	348 proyectos	3.754
Taxonomy	Scripts	1.164 casos	873
IntroClass	Scripts	6 tareas	998
ManyBugs	Proyectos	5,9 M líneas	185
SVCP4C	Archivos	2.378 archivos	9.983
Juliet 1.3	Scripts	64.099 casos	64.099
Wild C/C++	Archivos	12,1 M archivos	–
Draper VDISC	Funciones	1,27 M funciones	87.804

Draper VDISC consta de 1,27 millones de funciones, de las cuales 87.804 son vulnerables, identificadas mediante herramientas de análisis estático como Clang, CppCheck y Flawfinder, y clasificadas según cinco categorías CWE.

Una vez elegido el dataset, se procede a su preprocesamiento. Para determinar las muestras para entrenar de forma independiente al generador y al discriminador, se realizó un filtrado del dataset utilizando dos criterios principales: que la muestra fuera de código CC++ y que no superara las 500 palabras. Se obtuvieron 181.939 muestras, de las cuales 9.620 eran vulnerables.

A partir de estas muestras, se realizó una división independiente del conjunto de entrenamiento. Para ello, se utilizó como referencia la columna *func_before* del archivo original, que contiene las funciones vulnerables. Esta división permitió construir de forma separada los conjuntos de datos destinados a los modelos generadores y al modelo discriminador. Para el caso del discriminador, el conjunto se completó con un número equivalente de funciones no vulnerables, con el fin de preservar el balance entre clases. Los modelos generadores fueron entrenados con un total de 4.810 muestras. Para el almacenamiento de los datos, se emplearon dos formatos: texto plano, al que se aplicó un proceso de estandarización, y formato CSV, que se mantuvo sin modificaciones. Los archivos resultantes son:

- generator_dataset_raw.txt: funciones del generador sin estandarizar.
- generator_dataset_standardized.txt: funciones del generador estandarizadas, una por línea.
- generator_dataset.csv: muestras originales con sus atributos.
- discriminator_dataset.csv: funciones vulnerables y no vulnerables mezcladas aleatoriamente.

La estandarización aplicada garantiza que no se altere la sintaxis del código y se basa en las siguientes reglas: separar operadores y caracteres en tokens individuales, eliminar

comentarios, suprimir saltos de línea y tabulaciones, y evitar múltiples espacios consecutivos. Con ello obtenemos muestras uniformes, listas para su tokenización y procesamiento por los modelos. Ambos modelos se entrenan con el archivo *generator_dataset_standardized.txt* el cual contiene las muestras de código vulnerable escritas en cada línea.

A continuación se realiza la tokenización de las muestras utilizando la capa StringLookup de TensorFlow, que asigna a cada carácter único del conjunto de entrenamiento un índice entero. Este proceso permite representar las secuencias de texto en un formato adecuado para el modelo. Posteriormente, se lleva a cabo la vectorización. En esta etapa se generan las muestras y sus etiquetas como secuencias de índices numéricos. Antes de vectorizar, es necesario resolver el esquema de etiquetado.

El problema corresponde a generación de secuencias: dada una secuencia de entrada arbitraria, el modelo debe predecir su continuación. Para el aprendizaje contextual, se emplea un desplazamiento simple. Para una secuencia de longitud sequence_length, la muestra se define como los primeros sequence_length $-$ 1 tokens, y la etiqueta como esos mismos tokens desplazados una posición hacia la derecha respecto a la secuencia original.

Por ejemplo, con sequence_length $=$ 256, la entrada corresponde a los índices $[0, 254]$ y la etiqueta a $[1, 255]$. Este desplazamiento preserva el contexto token a token y favorece la generalización. Una mayor longitud de secuencia implica capturar dependencias más complejas y, por tanto, requiere modelos con mayor capacidad.

III-B. Creación y Entrenamiento de los modelos

Se evaluaron dos enfoques principales para la generación automática de código: modelos de generación en bucle basados en redes recurrentes y modelos LLM basados en arquitecturas de transformadores.

Modelos de generación en bucle

Los modelos de generación en bucle se basan en la producción iterativa de secuencias mediante retroalimentación. A partir de una secuencia de entrada, el modelo predice el siguiente token, lo incorpora a la cadena y vuelve a procesar la nueva secuencia para generar la siguiente predicción. Este procedimiento se repite hasta completar la longitud deseada. Aplicamos esta técnica en dos niveles de tokenización distintos. Aunque el proceso general de entrenamiento es similar en ambos casos, sus diferencias permiten observar variaciones en la forma en que cada tokenización procesa la información y genera resultados. Para cada una, entrenamos dos variantes de modelo: uno sencillo, basado en una única red neuronal recurrente, y otro más complejo, compuesto por dos capas recurrentes en secuencia y con dimensiones internas mayores.

Preprocesamiento, Entrenamiento y Generación de Muestras: El flujo completo del sistema generador se estructura en tres etapas integradas: (i) preprocesamiento y preparación del dataset, (ii) diseño y entrenamiento de los modelos, y (iii) generación automática de muestras. Partimos del archivo generator dataset standardized.txt, previamente normalizado, en el que cada línea contiene un fragmento de código vulnerable. El procesamiento comienza con la tokenización carácter a carácter mediante una capa StringLookup de TensorFlow, dando lugar a un vocabulario de 99 símbolos. Posteriormente se aplica una estrategia de etiquetado autogenerado propia de tareas de modelado del lenguaje: para una secuencia de longitud *sequence_length*, la entrada abarca los tokens *sequence_length -1* mientras que la etiqueta corresponde a la misma secuencia desplazada una posición a la derecha. Esto permite al modelo aprender dependencias contextuales a partir de una ventana fija; en este trabajo se emplea una longitud de 1024 tokens. Tras el etiquetado, las secuencias son vectorizadas para su uso en los modelos.

Con el dataset preparado, se implementan dos arquitecturas recurrentes basadas en LSTM, idénticas en flujo pero de distinta capacidad. Ambas reciben secuencias de 1024 tokens y producen logits del tamaño del vocabulario. La primera arquitectura (básica) emplea una capa Embedding de dimensión 64, seguida de una LSTM de 512 unidades, una capa Dropout con rate 0.2 y una capa Dense final. La versión compleja incrementa la capacidad mediante incrustaciones de 128 dimensiones y dos capas LSTM consecutivas de 1024 unidades cada una, ambas seguidas de Dropout. Los modelos se entrenan usando la función de pérdida Sparse Categorical Cross-Entropy, el optimizador Adam y un mecanismo de Early Stopping con patience = 3 y un umbral mínimo de mejora de 0.075. El entrenamiento finaliza tras 26 épocas para el modelo básico (pérdida $=$ 0.98) y 41 para el complejo (pérdida = 0.55), indicando una mayor capacidad de generalización de este último.

Para la fase de generación se implementa una clase generadora que actúa como interfaz con cada modelo. Este componente permite producir secuencias de longitud variable repitiendo iterativamente el proceso de inferencia: vectorización de la entrada, obtención de predicciones y estados internos, aplicación de una máscara para eliminar tokens no deseados, muestreo categórico sobre los logits (evitando un comportamiento determinista asociado al argmax), conversión del token generado a carácter y propagación del estado a la siguiente iteración. Además, se define un mecanismo de inicialización de secuencias a partir de 1331 palabras únicas extraídas del dataset, seleccionadas aleatoriamente. Para cada modelo se generan 500 muestras, variando aleatoriamente la longitud entre un mínimo y un máximo establecidos, y se almacenan en archivos de texto para su análisis posterior mediante el discriminador.

Modelos con tokenización en palabras

El modelo tokenizado por palabras se fundamenta en la misma base que el tokenizado por caracteres, trataremos de dar explicación a sus diferencias antes que volver a tratar de explicar lo que es común a ambos.

Preprocesamiento, Entrenamiento y Generación de Muestras: En esta variante, el procesamiento de las muestras se realiza a nivel de palabra, considerando cada token como una unidad separada por espacios. Esto hace especialmente relevante el proceso de estandarización, cuya función es reducir la variabilidad léxica y evitar la proliferación de tokens distintos que representan el mismo elemento. Sin esta normalización, el vocabulario crecería de forma incontrolada, por ejemplo: del token real *void* los tokens duplicados serían *void(, void() , void(){* , donde secuencias sin separación adecuada se interpretarían como tokens independientes. Para

garantizar la manejabilidad del modelo se fija un límite de 20 000 tokens de vocabulario. Una vez definida la tokenización, se vectorizan las muestras estableciendo una longitud máxima de 256 tokens; las secuencias más largas se truncan y las más cortas se rellenan con el token [PAD], que posteriormente se enmascara durante el entrenamiento. Las etiquetas se generan aplicando la misma estrategia de desplazamiento unipaso utilizada en el modelo basado en caracteres.

Los modelos construidos para esta configuración siguen la misma estructura general que los anteriores, pero sustituyen las celdas LSTM por redes GRU y ajustan su dimensionalidad a la mayor escala del vocabulario. Ambos comienzan con una capa Embedding de dimensión 128 en el modelo básico y 256 en el modelo complejo seguida de una o dos capas GRU, respectivamente, cada una con un dropout del 20 %. La capa final es una Dense de tamaño igual al vocabulario (20 000 unidades), lo que permite devolver la probabilidad para cada token. El modelo básico utiliza una GRU de 128 unidades, mientras que el complejo emplea una pila de dos GRU de 256 unidades cada una. Los modelos se entrenan utilizando el mismo esquema que en la arquitectura por caracteres: pérdida Sparse Categorical Cross-Entropy, optimizador Adam y un Early Stopping basado en un umbral mínimo de mejora de 0.075. El modelo básico converge tras 87 épocas con una pérdida de 0.9819, mientras que el modelo complejo lo hace tras 89 épocas, alcanzando un valor de pérdida de 0.5615.

Modelos LLM

Los modelos LLM son sistemas diseñados para comprender y generar lenguaje natural con alta precisión. Entrenados con grandes volúmenes de datos y dotados de miles de millones de parámetros, capturan patrones complejos que modelos más pequeños no pueden representar. Su entrenamiento inicial requiere infraestructuras avanzadas, y posteriormente pueden especializarse mediante fine-tuning para tareas concretas, como la generación de código.

La mayoría de los LLM se basa en arquitecturas transformer, cuyo mecanismo de atención permite identificar las partes más relevantes de una secuencia. Al no depender de un procesamiento estrictamente secuencial, ofrecen mayor eficiencia y rendimiento que las redes recurrentes tradicionales.

GPT-2: El primer modelo que se ajustó fue el GPT-2 para generar muestras vulnerables, con una versión completa alcanza 1,5 millones de parámetros. Para este trabajo se utilizó un modelo preentrenado para simplificar el proceso de tokenización, división de secuencias y enmascaramiento de tokens. Se realizó la configuración de parámetros de manera tradicional, do_sample=True, para el muestreo aleatorio, top_k= 50 para limitar la selección a los tokens con mayor probabilidad y top_p=0.95 para aplicar nucleus sampling. Las muestras generadas se estandarizan eliminando ruidos como saltos de línea, espacios duplicados y comentarios, almacenándose 500 de ellas para su análisis posterior.

CodeBERT: A partir de este modelo, han surgido diversas variantes orientadas a tareas específicas; entre ellas, CodeBERT, diseñado para comprender código y lenguaje natural. Para esta investigación se emplea CodeBERT para la generación de código vulnerable, siguiendo un flujo de trabajo basado en un modelo preentrenado para tareas de análisis de código. En particular, se utiliza la arquitectura RoBERTa especializada correspondiente como base del modelo.

CodeBERT tiene un tamaño comparable al modelo base de GPT-2, con 125 millones de parámetros. Su entrenamiento se realizó durante 10 épocas, punto en el que la pérdida se estabiliza, alcanzando un valor final de 0.654.

La generación de muestras sigue el procedimiento habitual: se parte de una secuencia inicial y se solicita al modelo predecir el siguiente token. Para ello se añade un token mask, se obtiene la predicción, y se repite el proceso hasta alcanzar la longitud deseada. Los logits devueltos por el modelo se transforman mediante softmax y se muestrean con una distribución multinomial usando Torch, obteniendo el siguiente token que se incorpora a la secuencia. Finalmente, almacenamos 500 muestras generadas para su posterior análisis.

III-C. Desarrollo del discriminador

El discriminador cuantifica principalmente el rendimiento de los modelos generativos. Este está entrenado con la mitad de las muestras vulnerables disponibles en el dataset, y la otra mitad de muestras no vulnerables equivalentes en tamaño para el equilibrio de las clases. Con esto se reduce el problema a una clasificación binaria, donde el modelo debe aprender a distinguir entre código vulnerable y no vulnerable.

Preprocesamiento, Entrenamiento y Generación de Muestras: El discriminador utiliza *discriminator_dataset.csv*, tomando *func_before* como entrada y *vul* como etiqueta. El dataset se divide en entrenamiento, validación y test (80/10/10). La estandarización, tokenización y vectorización se realiza con la capa TextVectorization de TensorFlow, usando tokenización por palabras y limitando el vocabulario a 30.000 tokens, independiente del preprocesamiento de los generadores para permitir experimentos sin modificar los datos originales. El modelo realiza una clasificación binaria de vulnerabilidad con umbral 0.5. Su arquitectura incluye: capa embedding de 256 dimensiones, LSTM bidireccional, regularizador L2, dropout de 0.3 y capa Dense con activación sigmoide. El entrenamiento se detiene mediante EarlyStopping cuando la pérdida de validación no mejora al menos 0.02 durante tres épocas consecutivas.

El entrenamiento finalizó en 7 épocas, alcanzando el mejor rendimiento en la cuarta, donde se restauraron los pesos. Después de este punto, la mejora ocurre solo en el conjunto de entrenamiento, indicando sobreajuste.

IV. Experimentos y Resultados

Tras generar un conjunto de muestras con cada generador, evaluamos su capacidad para engañar al discriminador, centrándonos en los falsos negativos (FN), es decir, muestras vulnerables clasificadas erróneamente como no vulnerables. Dado que todas las muestras generadas corresponden a la clase positiva, los verdaderos negativos (TN) no existen, por lo que métricas como precisión o F1-score no aportan información relevante. La métrica más adecuada para nuestro análisis es la exactitud, que representa la proporción de predicciones correctas sobre el total.

Los resultados obtenidos con el discriminador muestran un comportamiento ligado a la naturaleza y complejidad de los modelos generadores:

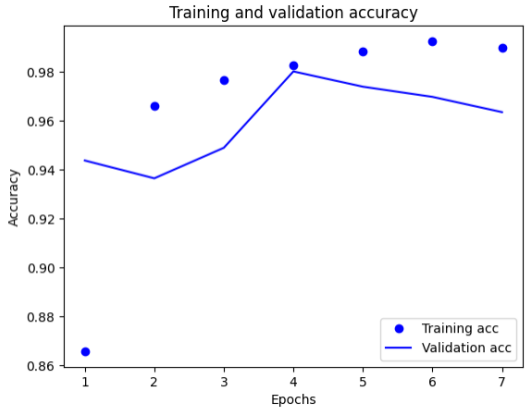


Figura 1. Precisión del discriminador

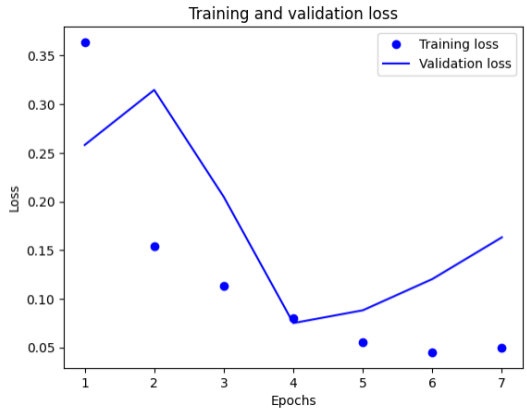


Figura 2. Pérdida del discriminador

- Modelos tokenizados por caracteres: presentan el peor desempeño. El modelo DoubleLSTM, de mayor complejidad, duplica el rendimiento de la versión básica LSTM, indicando que las secuencias por caracteres requieren estructuras más profundas para capturar dependencias coherentes.
- Modelos tokenizados por palabras: logran un engaño intermedio. La versión básica GRU alcanza un 52,4 % de falsos negativos, mientras que la versión más compleja DoubleGRU no mejora, sugiriendo que la mayor complejidad puede conducir a sobreajuste sin beneficios adicionales.
- Modelos LLM sobre transformadores: muestran los mejores resultados. GPT-2 obtiene la tasa de engaño más alta, y CodeBERT también supera a los modelos anteriores (excepto la GRU básica por palabras), logrando engañar al discriminador en más de la mitad de los casos.

A continuación, en la Figura 3 se muestra la tasa de engaño alcanzada por cada generador según la predicción del discriminador. Como se observa, los modelos LLM, en particular GPT-2, alcanzan los mejores resultados, mientras que los modelos tokenizados por caracteres presentan las tasas más bajas.

Estos resultados permiten varias conclusiones: primero, los modelos basados en transformadores son los más eficaces para la generación de código vulnerable; segundo, una mayor complejidad de la arquitectura no garantiza un mejor desempeño,

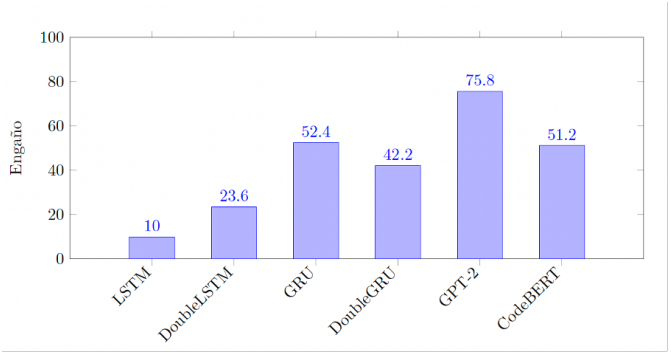


Figura 3. Tasa de engaños al discriminador por modelo

pues depende de la naturaleza del problema y del tamaño del vocabulario. En el caso de los generadores por caracteres, una estructura más profunda es beneficiosa, mientras que en los generadores por palabras, estructuras más complejas no necesariamente aportan mejoras.

IV-A. Rendimiento de los modelos

El tiempo de entrenamiento de los modelos depende de la arquitectura y del equipo utilizado; dado que todos se entrenaron en la misma máquina y con el mismo conjunto de datos, es posible compararlos. Todos los modelos se entrenaron hasta que la pérdida dejó de mejorar.

Como se observa en la Figura 4 el modelo DoubleLSTM presenta el mayor tiempo de entrenamiento, aproximadamente el doble que el siguiente, debido a su estructura secuencial y la dependencia directa entre sus dos redes LSTM. El DoubleGRU sigue un patrón similar, pero gracias a sus celdas más simples reduce a la mitad el tiempo comparado con DoubleLSTM.

Los modelos basados en transformadores, GPT-2 y CodeBERT, muestran tiempos de entrenamiento más cortos a pesar de sus 120 millones de parámetros, gracias al paralelismo que permiten estas arquitecturas, destacando su eficiencia frente a redes recurrentes.

Finalmente, el discriminador basado en LSTM bidireccional, dedicado a clasificación binaria, requiere mucho menos tiempo, evidenciando que la detección de patrones es significativamente menos compleja que la generación de muestras.

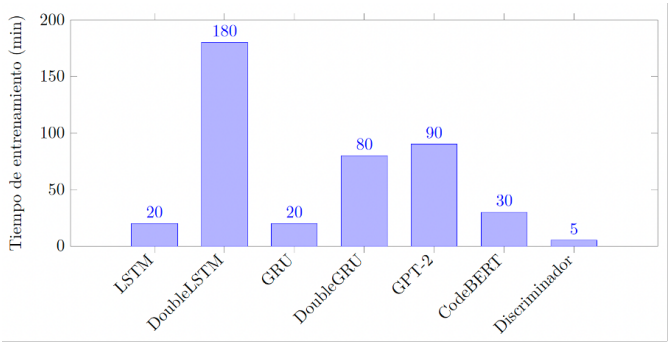


Figura 4. Rendimiento de los modelos

IV-B. Análisis de las Muestras

Se realizó un análisis de calidad de las muestras generadas, considerando la coherencia y el contexto del código.

- Modelos tokenizados por caracteres: El modelo básico pierde caracteres y contexto más fácilmente, mientras que el modelo complejo mantiene mejor la relación entre palabras, aunque puede fallar en detalles menores. La Figura 5 muestra ejemplos representativos de cada modelo. La tokenización por caracteres mantiene un contexto sólido, aunque limita la paralelización.

Básico	for (sip = 0 ; dont (invalidated . release) ; }
Complejo	for (i = 0 ; i < argc ; + +)

Figura 5. Ejemplos de código en modelos por caracteres

- Modelos tokenizados por palabras: Se observa una tendencia a generar tokens frecuentes (`'comodín'`) como if o) , penalizando tokens menos comunes y afectando la coherencia en bloques de código. Esto requiere embeddings grandes y datasets extensos. La Figura 6 ilustra estos errores típicos.

Básico	if = (char) ((unsigned)) ;
Complejo	if = (const + 6) ; break ; }

Figura 6. Ejemplos de código en modelos de palabras

- Modelos LLM: GPT-2 mantiene dependencias en secuencias largas con resultados casi perfectos, mientras que CodeBERT se ve penalizado por su enfoque bidireccional en tareas unidireccionales. La Figura 7 muestra ejemplos representativos.

GPT-2

```
PolygonExport int GetPolygonOpacity ( VncState * Vnc, const
int Opacity, int HasPolygonOpacity ) {
        Bool IsDefined ( Vnc -> HasPolygonOpacity ) ; return
        Opacity > = VP9_OPEN_OPEN_NONE ;
}
```

CodeBERT

```
case RT_BUG: return -EFAULT ;
case RT_ERROR: return -EFAULT ;
case RT_PRESS: return -EFAULT ;
case RT_USER: return -EFAULT ;
case RT_GROUP: return -EFAULT ;
```

Figura 7. Muestras generadas por LLM

V. Conclusiones

Los experimentos muestran que los modelos basados en transformadores, especialmente GPT-2, son los más eficaces para generar código vulnerable, ya que mantienen mejor la coherencia y las dependencias del código. En cambio, los modelos recurrentes tienen un rendimiento más irregular: las arquitecturas profundas por caracteres sí, mejoran los resultados, pero en los modelos por palabras la mayor complejidad no aporta ventajas claras y puede llevar a sobreajuste.

El análisis de las muestras confirma estas tendencias. Los modelos por caracteres conservan el contexto general, aunque con errores en detalles; los modelos por palabras tienden a repetir tokens frecuentes, lo que afecta la estructura del código; y los LLM generan fragmentos más completos y consistentes.

En términos de rendimiento, los transformadores entrenan más rápido que las redes recurrentes profundas gracias al paralelismo, y el discriminador resulta mucho más ligero al tratarse de una tarea de clasificación. En conjunto, los resultados indican que los modelos LLM ofrecen la vía más sólida para la generación de código vulnerable, mientras que los modelos recurrentes siguen siendo útiles para la detección por su simplicidad y estabilidad.

Agradecimientos

Este trabajo se ha realizado en el marco de los fondos del Plan de Recuperación, Transformacionales y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra "Ciberseguridad para la Innovación y la Protección y la Protección Digital" (Cátedra INCIBE-UCM). Asimismo, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, Proyecto CIRMA-CM (TEC-2024/COM-404).

Referencias

[1] D. Zou, S. Wang, S. Xu, Z. Li, and H. Jin, " μVulDeePecker: A Deep Learning-Based System for Multiclass Vulnerability Detection ," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 05, pp. 2224–2236, Sep. 2021. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/TDSC.2019.2942930

[2] Z. Yang, J. W. Keung, Z. Sun, Y. Zhao, G. Li, Z. Jin, S. Liu, and Y. Li, "Improving domain-specific neural code generation with few-shot meta-learning," *Information and Software Technology*, vol. 166, p. 107365, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950584923002203

[3] S. Almanasra and K. Suwais, "Analysis of chatgpt-generated codes across multiple programming languages," *IEEE Access*, vol. 13, pp. 23 580–23 596, 2025.

[4] A. Jain, P. William, F. T. Ayasrah, G. P. Lakshmi, and T. D. Diwan, "Analyzing automatic code generation for learning models in generative ai," in *2024 11th International Conference on Software Defined Systems (SDS)*, 2024, pp. 50–58.

[5] C. Xie, M. Jiao, X. Gu, and B. Shen, "Empowering ai to generate better ai code: Guided generation of deep learning projects with llms," in *2025 IEEE 49th Annual Computers, Software, and Applications Conference (COMPSAC)*, 2025, pp. 1394–1399.

[6] M. Paavilainen, A. Ylihurulaa, V. Niinimäki, A. Kumar, L. Loven, S. Pirttikangas, and S. Tarkoma, "Software development for scientific computing using ai code generation," WorkingPaper, Jan. 2025, a summerized version of the article was presented as a poster at AI Day 2024 + Nordic AI Meet in October 2024.

[7] W. Ma, Y. Song, M. Xue, S. Wen, and Y. Xiang, "The "code" of ethics: A holistic audit of ai code generators," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 5, pp. 4997–5013, 2024.

[8] G. Vargas-Monroy, D.-B. Gonzalez-Roldan, C. E. Montenegro-Marín, A.-P. Daza-Corredor, and D.-D. Leal-Lara, "Code generation system based on mda and convolutional neural networks," *Frontiers in Artificial Intelligence*, vol. Volume 8 - 2025, 2025. [Online]. Available: https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2025.1491958

[9] N. Ikumapayi, "Automated front-end code generation using openai: Empowering web development efficiency," *SSRN Electronic Journal*, 01 2023.

[10] N. Tihanyi, T. Bisztray, M. A. Ferrag, R. Jain, and L. Cordeiro, "How secure is ai-generated code: a large-scale comparison of large language models," *Empirical Software Engineering*, vol. 30, 12 2024.

Random Graph Models for the Lightning Network: A Comparative Topology Study

Yasser Hayali, Cristina Pérez-Solà, Jordi Herrera-Joancomartí
Departament d'Enginyeria de la Informació i les Comunicacions
Universitat Autònoma de Barcelona (UAB)
Yasser.Hayali@autonoma.cat, Cristina.Perez@uab.cat, Jordi.Herrera@uab.cat

***Abstract*—The Lightning Network (LN) is a second-layer payment protocol for Bitcoin whose behavior critically depends on its payment-channel graph. Realistic graph models are needed to evaluate routing, upgrades, and attacks in simulation. We compare four random graph families—Erdős–Rényi, Barabási–Albert (BA), Watts–Strogatz, and a dynamic disassortative BA variant—against a recent LN mainnet snapshot. For each model, we generate graphs of comparable size and analyze path lengths, clustering, assortativity, centrality, bridges, and articulation points. Our results show that no single model matches all LN properties, and we argue that a hybrid generator combining preferential attachment, disassortativity, and triadic closure is needed.**

***Index Terms*—lightning network, payment-channel networks, random graph models, network topology, assortativity**

I. Introduction

The Lightning Network (LN) is a second-layer payment protocol built on top of Bitcoin that enables fast, low-cost, and scalable transactions by moving most payments off-chain [1], [2], [3]. Instead of broadcasting every transaction to the Bitcoin blockchain, users open payment channels (bidirectional connections through which multiple payments can be routed without requiring global consensus for each one). Payments are then forwarded across a network of interconnected channels using multi-hop routing, allowing participants who do not share a direct channel to transact securely. This design significantly reduces on-chain congestion and transaction fees, while preserving the decentralized nature of Bitcoin. However, the LN's decentralized and evolving topology introduces complex network dynamics that are difficult to study without accurate modeling and simulation.

To test protocol upgrades, evaluate path-finding algorithms, and analyze potential attack vectors, researchers and engineers require a reliable framework for experimenting with the Lightning Network (LN) at scale. Conducting such experiments directly on the live network is often impractical, costly, or even unsafe, as it may disrupt real users or expose vulnerabilities. Consequently, a common and safer approach is to simulate the LN in a controlled environment.

A key component of any such simulation is the channel graph, a representation of all participating nodes and the payment channels connecting them. The structure of this graph underpins nearly every aspect of the LN's behavior, from routing efficiency to network resilience and privacy [4], [5], [6]. Therefore, it is crucial that the simulated channel graph closely mirrors the topology of the live network. If the generated graph diverges significantly from reality, the results of subsequent experiments (such as protocol evaluations or attack simulations) may be misleading or invalid.

In this paper, we systematically evaluate which random graph models are best suited to reproduce the structural properties of the real Lightning Network. Our goal is to identify generative models that most accurately capture the LN's topological characteristics, providing a foundation for realistic and reproducible simulations.

The rest of the paper is organized as follows: Section II reviews related work on empirical analyses of the Lightning Network and on random graph models. Section III introduces the LN channel graph, the generative models we consider, and the graph metrics used in our comparison. Section IV presents our experimental results. Finally, Section V concludes and outlines directions for future work.

II. Related work

Several empirical studies have analyzed public snapshots of the LN channel graph, focusing on metrics such as degree distribution, centrality, and assortativity [5], [6]. Early works observed the emergence of highly connected hubs and studied the impact of this structure on efficiency and resilience, as well as on privacy and centralization risks [5]. More recent longitudinal analyses follow the evolution of the network over time, reporting a persistent scale-free degree distribution and strongly negative assortativity, i.e., a disassortative tendency where high-degree nodes connect preferentially to low-degree nodes [5], [6].

A second line of work has developed tools and frameworks to simulate LN-like environments. Frameworks such as CLoTH [7] and similar discrete-event simulators model payment-channel networks and evaluate payment-level performance indicators like success probability and latency. For application development and testing, tools such as Polar [8] or Warnet [9] allow users to spin up local Bitcoin and LN nodes or large emulated networks, mainly targeting functional testing and protocol experimentation rather than an accurate reproduction of the real LN topology.

Regarding graph modeling, several authors have compared the LN channel graph with classical random graph models. Erdős–Rényi graphs and Barabási–Albert (BA) preferential-attachment graphs are natural baselines that match, respectively, only the density or the heavy-tailed degree distribution of the LN, but they fail to reproduce other structural features such as clustering or assortativity [10], [11], [12], [13]. Watts–Strogatz small-world graphs capture high clustering and short path lengths, but their degree distribution is much more homogeneous than what is observed in the LN [14].

To better match the empirical properties of the LN, disassortative variants of scale-free models have been proposed. In

particular, dynamic BA models inspired by LN data introduce edge deletion mechanisms whose probabilities depend on the degrees (and, in some variants, the capacities) of the endpoints [15]. By tuning a small number of parameters, these models preserve a power-law degree distribution while driving the assortativity towards negative values close to those measured in real LN snapshots [15].

Our work is complementary to these previous proposals. Instead of introducing yet another model, we take a recent snapshot of the LN and systematically compare several families of random graphs—Erdős–Rényi, Barabási–Albert, Watts–Strogatz and dynamic disassortative BA—generated with the same number n of nodes and the same number m of edges as the real network. The comparison is carried out across a broad set of structural metrics, with the goal of identifying which models and parameter ranges provide the closest approximation to the current LN topology.

III. Preliminaries

In this section we first present the LN channel graph, we briefly recall the random graph models that we will compare with the LN channel graph, and we introduce the graph-theoretic metrics used in our analysis.

A. The LN channel graph

The Lightning Network (LN) is a second-layer payment protocol built on top of Bitcoin that enables fast, low-fee payments through off-chain payment channels. A payment channel is a two-party smart contract funded on-chain, within which the participants can update their balances by exchanging signed commitment transactions without broadcasting them to the blockchain. Payments between two non-adjacent nodes are routed along paths of payment channels using hashed timelock contracts.

This routing structure induces a natural graph representation of the LN. At the gossip level, the LN topology is a directed, weighted graph: each public channel between nodes u and v is advertised by up to two `channel_update` messages (one per direction), and each direction has its own capacity and fee parameters.

For the purposes of this work, we are interested in the large-scale connectivity structure rather than directional or capacity information. We therefore work with the undirected projection

$$G_{\mathrm{LN}} = (V_{\mathrm{LN}}, E_{\mathrm{LN}}),$$

where V_{LN} is the set of LN nodes and $\{u, v\} \in E_{\mathrm{LN}}$ if a public payment channel between u and v exists in the gossip data. We denote by $n = |V_{\mathrm{LN}}|$ and $m = |E_{\mathrm{LN}}|$ the number of nodes and edges in this snapshot, and all graph-theoretic metrics in this paper are computed on G_{LN}.

B. Random graph models

In this section we recall the random graph models that we use to approximate the LN channel graph. For each LN snapshot with node set V_{LN} and edge set E_{LN}, we generate random graphs with n nodes and m edges, with:

$$n = |V_{\mathrm{LN}}|, \qquad m = |E_{\mathrm{LN}}|,$$

so that size and density match the observed network.

1) Erdős–Rényi (ER) $G(n, m)$ model: In this model, a graph is sampled uniformly at random from the set of all simple undirected graphs with n nodes and exactly m edges [10].[1] Equivalently, we choose an edge set of size m uniformly among all $\binom{\binom{n}{2}}{m}$ possible edge subsets. In our comparison we always take

$$n = |V_{\mathrm{LN}}|, \qquad m = |E_{\mathrm{LN}}|,$$

so that size and density match the observed Lightning Network snapshot.

2) Barabási–Albert (BA) model: This model generates scale-free networks via growth and preferential attachment [11]. Starting from a small seed graph G_0 with m_0 nodes, at each step a new node is added and connected to m existing nodes. Each endpoint u is chosen with probability proportional to its current degree,

$$\Pr[\text{new node connects to } u] \propto \frac{\deg(u)}{\sum_w \deg(w)}.$$

This mechanism produces a heavy-tailed degree distribution. In our experiments we use the Barabási–Albert model with

$$n = |V_{\mathrm{LN}}|, \qquad m = \left\lceil \frac{|E_{\mathrm{LN}}|}{|V_{\mathrm{LN}}|} \right\rceil,$$

so that the expected number of edges is comparable to the LN snapshot.

3) Watts–Strogatz (WS) model: This model generates small-world graphs combining high clustering with short average path lengths [14]. It is defined by parameters (n, k, p), where k is an even integer and $p \in [0, 1]$ is a rewiring probability. The construction process follows two stages:

1) Start from a ring lattice on n nodes where each node is connected to its $k/2$ nearest neighbors on each side.
2) For each edge, with probability p rewire one endpoint to a uniformly random node, avoiding self-loops and multiple edges.

For $p = 0$ the graph is a regular lattice; for p close to 1 it approaches a random graph. We use the Watts–Strogatz model with:

$$n = |V_{\mathrm{LN}}|, \qquad k = \left\lceil \frac{2|E_{\mathrm{LN}}|}{|V_{\mathrm{LN}}|} \right\rceil,$$

and explore several rewiring probabilities $p \in \{0.0, 0.1, \ldots, 1.0\}$.

4) Dynamic Disassortative Barabási–Albert (ddBA) model: This model extends the Barabási–Albert (BA) construction by allowing edges to disappear during the growth process, with disappearance probabilities that depend on the degrees of their endpoints (and optionally on edge capacities) [15]. A typical variant is parameterized by the number of nodes n, the number of neighbors d each new node connects to, a parameter $\beta > 0$ controlling the disappearance rate, and an initial graph G_0. At each step:

[1] In this paper, we adopt the uniform variant of the Erdős–Rényi (ER) model, also known as $G(n, m)$. An alternative formulation of the ER model specifies a connection probability p rather than a fixed number of expected edges m. Because the total number of edges in our network is known a priori, the uniform (fixed-edge) variant is more appropriate for our analysis and is therefore used in the paper.

1) A new node is added and connects to d existing nodes according to preferential attachment, as in the BA model.
2) Each existing edge $\{u, v\}$ may disappear with a probability that decreases with the degrees of u and v, for example:

$$q(u, v) = e^{-\beta \cdot \max\{\deg(u), \deg(v)\}}.$$

This model preserves a heavy-tailed degree distribution while driving the assortativity towards negative values that can be tuned through β.

In this work we consider the ddBA model with parameters

$$n = |V_{\mathrm{LN}}|, \qquad d = \left\lceil \frac{|E_{\mathrm{LN}}|}{|V_{\mathrm{LN}}|} \right\rceil,$$

and we explore several values of β in the range $[0.1, 0.9]$.

C. Graph metrics

Let $G = (V, E)$ be a simple undirected graph with $n = |V|$ vertices and $m = |E|$ edges. In this section, we briefly describe the metrics used in our comparison. When G is not connected, these metrics are computed on each connected component separately.

1) Size and connected components: For each graph we record the total number of vertices n and edges m, and the number of connected components. For a given connected component C, with vertex set V_C and edge set E_C, we denote:

$$n_C = |V_C|, \qquad m_C = |E_C|.$$

2) Distances, radius and diameter: For vertices $u, v \in V$, the (graph) distance $d(u, v)$ is the length, in number of edges, of a shortest path between u and v. The average shortest-path length of a connected graph is

$$\ell(G) = \frac{1}{n(n-1)} \sum_{u \neq v} d(u, v).$$

The eccentricity of a vertex u is:

$$\mathrm{ecc}(u) = \max_{v \in V} d(u, v).$$

The radius and diameter of G are then defined as:

$$\mathrm{rad}(G) = \min_{u \in V} \{\mathrm{ecc}(u)\}, \qquad \mathrm{diam}(G) = \max_{u \in V} \{\mathrm{ecc}(u)\}.$$

3) Clustering coefficient: For a vertex u with degree $\deg(u) \geq 2$, let $N(u)$ denote its neighborhood. The local clustering coefficient of u is

$$C(u) = \frac{\left|\{\{x, y\} \subseteq N(u) : \{x, y\} \in E\}\right|}{\binom{\deg(u)}{2}},$$

i.e., the fraction of pairs of neighbors of u that are adjacent. The (global) average clustering coefficient of G is the arithmetic mean of $C(u)$ over all vertices with degree at least two [4]. This metric captures the tendency of neighbors of a vertex to be interconnected.

4) Bridges and articulation points: An edge $e \in E$ is a bridge if its removal increases the number of connected components of the graph. A vertex $u \in V$ is an articulation point if removing u and all its incident edges increases the number of connected components. The numbers of bridges and articulation points provide simple indicators of structural fragility.

5) Degree assortativity: Let $e = \{x, y\}$ denote a uniformly random edge of G, and let K_x and K_y be the degrees of its endpoints. The degree assortativity coefficient is the Pearson correlation coefficient of K_x and K_y:

$$r = \frac{\mathrm{Cov}(K_x, K_y)}{\sqrt{\mathrm{Var}(K_x)\ \mathrm{Var}(K_y)}} \in [-1, 1].$$

Values $r > 0$ indicate assortative mixing (vertices tend to connect to others of similar degree), whereas $r < 0$ indicate disassortative mixing (high-degree vertices tend to connect to low-degree ones). [12]

6) Centrality measures and Freeman centralization: We consider three standard notions of vertex centrality.

Degree centrality.: The degree centrality of a vertex u is

$$C_D(u) = \frac{\deg(u)}{n-1},$$

which normalizes the degree to lie in $[0, 1]$.

Betweenness centrality.: For distinct vertices $s, t \in V$, let σ_{st} be the number of shortest paths from s to t, and let $\sigma_{st}(u)$ be the number of those paths that pass through u. The betweenness centrality of u is defined as:

$$C_B(u) = \sum_{\substack{s,t \in V \\ s \neq t,\, s \neq u,\, t \neq u}} \frac{\sigma_{st}(u)}{\sigma_{st}}.$$

Closeness centrality.: The closeness centrality of a vertex u is the inverse of its average distance to all other vertices:

$$C_C(u) = \frac{n-1}{\sum_{v \in V \setminus \{u\}} d(u, v)}.$$

To describe how centrality is distributed across the graph, we use the Freeman closeness centralization index. Let

$$C_C^{\max} = \max_{u \in V} C_C(u)$$

be the maximum closeness centrality in G. The (normalized) closeness centralization is

$$C_C^{\mathrm{F}}(G) = \frac{\sum_{u \in V} \left(C_C^{\max} - C_C(u)\right)}{\frac{(n-1)(n-2)}{2n-3}}.$$

The numerator measures the total deviation from the most central vertex, and the denominator is a normalizing factor chosen so that $0 \leq C_C^{\mathrm{F}}(G) \leq 1$ for connected graphs [16]. Small values indicate that closeness centrality is relatively evenly distributed; values closer to 1 correspond to highly centralized structures.

IV. Experimental results

We now evaluate how well several random graph models reproduce the structural properties of the Lightning Network. Using a recent mainnet snapshot as ground truth, we generate synthetic graphs with a similar number of nodes and edges (n, m) and compare them across standard metrics; unless stated otherwise, for each parameter setting we report the mean average of 15 independent runs.

A. Data collection

We obtained the Lightning Network (LN) topology by operating a Core Lightning (formerly c-lightning) node [17] that collects the public channel graph via the gossip protocol. After allowing the node approximately two weeks to fully synchronize, we captured a mainnet snapshot on 3 September 2025, contemporaneous with our experiments. This snapshot contains 11,555 nodes and 41,578 edges—figures comparable to datasets reported as early as January 2022 [13], [18]. We treat this snapshot as the ground truth for all comparisons. It records standard properties such as the number of nodes (n), the number of edges (m), and channel capacities. Although a channel is ideally represented by two directed edges (one per direction), our snapshot did not include both directions for every channel; when converting the directed graph of the snapshot to an undirected graph, we therefore add a single undirected edge for any non-reciprocated directed edge.

B. Experimental setup

For each random graph model, we generated synthetic graphs that attempt to match the LN snapshot size (with a number of nodes (n) and edges (m) comparable to that of the LN snapshot). This ensures a fair comparison, and thus any differences in metrics are due to structural characteristics of the models rather than trivial size or density differences.

We evaluate four families of models: Erdős–Rényi $G(n, m)$ (ER), Barabási–Albert (BA), Watts–Strogatz (WS) with degree $k \approx 2m/n$ and rewiring probability $p = 0.4$; we fix $p = 0.4$ because it produces graphs whose clustering coefficient matches that of the LN snapshot. For the ddBA model, we evolve the network for $2{,}000$ time steps (days) and explore several values of β; we fix $\beta = 0.4$ because it yields assortativity similar to that of the LN snapshot. For each model–parameter setting we generate 15 independent instances and compute the metrics defined in Section III (degree distribution, average shortest-path length, radius/diameter, clustering, assortativity, centrality and centralization). Per setting, we report the mean over iterations.

C. Random graph models comparison

Table I reports all metric values for the compared graph models; entries in **bold** indicate the metric values that best match those of the Lightning Network (LN) snapshot.

Overall, no single model reproduces all structural characteristics of the LN snapshot; each captures a different subset of features. Erdős–Rényi (ER) and Watts–Strogatz (WS) models do not generate pronounced hubs, whereas Barabási–Albert (BA) and dynamic disassortative BA (ddBA) models yield heavy-tailed degree distributions with dominant hubs, more closely resembling LN. The LN snapshot is markedly disassortative ($r < 0$), and this property is best approximated by ddBA. LN exhibits modest but non-negligible clustering; WS at intermediate rewiring probabilities can match this level, while ER, BA, and ddBA display substantially lower clustering. All models are small-world; BA produces the shortest path lengths, whereas LN has a larger diameter, which ddBA partially reduces by discouraging hub–hub connections. Finally, centralization in LN is high and is most closely matched by BA and ddBA, while ER and WS remain comparatively homogeneous.

Takeaway. Preferential attachment (BA) is essential to reproduce LN's heavy-tailed degrees and centralization; explicit disassortativity (ddBA) is needed to match LN's negative mixing and larger diameter; and a small-world mechanism (WS with intermediate p) is needed to reach LN-like clustering. A topology generator that closely matches LN structural properties will likely need to combine these ingredients.

V. Conclusions and future work

Accurate graph models of the Lightning Network (LN) are crucial for safely simulating protocol upgrades, routing algorithms, and attack vectors; if the synthetic topology diverges from the real LN, the resulting conclusions may be misleading.

Our comparison shows that no single model of the ones we tested reproduces all properties of the Lightning Network (LN). ER attains short paths but lacks hubs; WS at intermediate p matches LN's modest clustering yet remains near-regular; BA yields heavy-tailed degrees and high centralization but only weakly negative assortativity; the dynamic disassortative BA (ddBA) variant achieves LN-like $r < 0$ and a larger diameter, while still underestimating clustering. Taken together, these results indicate that a realistic LN generator must blend the emergence of high-degree hubs, disassortative mixing, and limited local closure.

A compact hybrid model tailored to the Lightning Network should combine three ingredients: (i) *preferential attachment* (a rich–get–richer rule) to create hubs; (ii) an explicit *disassortative bias* that downweights connections between nodes with similar degree so the network remains negatively assortative and the diameter increases toward LN values; and (iii) *triad closure* which, when attaching an edge between (x, y), is the probability of adding a new edge from x to a neighbor of y to increase clustering to LN levels. Optionally, the model should include *reciprocity* to mimic how channels are advertised on the Lightning Network, where each endpoint can independently announce its side of the channel, and channels are typically—but not always—advertised in both directions; and *capacity-aware* attachment, whereby high-capacity nodes are more likely to attract new channels, consistent with observations of the LN.

Future work includes (i) implementing and releasing the hybrid generator with reproducible seeds, (ii) extending it with capacity/fee-aware dynamics, and (iii) coupling generation with routing experiments (success rate, latency, and targeted-failure robustness) to evaluate its practical impact on LN-like deployments.

Table I
LN VS. RANDOM GRAPH MODELS (AVERAGES OVER 15 RUNS). METRICS ON THE LARGEST CONNECTED COMPONENT. FOR WS WE USE $p = 0.40$ (CHOSEN TO MATCH LN CLUSTERING); FOR DDBA WE USE $\beta = 0.4$ (CHOSEN TO MATCH LN ASSORTATIVITY).

Metric	**LN**	**ER**	**BA**	**WS**	**ddBA**
Nodes n	11,555	**11,547**	**11,555**	**11,555**	9,617
Edges m	41,578	**41,578**	46,204	23,110	19,579
Avg. path length	3.564	4.963	3.950	8.089	4.084
Diameter	9	**9**	6	13.64	7.60
Radius	5	6	4	10	4.40
Avg. clustering	0.1150	0.000592	0.00557	**0.1168**	0.0241
Assortativity r	-0.215	0.000534	-0.0308	-0.0786	**-0.2198**
Avg. degree centrality	6.030×10^{-4}	6.238×10^{-4}	6.922×10^{-4}	3.462×10^{-4}	4.234×10^{-4}
Avg. betweenness	2.260×10^{-4}	3.433×10^{-4}	2.554×10^{-4}	6.136×10^{-4}	3.208×10^{-4}
Avg. closeness	0.285	0.202	0.255	0.1238	0.2469
Closeness centralization (Freeman)	0.366	0.0531	0.273	0.0348	0.2793
Bridge count	6,368	61.53	0	0	3,651.20
Articulation points	916	61.33	0	0	652.53

ACKNOWLEDGMENT

This work is funded by the projects PID2021-125962OB-C33 SECURING/NET and PID2024-156914OB-C43 SAFE-BLOCKCHAIN, funded by the Ministerio de Ciencia e Innovación, la Agencia Estatal de Investigación and the European Regional Development Fund (ERDF); the DANGER Strategic Project of Cybersecurity C062/23, funded by the Spanish National Institute of Cybersecurity through the European Union - NextGenerationEU and the Recovery, Transformation and Resilience Plan; and the Catalan Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) grant SGR2021-00643.

REFERENCES

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008, white paper.
[2] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," https://lightning.network/lightning-network-paper.pdf, 2016, white paper.
[3] A. M. Antonopoulos, O. Osuntokun, and R. Pickhardt, *Mastering the Lightning Network: A Second Layer Blockchain Protocol for Instant Bitcoin Payments*. O'Reilly Media, 2021, https://github.com/lnbook/lnbook.
[4] M. Newman, *Networks: An Introduction*. Oxford University Press, 2010.
[5] S. Martinazzi and A. Flori, "The evolving topology of the lightning network: Centralization, efficiency, robustness, synchronization, and anonymity," *PLOS ONE*, vol. 15, no. 1, p. e0225966, 2020.
[6] P. Zabka, K. Förster, C. Decker, and S. Schmid, "A centrality analysis of the lightning network," *Telecommunications Policy*, vol. 48, no. 2, p. 102696, 2024.
[7] M. Conoscenti, A. Vetrò, and J. C. De Martin, "CLoTH: A lightning network simulator," *SoftwareX*, vol. 15, p. 100717, 2021.
[8] J. James, "Polar: Regtest lightning networks, made easy," https://lightningpolar.com/, 2019, desktop tool for creating local Bitcoin and Lightning Network test networks.
[9] bitcoin-dev-project, "Warnet: Monitor and analyze bitcoin networks," https://github.com/bitcoin-dev-project/warnet, 2023, framework for launching and monitoring large Bitcoin/Lightning test networks.
[10] P. Erdős and A. Rényi, "On random graphs i," *Publicationes Mathematicae Debrecen*, vol. 6, pp. 290–297, 1959.
[11] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.
[12] M. E. J. Newman, "Assortative mixing in networks," *Physical Review Letters*, vol. 89, no. 20, p. 208701, 2002.
[13] Z. Wang, R. Zhang, Y. Sun, H. Ding, and Q. Lv, "Can lightning network's autopilot function use BA model as the underlying network?" *Frontiers in Physics*, vol. 9, p. 794160, 2022.
[14] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
[15] T. E. Abedesselam, F. Giacomelli, and F. Pasquale, "Disassortative dynamic BA models inspired by the bitcoin lightning network," in *2025 IEEE International Conference on Pervasive Computing and Communications Workshops and Other Affiliated Events (PerCom Workshops)*. IEEE, 2025.
[16] L. C. Freeman, "Centrality in social networks: Conceptual clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1979.
[17] Core Lightning Developers, "Core lightning," https://github.com/ElementsProject/lightning, 2016, bitcoin Lightning Network implementation (formerly c-lightning).
[18] C. Decker, "Lightning network research – topology datasets," https://github.com/lnresearch/topology, 2018, dataset of historical Lightning Network topologies.

Revisión de tácticas, técnicas y procedimientos (TTPs) en MITRE ATT&CK para Compromiso del Correo Empresarial (BEC)

Sebastien Kanj
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya (UPC)
c. Jordi Girona, 1-3
08034, Barcelona, Spain
sebastien.kanj@upc.edu

Olga León
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya
c. Esteve Terradas, 7
08860, Castelldefels, Spain
olga.leon@upc.edu

Enrica Zola
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya
c. Jordi Girona, 1-3
08034, Barcelona, Spain
enrica.zola@upc.edu

Oriol Rosés
INCIDE Digital Data SL
Defion Security Spain
Av. Diagonal, 622, 7A
08021, Barcelona, Spain
oroses@incide.es

Josep Pegueroles
Departament d'Enginyeria Telemàtica
Universitat Politècnica de Catalunya (UPC)
c. Jordi Girona, 1-3
08034, Barcelona, Spain
josep.pegueroles@upc.edu

***Abstract*—Este es un resumen extendido de la publicación [1]. El marco MITRE ATT&CK® es un repositorio global que organiza tácticas, técnicas y procedimientos (TTPs) basados en amenazas reales, permitiendo analizar comportamientos de ataque y mejorar la atribución de incidentes. Los fraudes de Business Email Compromise (BEC), una amenaza creciente que explota el correo electrónico para obtener beneficios financieros, no suelen analizarse con MITRE debido a la falta de una matriz específica. Este estudio mapea los TTPs usados por actores BEC dentro de MITRE ATT&CK® mediante el análisis de literatura académica, informes de Cyber Threat Intelligence (CTI) y datos reales de INCIDE Digital Data S.L. Se identificaron 10 tácticas, 34 técnicas y 46 sub-técnicas, junto con 5 nuevas sub-técnicas relacionadas con manipulación de buzones y evasión de defensas. También se fusionaron las tácticas *Privilege Escalation, Lateral Movement y Credential Access*, y se excluyó *Execution* por no ser relevante. Para validar el marco, se caracterizaron dos actores reales: Cosmic Lynx, sofisticado, y Chiffon Herring, de menor complejidad. Los resultados muestran que la matriz facilita la atribución, detección y prevención de ataques BEC.**

***Index Terms*—BEC, Business Email Compromise, DFIR, MITRE, Procedures, Tactics, Techniques, TTP**

I. Introducción

El marco MITRE ATT&CK® [2] es una de las referencias globales más utilizadas para describir el comportamiento de ciberatacantes a través de Tácticas, Técnicas y Procedimientos (TTPs). Aunque se ha aplicado ampliamente en incidentes como ransomware, intrusiones en redes o ataques dirigidos, su uso para caracterizar ataques de Business Email Compromise (BEC) ha sido limitado.

Los ataques BEC son una de las amenazas económicas más significativas del panorama actual [3]. Se basan en la manipulación maliciosa de comunicaciones por correo electrónico para obtener beneficios financieros mediante ingeniería social, robo de cuentas o suplantación. En 2023, el FBI reportó más de 21.000 incidentes con pérdidas superiores a 2.9 mil millones de dólares, reflejando la gravedad y el impacto global de este tipo de fraude [4].

Aunque existen estudios sobre phishing, spoofing, ingeniería social o uso de malware en BEC, ninguno de ellos realiza una investigación sistemática que mapee las técnicas más relevantes de estos ataques en el marco MITRE ATT&CK®. Este artículo aborda ese vacío proponiendo una matriz específica de TTPs aplicables al BEC, basada en literatura académica, informes de inteligencia de amenazas (CTI) [5] [6] [7] e incidentes reales analizados por INCIDE Digital Data S.L.

Los autores identifican 10 tácticas, 34 técnicas y 46 sub-técnicas aplicables al BEC, proponiendo además 5 nuevas sub-técnicas no contempladas actualmente en MITRE. También se fusionan ciertas tácticas –Privilege Escalation, Lateral Movement y Credential Access– por su solapamiento, y se elimina la táctica de Execution, irrelevante en el contexto del BEC. Finalmente, se demuestra la utilidad de la matriz con la caracterización de dos actores reales: Cosmic Lynx (altamente sofisticado) [8] [9], [10] y Chiffon Herring (pequeño actor con métodos simples) [11]. Por temas de espacio, para un análisis más extendido se remite a [1].

II. Estado del Arte

Los estudios previos sobre BEC se han enfocado en técnicas aisladas: phishing, spoofing, ingeniería social, malware o manipulación documental [12] [13] [14] [15]. Aunque ofrecen información valiosa, tienen una limitación: no utilizan un marco estandarizado como MITRE ATT&CK®, lo que dificulta comparar metodologías entre actores y atribuir incidentes.

Otros marcos como la Cyber Kill Chain [16] o el Diamond Model [17] ayudan a analizar ataques, pero no tienen la granularidad de MITRE para caracterizar tácticas específicas. En contraste, MITRE ATT&CK® proporciona un enfoque

flexible y detallado, permitiendo clasificar comportamientos de forma modular y reutilizable.

Hasta ahora, MITRE se ha usado para ransomware, exfiltración o amenazas ICS, pero no existe una selección curada de TTPs específicos para BEC. Este estudio, que es un resumen de [1], es la primera propuesta integral en ese sentido.

III. Metodología

Para alcanzar los objetivos de esta investigación, se diseñó una metodología destinada a revisar y mapear de forma sistemática los TTPs utilizados por los Threat Actors (TAs) en fraudes BEC dentro del marco MITRE ATT&CK®. Integrando datos de diversas fuentes –literatura académica, informes CTI de la industria y casos reales de respuesta a incidentes proporcionados por INCIDE Digital Data S.L.– se buscó ofrecer un análisis completo de los TTPs específicos de BEC (Figura 1). El proceso incluyó la identificación, mapeo y estandarización de tácticas y técnicas relacionadas con BEC dentro de la matriz ATT&CK. A continuación, se describen las fases seguidas.

Análisis de la matriz MITRE ATT&CK®. Este marco reconocido globalmente cataloga TTPs observados en amenazas reales. Las tácticas representan los objetivos del atacante, mientras que las técnicas y sub-técnicas describen los métodos empleados. En esta fase se identificaron los elementos del marco más relevantes para BEC, priorizando técnicas relacionadas con el entorno del correo, sus componentes e infraestructuras que suelen explotarse en este tipo de fraude.

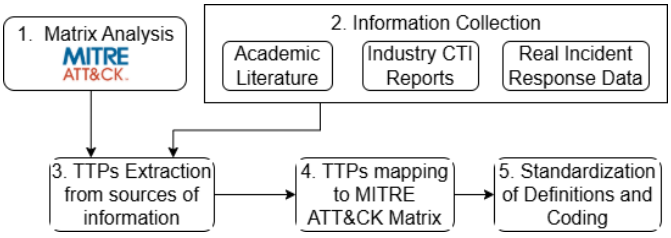


Fig. 1. Proposed methodology for BEC TTPs review [1].

Recopilación de información. Uno de los principales retos en el estudio del BEC es la falta de grandes conjuntos de datos públicos o informes detallados. A diferencia del malware o ransomware, el BEC –basado en ingeniería social– suele estar subdocumentado. Por ello, se recurrió a tres tipos de fuentes:

- Literatura académica, que aportó una base teórica sobre phishing, spoofing, manipulación de documentos y otros vectores de ataque.
- Informes CTI de la industria, que ofrecieron ejemplos recientes y tácticas usadas en campañas BEC reales.
- Datos de incidentes reales, obtenidos de casos gestionados por INCIDE, que aportaron una visión práctica de los TTPs observados durante ataques reales.

Extracción de TTPs. Los datos recopilados se analizaron, armonizando nombres y descripciones procedentes de distintas fuentes. Se identificaron tácticas basadas en los objetivos del atacante, técnicas específicas para alcanzarlas y sub-técnicas que reflejan variaciones significativas relevantes para BEC.

Mapeo al marco ATT&CK. Cada TTP identificado se asignó a la categoría correspondiente dentro de la matriz, siguiendo criterios de consistencia y funcionalidad dentro del ciclo de ataque. Se alinearon fases, se definieron objetivos, se compararon técnicas con las existentes en ATT&CK® y se evaluaron sub-técnicas para seleccionar la mejor correspondencia o proponer nuevas cuando no existía una equivalencia directa. También se eliminó redundancia y se garantizó la unicidad de cada TTP mapeado.

Estandarización final. Se asignaron identificadores ATT&CK a cada táctica, técnica y sub-técnica; se redactaron descripciones claras; se añadieron referencias correspondientes; y se elaboró una representación visual del mapeo final, proporcionando una visión global de los TTPs específicos del BEC dentro del marco ATT&CK.

IV. Resultados

En esta sección se especifican las tácticas, técnicas y sub-técnicas propuestas para BEC, resultado del análisis. Cada elemento incluye una definición concisa y sus referencias. En la Figura 2 se muestra la matriz resultante. El análisis partió de técnicas relacionadas con la recopilación de información y con infraestructuras de correo y nube, por ser fundamentales en el comportamiento de los atacantes en BEC. Se utilizaron más de 54 fuentes –24 académicas, 10 informes CTI de la industria y más de 20 casos reales de respuesta a incidentes–, lo que permitió refinar las técnicas preseleccionadas y asegurar su pertinencia. Además, se proponen cinco nuevas sub-técnicas que cubren vacíos del marco ATT&CK y se explica la fusión de tres tácticas (Privilege Escalation, Lateral Movement y Credential Access) debido a su solapamiento en el contexto BEC. La táctica *Execution* se excluyó por no ajustarse a este tipo de fraude.

Las tácticas resultantes se describen a continuación.

A. Reconnaissance (TA0043) El atacante recoge información del objetivo para preparar la intrusión. Incluye técnicas como Active Scanning (T1595), que identifica vulnerabilidades en servidores de correo; Victim Organization Information (T1591), que recopila relaciones comerciales y roles clave; y Brute Force (T1110), especialmente Credential Stuffing, basado en credenciales filtradas.

B. Resource Development (TA0042) El atacante construye su infraestructura: adquisición de dominios, servidores o botnets (T1583); compra de accesos (T1650); creación o compromiso de cuentas (T1585, T1586); desarrollo de malware (T1587); y preparación de recursos para campañas de phishing (T1608).

C. Initial Access (TA0001) Incluye explotación de servidores de correo (T1190), uso de cuentas válidas (T1078), múltiples variantes de phishing (T1566), abuso de relaciones de confianza (T1199), uso de tokens o cookies para autenticación (T1550) y ataques de fuerza bruta (T1110).

D. Persistence (TA0003) Busca mantener el acceso prolongado al buzón. Incluye manipulación de cuentas y reglas de reenvío (T1098), uso de tokens o cookies (T1550), modificaciones de autenticación (T1556) y técnicas Adversary-in-the-Middle como Reply-To Field o Mail Thread Export.

E. Defense Evasion (TA0005) El atacante evita la detección mediante suplantación (T1656), eliminación de

evidencias (T1070), ocultación con reglas de correo (T1564) y el uso de infraestructuras confiables o configuraciones SPF/DKIM/DMARC (T1553).

F. Discovery (TA0007) Busca identificar cuentas valiosas y documentos financieros mediante Account Discovery (T1087) o revisión de almacenamiento en la nube (T1619).

G. Privilege Escalation / Lateral Movement / Credential Access Estas tres tácticas están unificadas en BEC, ya que su objetivo común es acceder a cuentas con mayor autoridad. Incluyen búsqueda de credenciales expuestas (T1552) e internal spearphishing (T1534).

H. Collection (TA0009) El atacante reúne documentos financieros (T1530) y correos relevantes mediante búsquedas o reglas de reenvío (T1114).

I. Exfiltration (TA0010) No siempre esencial, pero posible. Incluye descarga directa desde servicios web (T1567), extracción desde la nube (T1530) o uso de APIs y consolas (T1048).

J. Impact (TA0040) Fase final, donde el atacante manipula datos transmitidos (T1565) o ejecuta el fraude financiero (T1657), como transferencias o pagos desviados.

V. Discusión

Siguiendo la metodología descrita, en el análisis de fraudes BEC se identificó un total de 10 tácticas, 34 técnicas y 46 sub-técnicas. Todas fueron mapeadas al marco MITRE ATT&CK®, poniendo especial atención en los comportamientos específicos de BEC. Además, se propusieron 5 nuevas sub-técnicas para cubrir vacíos del marco original.

Dos de estas nuevas sub-técnicas pertenecen a la táctica de Persistencia (TA0003) y a la técnica Adversary-in-the-Middle (T1557): Reply-To Field (T1557.XX) y Mail Thread Export (T1557.XX). Son propias de la manipulación del buzón, práctica muy común en BEC, ya que permiten interceptar y controlar las comunicaciones de correo para acceder a información sensible o inducir a las víctimas a realizar acciones fraudulentas. Para mitigar el riesgo de la manipulación del campo Reply-To, las organizaciones pueden configurar reglas de seguridad que generen alertas cuando haya discrepancias entre los campos "From" y "Reply-To", complementar con mecanismos de autenticación avanzada y formar al personal para reconocer estas señales.

Las otras tres nuevas sub-técnicas se enmarcan en la táctica de Defensa Evasiva y la técnica Subvert Trust Controls (T1553): Trustful IP Provider (T1553.XX), Set Authentication Protocols (SPF, DKIM, DMARC) (T1553.XX) y Trustful Mail Providers (T1553.XX). Todas son específicas de ataques basados en correo y buscan eludir las defensas del entorno de la víctima, asegurando que los correos maliciosos lleguen a la bandeja de entrada sin ser bloqueados. El uso de direcciones IP de confianza y la configuración maliciosa de protocolos de autenticación permiten evadir sistemas de filtrado habituales. Mitigar estas técnicas es complejo, pues se apoyan precisamente en mecanismos de seguridad legítimos diseñados para garantizar la entregabilidad y autenticidad del correo.

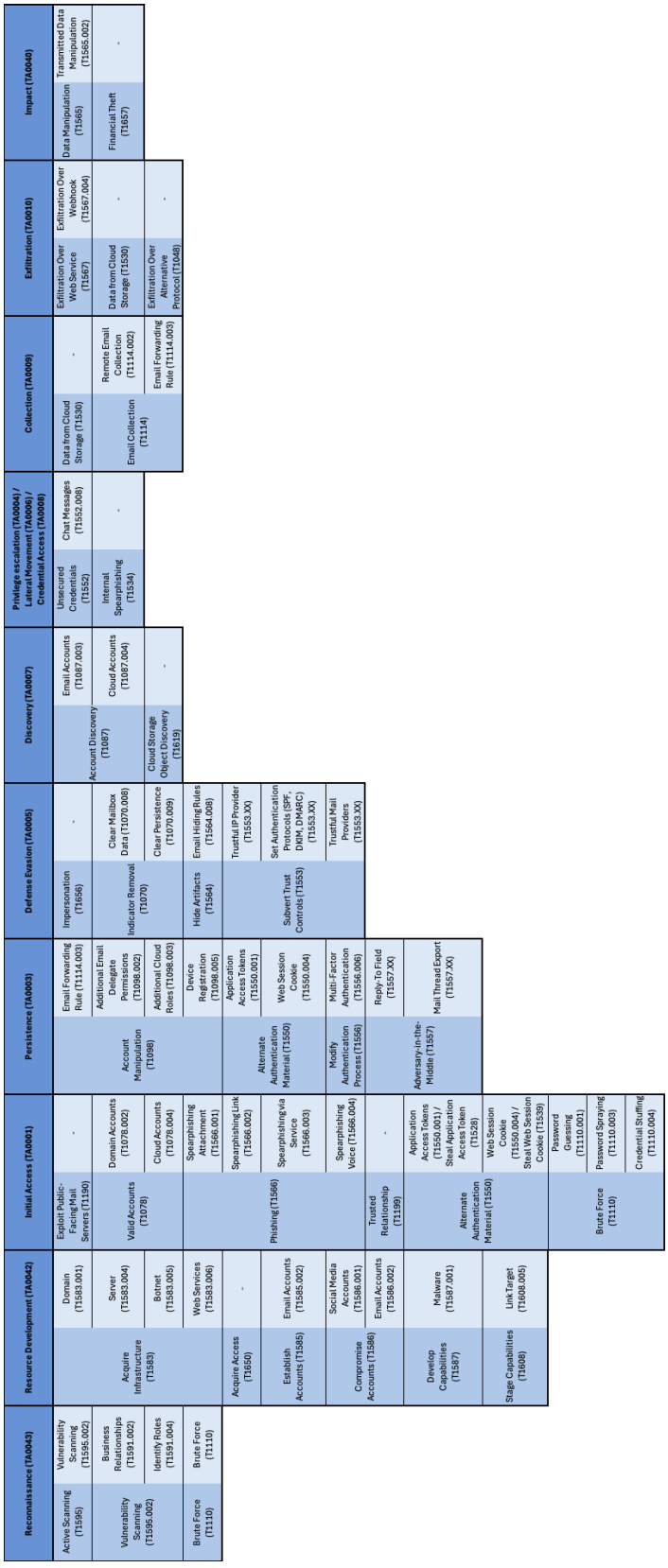


Fig. 2. BEC TTPs MITRE ATT&CK® Matrix

Asimismo, tres tácticas originales de MITRE –Privilege Escalation (TA0004), Lateral Movement (TA0006) y Credential Access (TA0008)– se fusionaron en una sola fase de ataque. La razón es que, en el contexto de BEC, comparten el mismo objetivo: obtener acceso a cuentas con mayores privilegios o capacidad de decisión financiera. Además, las técnicas y sub-técnicas empleadas en estas fases son prácticamente las mismas. Aunque podría haberse

propuesto un nuevo nombre para esta fase combinada, se decidió mantener la nomenclatura de MITRE por coherencia y claridad.

La táctica de Execution (TA0002) se omitió deliberadamente, dado su escaso peso en BEC. A diferencia de otros ciberataques (como ransomware), donde la ejecución de código malicioso es central, en BEC el foco está en la ingeniería social, la manipulación de correo y el abuso de credenciales. El éxito del ataque depende más de explotar relaciones de confianza y canales legítimos que de desplegar malware. En los pocos casos donde puede intervenir malware (por ejemplo, stealers para robo de credenciales), su ejecución se considera mejor encuadrada en tácticas como Initial Access (TA0001) o Credential Access (TA0008), al tratarse de acciones preparatorias y no del núcleo del ciclo BEC.

En algunos casos, se añadieron sub-técnicas bajo técnicas distintas a las contempladas originalmente en MITRE, sin mantener el código "padre" original. Esto se debe a que, en BEC, dichas sub-técnicas se asocian de forma más práctica a otras técnicas ya existentes. Resultaba más eficiente referenciar una sub-técnica ya definida que crear otra redundante con la misma descripción.

Finalmente, se incluyeron estudios de caso para mostrar la aplicación práctica de la matriz: Cosmic Lynx (actor sofisticado, orientado a grandes organizaciones y altos ejecutivos) [8] [9], [10] y Chiffon Herring (actor de menor escala, centrado en fraudes más simples como desvío de nóminas) [11]. En ambos casos se extrajeron sus TTPs de informes CTI y se mapearon a la matriz propuesta [1]. Estos casos ilustran cómo la matriz permite estandarizar la representación de TTPs, facilitar la atribución de incidentes, comparar actores con distintos niveles de sofisticación y orientar medidas defensivas concretas según las técnicas empleadas y los objetivos del adversario.

VI. Conclusiones y líneas futuras

En este trabajo se han identificado y categorizado las técnicas utilizadas por los Threat Actors (TAs) en fraudes de Business Email Compromise (BEC). Mediante la aplicación del marco MITRE ATT&CK®, se identificaron 12 tácticas, fusionándose Privilege Escalation, Lateral Movement y Credential Access en una sola fase por el solapamiento de objetivos y técnicas. La táctica Execution fue excluida por no ser relevante para el ciclo de ataque típico de BEC. Además, se identificaron 34 técnicas y 46 sub-técnicas, incluyendo cinco nuevas propuestas –Reply-To Field, Mail Thread Export, Trustful IP Provider, Set Authentication Protocols (SPF, DKIM, DMARC) y Trustful Mail Providers– que cubren vacíos existentes, especialmente en Adversary-in-the-Middle y Defense Evasion.

La propuesta permite a investigadores de CTI caracterizar de forma precisa a los atacantes BEC, facilitando la atribución de incidentes, mejorando las capacidades de investigación y apoyando estrategias de mitigación más efectivas.

Los estudios de caso de Cosmic Lynx y Chiffon Herring validan la utilidad del marco. La aplicación de la matriz de TTPs demuestra su eficacia tanto para un TA sofisticado como Cosmic Lynx, que opera campañas organizadas dirigidas a víctimas de alto valor, como para un actor más pequeño como Chiffon Herring, cuyos métodos más simples igualmente manipulan la confianza y evaden la detección.

Como trabajo futuro se propone la caracterización extensiva y exhaustiva de TAs para reforzar estrategias de ciberseguridad, que incluye actualización con nuevas técnicas y tácticas conforme aparezcan y garantizar defensas adaptativas frente a la evolución del fraude BEC. Otra vía importante será utilizar la matriz para analizar el modus operandi de TAs conocidos, ayudando a correlar grupos que operen bajo distintos nombres. Además, dada la limitada disponibilidad de datos abiertos sobre incidentes BEC, futuros estudios evaluarán el marco en bases de datos de distintas organizaciones, ampliando su validación y garantizando su aplicabilidad en entornos diversos.

Agradecimientos

Esta publicación cuenta con el apoyo del programa de doctorados industriales de AGAUR, Generalitat de Catalunya (2021 DI 092), de la Cátedra de Ciberseguridad CARISMATICA financiada por la UE e INCIBE y de la empresa INCIDE Digital Data S.L., que ha proporcionado acceso a datos confidenciales para realizar el estudio.

References

[1] Kanj, Sebastien and et al.. *A Review of Tactics, Techniques, and Procedures (TTPs) of MITRE Framework for Business Email Compromise (BEC) Attacks*. (2024).
[2] The MITRE Corporation. *MITRE A*. (2024). https://attack.mitre.org
[3] Steve Mansfield-Devine. *The imitation game: how business email compromise scams are robbing organizations*. (2016).
[4] Federal Bureau of Investigation (FBI). *2023 Internet Crime Report*. https://www.ic3.gov/Media/PDF/AnnualReport/2022_IC3Report.pdf
[5] Raj, Arpit and Narayan, Vedant and Muskan, Vivek and Sani, Abhilash and Sharma, Pankaj and Sarma, SS. *Modern ransomware: Evolution, methodology, attack model, prevention and mitigation using multi-tiered approach*. (2024).
[6] Mundt, Michael and Baier, Harald. *Towards mitigation of data exfiltration techniques using the MITRE ATT&CK framework*. (2021).
[7] Izzuddin, Ahmad Balya and Lim, Charles. *Mapping threats in smart grid system using the mitre att&ck ics framework*. (2022).
[8] Agari Data, Inc.. *Cosmic Lynx The Rise of Russian BEC*. https://static.fortra.com/agari/pdfs/guide/ag-acid-cosmic-lynx-gd.pdf
[9] Verdict Editorial Team. *Cosmic Lynx, Russian BEC Group*. https://www.verdict.co.uk/cosmic-lynx-russian-bec/
[10] Threatpost Editorial Team. *Russian BEC Gang Cosmic Lynx Uncovered*. https://threatpost.com/russian-bec-gang-cosmic-lynx-uncovered/157166/
[11] Abnormal Security Intelligence. *BEC Group Targets Payroll Diversion Attacks*. (2023). https://intelligence.abnormalsecurity.com/blog/bec-group-targets-teachers-payroll-diversion-attacks
[12] Papathanasiou, Anastasios and Liontos, George and Liagkou, Vasiliki and Glavas, Euripidis. *Business Email Compromise (BEC) Attacks: Threats, Vulnerabilities and Countermeasures—A Perspective on the Greek Landscape*. (2023).
[13] T N, Nisha and Bakari, Digant and Shukla, Charmi. *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*. (2021).
[14] Ogwo-Ude, Okechukwu. *Business Email Compromise Challenges to Medium and Large-Scale Firms in USA: An Analysis*. (2023).
[15] Al-Musib, N.S. and Al-Serhani, F.M. and Humayun, M. and Jhanjhi, N.Z.. *Business email compromise (BEC) attacks*. (2023).
[16] Yadav, Tarun and Rao, Arvind Mallari. *Technical aspects of cyber kill chain*. (2015).
[17] Caltagirone, Sergio and Pendergast, Andrew and Betz, Christopher. *The diamond model of intrusion analysis*. (2013).

Topological Safeguard for Evasion Attack based on the Interpretability of Artificial Neural Network Behavior (Extended Abstract)

Xabier Echeberria-Barrio
Vicomtech Foundation, Basque Research and Technology Alliance (BRTA)
Donostia-San Sebastian, 20009, Spain
xetxeberria@vicomtech.org

***Resumen*—This is an extended abstract of the publication [1]. Deep Learning technology is gaining importance and influence across various fields, including critical areas like healthcare and autonomous vehicles, where decision-making errors can have severe consequences. Concurrently, its widespread adoption has raised new cybersecurity threats, specifically the evasion attack, where an attacker adds specially designed, often imperceptible noise to an input sample, modifying the model's prediction. While researchers have developed various defenses, no single defense is perfect against all known evasion algorithms (such as FGSM, BIM, and PGD). This work introduces a novel evasion attack detector focusing on two key aspects: the information contained in neuron activations and the topology of the targeted Deep Learning model. The detector is motivated by findings that the model's topological structure contains essential information regarding whether an evasion attack is occurring, as these attacks widely alter the previously traversed prediction pathway. A critical finding prior to this work was that no literature analyzed activations while simultaneously maintaining their topological relations for evasion attack detection.**

***Index Terms*—Artificial neural network, Interpretability, Cybersecurity, Adversarial learning, Evasion attack**

I. Introduction

Deep Learning (DL) technology is increasingly deployed across various fields where decision-making errors can have severe consequences. While DL offers significant advances, its widespread adoption has concurrently raised new cybersecurity threats and vulnerabilities.

A major concern is the evasion attack [2], a well-known vulnerability where an attacker adds a specifically designed, often imperceptible noise to an input sample, modifying the model's prediction. Since the introduction of the L-BFG algorithm, researchers have been actively developing defenses, though no single defense is perfect against all known evasion algorithms (such as FGSM, BIM, PGD, and Carlini–Wagner).

This work introduces a novel evasion attack detector focusing on two key aspects: the information contained in neuron activations and the topology of the targeted Deep Learning model. The detector is motivated by findings that the model's topological structure contains essential information regarding whether an evasion attack is occurring. Disturbances caused by evasion attacks widely alter the previously traversed prediction pathway within the model.

The proposed detector uses Graph Convolutional Neural Network (GCN) technology to understand and leverage the target model's topology. To manage the computational complexity associated with modern large models, this approach focuses specifically on analyzing the classifier block (ϕ_C) of the targeted deep learning model, rather than the entire network. The goal is to improve detection rates found in the literature and offer a new methodology in this field.

II. Literature Review

Defenses against evasion attacks often fall into two categories: hardening the model (e.g., adversarial training [6] or dimensional reduction methods like PCA/Autoencoders [5]), or using an external detector (an auxiliary model). External detectors typically use a second classifier to distinguish malicious inputs. While successful, these detectors can sometimes be evaded simultaneously with the main classifier. Similar detector approaches analyze activation values or introduce new branches in the main network [3], [4]. Crucially, the sources emphasize that, prior to this work, no literature was found that analyzed activations while simultaneously maintaining their topological relations for evasion attack detection.

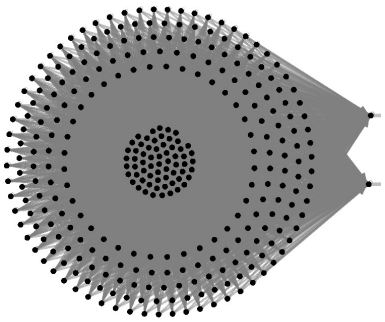


Figura 1: A classifier's behavior graph example, where the central cluster corresponds to the input layer of the classifier, the surrounding nodes are the neurons of the hidden layer, and the last right nodes correspond to the output layer. Notice that the neurons without any connection do not appear.

III. Method

The target deep learning model (ϕ) is conceptualized as having a feature-extractor block (ϕ_F) and a dense classifier block (ϕ_C). The detector operates on ϕ_C, analyzing features computed from its activations when an input image x_i is processed.

III-A. The Behavior Graph

When an input x_i is fed into the classifier ϕ_C, it generates a set of activations. These activations and the neurons of the classifier define a graph representation called the Behavior Graph ($G^{x_i}_{\phi_C}$).

The Behavior Graph is a Weighted Digraph:

- Nodes (N_{ϕ_C}): The neurons of the classifier ϕ_C.
- Weighted Edges ($A^{x_i}_{\phi_C}$): The activations that connect two neurons, defining the influence between them for input x_i.

This graph structure captures the decision-making process, showing how neurons are activating and their relative influence, thus reflecting the targeted model's behavior.

III-B. Classifier Node Attributes

To enrich the topological data for the GCN detector, several neuron attributes are computed based on the behavior graph. These attributes are normalized by layer to ensure comparability across layers with different activation functions.

III-B1. Impact ($I^x_{\phi_C}$): This modified attribute compares a neuron's normalized output activation with the sum of its normalized input activations. It reveals how much a neuron modifies the received values (positively or negatively) during the prediction of x_i. It is a measure of the neuron's influence on the flow of information.

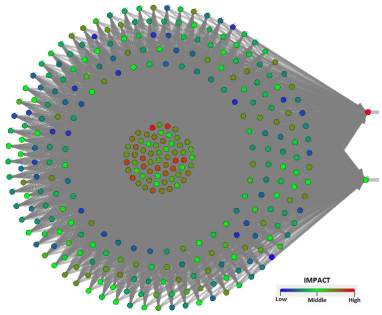


Figura 2: Classifier Behavior Graph example, where the nodes are coloring according to their impact attribute.

III-B2. Influence ($\Xi^x_{\phi_C}$): This attribute highlights the neurons per layer that participate in the prediction with the highest activation values. It is calculated by normalizing activations by layer and then setting a binary value (1 or 0) based on a percentile parameter (p), identifying the most influential neurons in the classification.

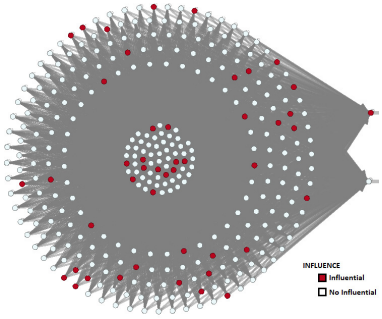


Figura 3: Classifier Behavior Graph example, where the nodes are coloring according to their influence attribute.

III-B3. Input Proportion ($P^x_{\phi_C}$): Proposed specifically in this work, this attribute computes the percentage of non-null input values a neuron receives relative to all possible inputs. This is particularly useful when the classifier uses activation functions (like ReLU) that map values to zero. Since the classifier in the experiment is a fully connected model, all neurons in a specific layer receive the same input proportion value.

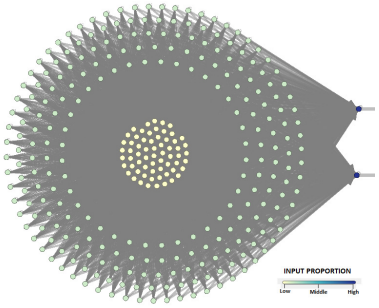
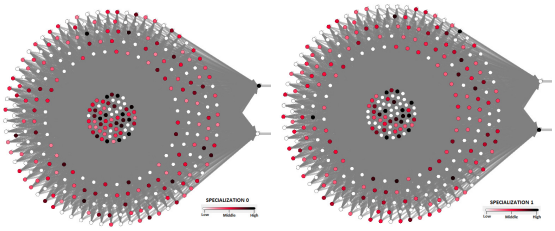


Figura 4: Classifier Behavior Graph example, where the nodes are coloring according to their input proportion attribute.

III-B4. Specialization ($Z^{x,\mathfrak{c}_i,k}_{\phi_C}$): This attribute quantifies a neuron's frequency of participation in predicting a specific class c. It measures how often a neuron's activation is among the $k-$top values when images of class c are processed.



(a) $0-$specialization (b) $1-$specialization

Figura 5: Classifier Behavior Graph example, where the nodes are coloring according to their specialization attributes.

III-C. Data Preprocessing and Detector Architecture

The preprocessing phase takes a set of adversarial and original images X and the target model ϕ to generate the enriched dataset for the detector. For each image x in X, the process generates:

- The Adjacency Matrix ($Ad^x_{\phi_C}$), representing the topology of ϕ_C.
- The set of all computed node attributes

$$\Delta^{x,p,k,t}_{\phi_C} = (I^x_{\phi_C}, \Xi^x_{\phi_C}, P^x_{\phi_C}, Z^{x,c_i,k}_{\phi_C}) \quad (1)$$

- A binary label

$$\lambda^x_{\phi_C} = \begin{cases} 1, & \text{if is an adversarial example} \\ 0, & \text{if is not an adversarial example.} \end{cases}$$

, indicating whether the image is adversarial (1) or not (0).

-

GCN Technology: Graph Neural Network (GNN) technology, specifically Graph Convolutional Networks (GCNs), was

chosen for the detector because of its ability to collect and process topological information through the graph structure. The selected architecture uses a sequence of GCN layers (with hyperbolic tangent activation) followed by one-dimensional convolution layers and a final classifier block.

IV. Experimental Setup and Scenario

IV-A. Scenario Details

The experiment uses a target deep learning model comprising a VGG16 network as the feature-extractor (ϕ_F) and a dense neural network as the classifier (ϕ_C). The classifier in this specific scenario uses the sigmoid activation function. The scenario employs the breast cancer dataset, which contains two image classes (non-cancer and cancer).

Adversarial examples were generated using three algorithms, all with the same perturbation budget ($\epsilon = 5/255$):

- Fast Gradient Sign Method (FGSM) (1 step).
- Basic Iterative Method (BIM) (10 steps).
- Projected Gradient Descent Method (PGD) (40 steps).

Four datasets were created: FGSM, BIM, PGD, and a combined Total dataset (53,908 total images, 26,954 adversarial). The parameter p for Influence was fixed at 0.5, and k for Specialization was fixed at 10.

IV-B. Attribute Analysis

An analysis of the generated node attributes showed that they were generally lowly correlated with each other and, importantly, none was directly correlated with the adversarial label. This confirms that a complex model, such as the GCN detector, is necessary to associate the attributes with the label.

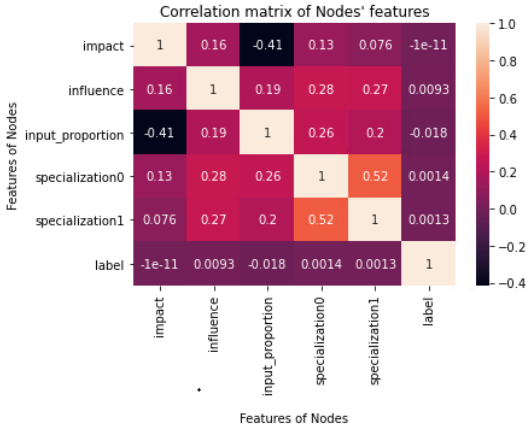


Figura 6: Correlation between the node features in the generated dataset

V. Results and Discussion

Four detectors were trained: one for each specific attack (FGSM, BIM, PGD) and one trained on the combined data (Total detector).

V-A. Detection Performance

The detectors achieved promising results in identifying adversarial examples (see Table I):

- BIM Detector: Highest accuracy at 0.969 (±0.0075).
- PGD Detector: High accuracy at 0.9573 (±0.0037).
- Total Detector (Generalized): Accuracy of 0.9219 (±0.0057), demonstrating effectiveness regardless of the attack type.
- FGSM Detector: Lowest accuracy at 0.8832 (±0.0023).

When the target model is defended by the proposed detector, the original success rate of the evasion algorithms drops significantly. For example, the success rate of the BIM attack fell from 0.9378 (original model) to 0.0290 (defended model).

Tabla I: The results obtained from the developed detectors

Metrics	FGSM detector	BIM detector	PGD detector	Total detector
Loss	$0{,}3623 \pm 0{,}0052$	$0{,}1394 \pm 0{,}0204$	$0{,}1852 \pm 0{,}0060$	$0{,}2566 \pm 0{,}0057$
Accuracy	$0{,}8832 \pm 0{,}0023$	$0{,}969 \pm 0{,}0075$	$0{,}9573 \pm 0{,}0037$	$0{,}9219 \pm 0{,}0057$

V-B. Validation of Topological Information

To validate the core hypothesis regarding the importance of topology, the GCN detector was compared against a basic MLP detector lacking the ability to capture topological information (see Table II).

The GCN detector performed considerably better in all cases:

- Total Detector (GCN): 0.9219 accuracy.
- MLP Detector: 0.8465 accuracy.

This outcome supports the investigation's original idea that topological information is crucial for understanding and detecting anomalies in deep learning models.

Tabla II: The detectors were trained with each dataset obtaining different results. The Total column shows the results obtained by training the detector with the total dataset.

Detectors	FGSD attack	BIM attack	PGD attack	Total
Total detector (GCN)	0,8832	0,969	0,9573	0,9219
MLP detector	0,8140	0,9170	0,8930	0,8465

V-C. Cross-Analysis and Transferability

Evaluation across different datasets showed the detector's ability to transfer detection patterns between adversarial types. Detectors trained on BIM or PGD attacks performed well when tested against other attack types. For instance, the BIM detector achieved high accuracy against the PGD test data (0.9637), and the PGD detector performed well against BIM test data (0.9501) [see Table III]. This cross-analysis suggests that training a single, generalized detector (like the Total detector) may not be strictly necessary, as detectors trained on BIM or PGD attacks can cover those threats effectively.

Tabla III: The results obtained from the combination of the trained detectors and the different datasets.

	FGSM test	BIM test	PGD test
FGSM detector	0,8832	0,7097	0,6923
BIM detector	0,8704	0,969	0,9637
PGD detector	0,8645	0,9501	0,9573
Total detector	0,8435	0,9614	0,9524

V-D. Attribute Contribution Analysis

Evaluating the contribution of each defined attribute (Impact, Influence, Input Proportion, Specialization) showed varying levels of importance (see Table IV).

- Impact and Influence attributes provided the most information for detection.
- Impact was most informative for BIM and PGD detection.
- Influence contained the most relevant information for the FGSM threat.
- Input Proportion and Specialization performed worse when used alone. The lower utility of Input Proportion in this scenario is likely due to the use of the sigmoid activation function in the classifier, which rarely returns null values, thus reducing the information this attribute can provide.

Tabla IV: The results are obtained in the total dataset from the attribute detectors. The first column shows the results of the detectors in the total dataset. The rest columns detail the accuracy of each detector in the different attacks.

Detectors	(Accuracy ± std, Loss ± std)	FGSD attack	BIM attack	PGD attack
Impact detector	(0,9047 ± 0,0081, 0,3104 ± 0,0239)	0,8053	0,9560	0,9502
Influence detector	(0,9131 ± 0,0075, 0,2865 ± 0,0215)	0,8493	0,9301	0,9276
Proportion detector	(0,8531 ± 0,0060, 0,4132 ± 0,0133)	0,8148	0,8664	0,8636
Specialization detector	(0,8598 ± 0,0063, 0,4159 ± 0,0134)	0,8182	0,8891	0,8714
Total detector	(0,9219 ± 0,0057, 0,2566 ± 0,0057)	0,8435	0,9614	0,9524

V-E. Ablation Analysis

A subsequent ablation analysis confirmed that all attributes provided extra, complementary information to the detection process, as the Total detector achieved the best generalized results (see Table V).

Tabla V: The results are obtained in the total dataset from the non-attribute detectors. The first column shows the results of the detectors in the total dataset. The rest columns detail the accuracy of each detector in the different attacks.

Detectors	(Accuracy ± std, Loss ± std)	FGSD attack	BIM attack	PGD attack
Non-impact detector	(0,9101 ± 0,0077, 0,2978 ± 0,0109)	0,8437	0,9453	0,9285
Non-influence detector	(0,9125 ± 0,0031, 0,2794 ± 0,0021)	0,8345	0,9560	0,9394
Non-proportion detector	(0,9057 ± 0,0055, 0,3056 ± 0,0127)	0,8472	0,9490	0,9425
Non-specialization detector	(0,9192 ± 0,0013, 0,2659 ± 0,0056)	0,8443	0,9613	0,9520
Total detector	(0,9219 ± 0,0057, 0,2566 ± 0,0057)	0,8435	0,9614	0,9524

V-F. Comparison with Literature

Auxiliary Model Comparison: The proposed detector achieved superior detection rates compared to well-known auxiliary detectors (LID, NSS, and KD+BU) found in the literature for this specific scenario [97, 98, Table 12]. For instance, against the BIM attack, the proposed detector achieved 0.969 accuracy, significantly improving upon the NSS detector (0.8028). Theoretical Comparison to Similar Methods: A theoretical comparison was made with the work of Pawlicki et al., which also analyzes activation values. Their approach treats activations as a single, long 1D vector, ignoring topology and considering all activations of the targeted model. This novel detector improves on that by focusing only on the classifier block for scalability and crucially, by integrating the target model's topology as an essential feature for detection.

VI. Conclusion and Future Work

This work successfully developed a novel evasion attack detector that achieves outstanding results by incorporating and leveraging the topological information of the targeted neural network's classifier block, utilizing Graph Convolutional Networks. The introduction of four novel or adapted node attributes—Impact, Influence, Input Proportion, and Specialization—enriches the dataset, providing the GCN with detailed information about the model's behavior. The analysis supports the importance of topology in understanding and detecting model perturbations.

Future research aims to refine the system by:

1. Testing different GCN architectures and optimizing hyperparameters.
2. Using interpretability techniques on the trained detector to identify vulnerable neurons, which could lead to the generation of local defenses.
3. Extending the analysis to the highly complex feature extractor block (ϕ_F) of the model. This extension would require implementing simplification techniques, such as aggregations or graph splitting, to manage the immense number of parameters involved.

Aknowledgements

This work was partially supported by the European Commission under the Horizon Europe Programme as part of the FALCON and TESTUDO projects (Grant Agreements No. 101121281 and No. 101121258)

Referencias

[1] Echeberria-Barrio, X., Gil-Lerchundi, A., Mendialdua, I., Orduna-Urrutia, R.: "Topological safeguard for evasion attack interpreting the neural networks' behavior", en *Pattern Recognition*, 147, 110130, 2024.

[2] Jiang, W., Li, H., Liu, S., Luo, X., Lu, R.: "Poisoning and evasion attacks against deep learning algorithms in autonomous vehicles". *IEEE transactions on vehicular technology*, 69(4), 4439-4449, 2020.

[3] Pawlicki, M., Choraś, M., Kozik, R.: "Defending network intrusion detection systems against adversarial evasion attacks". *Future Generation Computer Systems*, 110, 148-154, 2020.

[4] Metzen, J. H., Genewein, T., Fischer, V., Bischoff, B.: .on detecting adversarial perturbations". *arXiv preprint arXiv:1702.04267*, 2017.

[5] Sahay, Rajeev and Mahfuz, Rehana and El Gamal, Aly: Çombatting adversarial attacks through denoising and dimensionality reduction: A cascaded autoencoder approach". *53rd Annual conference on information sciences and systems (CISS)*, 2(3), 1–6, 2019.

[6] Yu, X., Smedemark-Margulies, N., Aeron, S., Koike-Akino, T., Moulin, P., Brand, M., ... Wang, Y.: İmproving adversarial robustness by learning shared information". *Pattern Recognition*, 134, 109054, 2023.

Is Hardware Wallet Fingerprinting Even Possible?

Domènec Madrid Hernández, Jordi Herrera-Joancomartí
Departament d'Enginyeria de la Informació i les Comunicacions
Universitat Autònoma de Barcelona (UAB)
Domènec.Madrid@uab.cat Jordi.Herrera@uab.cat

***Abstract*—Cryptocurrency wallets are software programs or hardware devices that help users manage their cryptocurrencies by storing private keys and creating and signing the transactions that enable payments. In this context, wallet fingerprinting is the technique used to identify which type of wallet created a particular transaction. Information contained in a transaction—beyond the details determining the source and destination of funds—may provide subtle hints that can be used to determine which specific software wallet or hardware device generated the transaction, representing a privacy weakness. This paper analyzes the specific problem of hardware wallet fingerprinting in Bitcoin, a subdomain of wallet fingerprinting, and tests whether hardware wallets can be fingerprinted or not. Results show that, although the observable information hardware wallets introduce into a transaction is small, there still exists data that could be used to weaken users' privacy.**

***Index Terms*—Bitcoin, Cryptocurrency Wallets, Hardware Wallets, Wallet Fingerprinting, Digital Signatures**

I. Introduction

A Bitcoin wallet is a key component in the Bitcoin system that enables the execution of payments by performing three essential operations: the creation, signing, and broadcasting of transactions.

During transaction creation, the wallet is in charge of specifying, among other information, the unspent transaction outputs (UTXOs) to be spent, the recipient addresses, and the amount transferred. The signing phase authorizes this transaction by generating digital signatures using the private keys that control the selected UTXOs. Finally, the broadcasting phase disseminates the signed transaction to the peer–to–peer network, allowing nodes and miners to verify and confirm it on the blockchain.

These three operations can be performed on a single device or distributed across multiple devices. In software oriented wallets (e.g., desktop or mobile applications), all operations typically occur within the same environment. In contrast, hardware wallets and other security-oriented setups isolate these processes to minimize exposure of sensitive data. A common configuration involves constructing an unsigned transaction on a network-connected device, transferring it to an isolated signing device, such as a hardware wallet (HW), for authorization, and subsequently returning the signed transaction to the online environment for broadcasting.

From a security perspective, the signing process constitutes the most sensitive stage, as it directly involves the use of private keys. Any compromise during signing can result in the leakage or unauthorized use of these keys, leading to irreversible loss of funds. Consequently, the design of secure wallets focuses heavily on isolating the signing environment and ensuring that private keys remain inaccessible to distrusted systems or network interfaces. The most common path to key isolation is the use of HWs where secure element enclaves are used to ensure that private keys do not exit the device.

From a privacy perspective, all three processes—creation, signing, and broadcasting—are potentially relevant, as each can leak information that may assist in identifying the entity initiating a transaction. This information includes network–level, transaction–level, and signature–level data.

If an adversary can observe the broadcasting process, network–layer information is leaked, and they may correlate the broadcast transaction with the originating IP address or network route. This correlation can reveal geographic or identity–related information about the sender.

The transaction creation process may introduce observable implementation–dependent patterns. For instance, wallet software may differ in input and output ordering, change output positioning or fee estimation algorithms. If these processes are not strictly deterministic or allow discretionary choices, the resulting transactions may encode subtle but consistent patterns that can serve as fingerprints of specific wallet implementations.

Finally, regarding the signing process, although digital signatures are mathematically linked to private keys, the manner in which signatures are generated and encoded may also reveal information about the signing device. Variations in signature canonization or nonce generation may reveal additional distinguishing features.

The purpose of the paper is to demonstrate that hardware wallet fingerprinting, although limited, is not entirely infeasible and to identify those traits that could weaken Bitcoin users privacy.

The rest of the paper is organized as follows. In Section II, we provide the state of the art of the topic, that appears to be scarce. Section III reviews the digital signature schemes used in Bitcoin transactions, and introduces the concept of hardware wallet fingerprinting. In Section V, we describe the experimental setup, including the custom `mainlabnet` network designed to enable realistic testing with hardware wallets. Section VI presents the results of our fingerprint analysis. Finally, Section VII summarizes our findings and outlines future research directions.

II. State of the art

Software wallets (SW) are the primary actors in the transaction creation process, which means their traces are broader and more diverse than those of hardware wallets, making them more susceptible to fingerprinting.

Nevertheless, the field of software wallet fingerprinting remains underexplored in both academic literature and practical implementations, while even less attention has been devoted to hardware wallet fingerprinting.

A notable contribution to this area is the work of Ishaana Misra [1], who investigates this field exclusively at the software level. Her research demonstrates the feasibility of associating transactions with specific wallets by broadcasting transactions — whose origin wallet is known — to the blockchain and analyzing their observable traits, thereby allowing further investigation about the transaction construction strategies employed by each wallet.

Although no existing work explicitly investigates fingerprinting techniques for Bitcoin hardware wallets, there is research that explores potential vulnerabilities in these devices. M. Sorf et al. [2] present a test-bench designed to evaluate hardware wallet behavior using mechanisms such as servos, OCR, and audio capture. This setup enables automated interaction with the physical device, allowing large-scale testing. While the study does not directly address fingerprinting, it highlights that HWs typically perform a deterministic digital signature operation. This implies that, under normal conditions, the same operation over different HWs should remain identical.

Consequently, it is often assumed that fingerprinting hardware wallets is infeasible due to the lack of behavioral diversity.

However, this assumption may overlook subtle signature–level differences in HW operations. For instance, variations in signature schemes — such as support for different SIGHASH types, the use of anti–exfiltration protocols, or low–r grinding — may still leave traces that can be analyzed for fingerprinting purposes, even if they represent residual behavior.

III. Background

A. Digital signatures in a Bitcoin transaction

To authorize a digital payment in Bitcoin, a user must spend one or more of their UTXOs, each of them referenced by a transaction input individually signed to prove ownership of the corresponding private key. Consequently, a transaction that spends multiple UTXOs will contain the same number of signatures.

As Bitcoin evolved, signature placements and its schemes have undergone significant changes. In *legacy* transactions, signatures are embedded within the **scriptSig** field of each input. With the introduction of *Segregated Witness* [5], the signature data was moved to the **witness** field, segregating it from the main transaction structure. Regarding the signature schemes used, while *legacy* and *SegWit_v0* (SegWit) inputs rely on ECDSA signatures, *SegWit_v1* (Taproot) [6] adopts Schnorr.

In this paper, we focus exclusively on the signature process performed by hardware wallets. The generation of a valid signed transaction with these devices is relatively straightforward. The software wallet constructs a Partially Signed Bitcoin Transaction (PSBT) [7], specifies the transaction parameters, and transmits it to the hardware wallet through a communication channel. Upon receiving the PSBT, the HW analyzes the locking script associated with each input and performs the appropriate signature operation.

Subsequently, what is yielded to the SW is only the signature, which, regardless of the scheme used, can be abstracted as a pair of values: a point component and a scalar value, (r, s). In ECDSA, the point component r corresponds to the x–coordinate, R_x, of a nonce–derived elliptic curve point $R = (R_x, R_y)$, while in Schnorr, the first component is the full point itself R.

Although a direct examination of these values does not reveal sensitive information, certain traits of their generation and further presentation may still be susceptible to fingerprinting.

B. ECDSA Signatures

Historically, the standard signature scheme used in Bitcoin has been ECDSA, whose security relies on the hardness of the Discrete Logarithm Problem over elliptic curves.

To construct a signature with ECDSA it is mandatory first to define the domain parameters, which include the curve $E/\mathbb{Z}_p$, two coefficients a and b, and a generator point G that generates a cyclic group of order n. With these values, the ECDSA pair key can be generated.

Pair key generation

1) Choose a random integer $d \in_R (1, n-1)$
2) Compute $P = dG$

Having d as the private key of the user and P as the point that accounts for the public key.

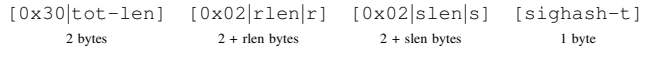
Signing The signature is generated with the user's private key.

1) Select a random integer $k \in_R (1, n-1)$.
2) Compute $R = kG = (R_x, R_y)$.
3) Compute $r = R_x \mod n$. If $r = 0$, go to Step 1.
4) Compute $e = H(m)$.
5) Compute $s = k^{-1}(e + dr) \mod n$. If $s = 0$, go to Step 1.
6) The signature of the message m is (r, s)

Verification To verify a signature (r, s) for a message m, the verifier must have the domain parameters used to sign it and the public key of the signer.

1) Verify that r and s are integers between $[1, n-1]$.
2) Compute $e = H(m)$
3) Compute $w = s^{-1} \mod n$
4) Compute $u_1 = ew \mod n$ and $u_2 = rw \mod n$
5) Compute $R = u_1 G + u_2 P$.
6) If $R = O$, then reject the signature. Otherwise, compute $v = int(x_1) \mod n$.
7) Accept the signature if and only if $v = r$.

1) ECDSA encoding: ECDSA uses Distinguished Encoding Rules (DER) [8] to correctly interpret signature values. The general structure is as follows:



Both, r and s values of the signature are 256 bits. In that type of encoding, if those values begin with a byte of `0x80` or higher, they must be preceded by a byte of `0x00`. This requirement stems from the fact that ASN.1 DER represents integers as signed values in two's complement. Without the leading zero, any value whose most significant bit is set would be interpreted as negative. Consequently, the encoded length

of each integer may vary in 1 byte depending on whether this information is needed.

Historically in Bitcoin, the s field could take two possible values in the curve, leading to a problem with transaction malleability in which there could be multiple valid encodings of the same signature.

To address these issues, BIP–62 [9] proposed accepting only S values within the low range of the curve, mitigating transaction malleability, and reducing the maximum signature size from 73 to 72 bytes.

C. Schnorr signatures

This signature scheme, now adopted in Bitcoin, was proposed decades ago, but remained under patent protection for more than 20 years, delaying its adoption despite its efficiency advantages over ECDSA. The main benefit relies on its mathematical simplicity. Schnorr is based directly on the discrete logarithm problem on elliptic curves, something that ECDSA introduces with a more complex and non–linear structure. The fact that Schnorr enables signature aggregation allows multiple signatures to be combined into one, significantly improving efficiency by reducing data size and enabling parallel verification.

Pair key generation The key pair is generated in the same manner as described in Section III-B.

1) Choose a random integer $d \in_R (1, n-1)$
2) Compute $P = dG$

Having d as the private key of the user and P as the point that works as the public key.

Signing As before, the process[1] involves the private key d and the message m.

1) In case y coordinate of P is odd $\Rightarrow d' = n - d$
2) Select a random integer $k \in_R (1, n-1)$.
3) Compute $R = kG = (R_x, R_y)$.
4) In case y the coordinate of R is odd $\Rightarrow k' = n - k$. Then, return to Point 3.
5) Compute the challenge $e = int(h(R_x||P_x||m)) \mod n$
6) Compute $s = (k + ed) \mod n$
7) The signature is (R, s)

Verification As in ECDSA, the verifier needs some specific public values to perform the validation.

1) Obtain the public key of the signer $P = dG$, the message m, and the signature (R, s).
2) Compute the challenge $e = int(h(R_x||P_x||m)) \mod n$.
3) Compute $R' = sG - eP$.
4) If the y coordinate of R' is odd, $\Rightarrow$ reject the signature.
5) Accept the signature if and only if $R'_x = R_x$.

1) Schnorr signature encoding: Notice that in Schnorr, the signer sends R and s, where R is a curve point. To compress the R value, Schnorr signature encodes R only with its x–coordinate and implicitly determines the y–coordinate. With that approach, the signature size is fixed at 64 bytes: 32 bytes for the x–coordinate of R and 32 bytes for s. Unlike ECDSA, which requires DER encoding and produces variable-length signatures, Schnorr uses a fixed binary format. This not only simplifies parsing and verification, but also ensures a consistent signature size, eliminates malleability issues, and omits additional operations such as grinding.

[1]Steps 1 and 4 of the protocol are not standard for Schnorr signatures, but reduce the public key size while maintaining a sufficiently large key space.

IV. Hardware wallet fingerprinting

As seen so far, HWs are in charge of including digital signatures in transactions, so to infer the HW that has signed a transaction, we have to fingerprint it based solely on the information included in the signature.

To perform effective fingerprinting of hardware wallets, it is essential to identify variations in signature **structure** or **value**. Although certain characteristics, such as the encoding size, can be easily observed, other features may not be practical for fingerprinting, as they are indistinguishable between signatures generated under normal conditions.

It is important to note that, although some of the following traits can separate HWs into distinct groups based on their firmware implementation, they do not ensure that a given signature was produced by one of these devices.

A. *Value*

Ideally, the values within these signatures should appear completely arbitrary, as they rely on operations involving a nonce k chosen at random. However, reusing this nonce across signatures with the same private key introduces a critical vulnerability [10], potentially leading to full private key recovery. To mitigate such potential vulnerability, RFC 6979 [12] defines a deterministic method for generating a pseudo–random nonce based on the private key and the message being signed. This ensures that signing the same message twice with the same key produces an identical signature.

Although detecting whether a HW uses deterministic nonces would be a desirable fingerprinting approach, it is practically infeasible. The nonce generation process is internal, and distinguishing deterministic from non–deterministic behavior from the resulting signature alone is not possible.

Therefore, from the point of view of an external observer, no additional fingerprints can be inferred from the signature value.

B. *Structure*

As discussed in Section III, the only structural difference between both signature schemes lies in their encoding: Schnorr signatures have a fixed size of 64 byte, whereas ECDSA signature normally varies between 70 and 73 bytes depending on their r and s values.

Although Schnorr signatures have an immutable length, as mentioned in Section III-B1, ECDSA exhibit variability due to their DER encoding. This variability opens the door to size optimizations. By consensus, the s component is normalized to fall within the lower half of the curve to prevent malleability issues; however, applying low–r grinding remains an optional feature that wallets may implement to further reduce signature size.

Similarly, just as wallets can be distinguished by whether they apply low-r grinding, the SIGHASH byte may also serve as a fingerprinting indicator. Differences in the chosen hash

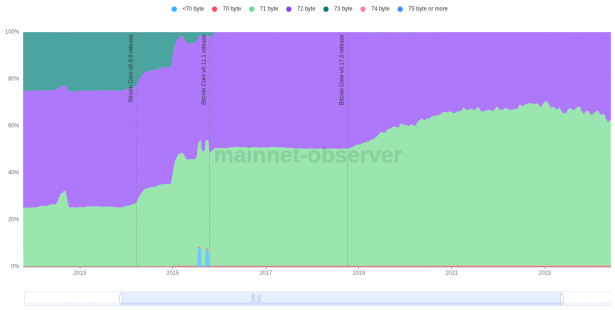


Fig. 1. Length of ECDSA signatures over time *(source: mainnet.observer)*.

type allow the classification of wallet implementations based solely on whether they adopt these features.

a) ***Low–R Grinding****:* Since October 2015, with the introduction of BIP–62, the signature lengths have been evenly distributed, reflecting the roughly 50/50 probability that r falls in the low or high range of the curve. As shown in Figure 1, a noticeable shift occurred around October 2018, when signatures started to measure 71 bytes instead of 72. This change coincides with the release of Bitcoin Core v0.17.0, which implemented low–r grinding to optimize the size of the transaction.

The process is conceptually simple: the signer iterates over candidate nonces until the resulting R_x value falls within the desired range. Each iteration produces a different signature for the same message and key pair, so the operation is repeated until the condition is met.

However, this optimization comes at a computational cost. As s grinding has already been performed, introducing r grinding increases the expected complexity from $O(2n)$ to $O(4n)$, where n represents the cost of a single signing operation.

Although this trait can be used for fingerprinting, its applicability is limited to a probabilistic framework. If a signature has a high–r value, it can be confidently classified within the groups of wallets that do not implement low–r grinding. However, the presence of a low–r value does not guarantee that the signature was produced by a wallet applying this optimization.

b) ***SIGHASH****:* When signing a Bitcoin transaction, a flag byte can be set to define which parts of the transaction are included in the signature. This mechanism allows for the following signing behaviors:

TABLE I
BITCOIN SIGHASH TYPES AND THEIR BEHAVIOR

SIGHASH Type	Behavior
SIGHASH_ALL	Signs all inputs and all outputs.
SIGHASH_NONE	Signs all inputs but no outputs.
SIGHASH_SINGLE	Signs all inputs and only the output with the same index.
SH_ALL – ACP	Signs only the current input and all outputs.
SH_NONE – ACP	Signs only the current input and no outputs.
SH_SINGLE – ACP	Signs only the current input and the corresponding output.

Some hardware wallets support the full range of SIGHASH options, while others restrict usage to **SIGHASH_ALL** to minimize security risks when spending transactions.

Similarly to the fingerprinting approach based on low–r grinding, the use of alternative SIGHASH types can serve as an identification trait. If an external observer detects a signature that employs a SIGHASH value other than **SIGHASH_ALL**, it can be confidently classified within the group of wallets that allow full SIGHASH functionality. In contrast, when **SIGHASH_ALL** is used, no distinction can be made, as this trait is universally supported across wallets.

V. TEST ENVIRONMENT

To evaluate the feasibility of fingerprinting the traits discussed in Section IV, we acquired several state-of-the-art hardware wallets currently available on the market for testing (see Table II for the detailed list). Despite some of them can be evaluated using standard Bitcoin test networks (such as testnet3, testnet4, or signet), others do not support these networks and can only operate on main–net. In such cases, the economic cost of conducting extensive tests would be prohibitively high.

To overcome this limitation, we created a parallel Bitcoin main–net, *mainlabnet*. The key property of *mainlabnet* is that it shares the same genesis block as the real main–net; from that block onward, however, we construct the blockchain according to our needs by mining the subsequent blocks ourselves.

A. The mainlabnet network

The main goal of creating a parallel main–net is to establish a fully independent and functional blockchain network for testing, research, and development.

At the same time, by creating a fully functional parallel main-net, we can perform extensive testing with a large volume of transactions without spending any real Bitcoin, avoiding any associated financial cost.

Although this new main-net will be built from the Bitcoin Core original source [13] it must ensure total isolation on the network to avoid any risk of "infection" with the real Bitcoin network. Otherwise, it would synchronize undesired data and compromise the integrity of our network.

Moreover, it is crucial to integrate SegWit and Taproot from the beginning, as these features were activated in later blocks, and any transaction using them in earlier blocks would be invalidated by consensus.

1) ***Architecture****:* The Bitcoin network is made up of nodes that run the Bitcoin protocol. These nodes play different roles, but the two main ones are users, who send transactions and verify blocks, and miners, who collect transactions and create new blocks including them. When a miner successfully creates a block, it shares it with the network. As these transactions spread across the network, they gradually become accepted as valid.

This architecture is simulated by building an interconnected container system with Docker, implementing a Bitcoin node on each of them.

a) Nodes: For this parallel main-net, two essential nodes are required, one serving as a "router", to facilitate information transfer between nodes, and one as a miner responsible for block creation. Additionally, other nodes joining the

network will act as block validators, relaying transactions and maintaining the blockchain's integrity.

b) Network: As mentioned above, network isolation must be accomplished. A Docker-based environment allows for full isolation on a single device. This setup enables the deployment of containerized Bitcoin nodes with a single command, creating a flexible and scalable architecture. Communication between nodes is managed using Bitcoin's *addnode* option, which connects each new node to a central "router" node through port 8333. Connections are restricted to an internal Docker network to ensure isolation. However, for broader use, nodes can also connect point–to–point to an external server-based router node, allowing users to join with pre-configured Docker containers or connect directly via port 8332 using a wallet.

*2) **Docker Node Configuration**:* Every container has a $\approx 5.11GB$ weight Ubuntu image with Bitcoin Core 24.0.1. The Docker setup includes:

a) DockerFile: Containing all the instructions needed to build the image. It downloads the source code directly from Github repository, allowing to modify **chainparams.cpp** file before compilation, enabling the establishment of different consensus rules between nodes on our network. The main differences in our modified file are twofold: the former sets the *SegwitHeight* parameter to activate the Segregated Witness addresses from the genesis block, while the latter adjusts Taproot's activation height to achieve the same effect. The Dockerfile outlines this sequence of steps, *install Dependencies*, *clone Bitcoin Core repository*, *modify chainparams.cpp* and *compile*. Additionally, some files are copied into the container: a customized *bitcoin.conf*, a docker entrypoint script, and a simple mining script. These components are essential for automating the node configuration and for setting up the Docker container environment. Furthermore, the Dockerfile manages the exposure of port 8333 by default and 8332 if RPC calls are expected to occur on the node.

b) docker-entrypoint.sh: This script performs several critical tasks to ensure the secure and reliable execution of Bitcoin processes within the container.

- Configures permissions for the Bitcoin data directory.
- Executes the Bitcoin daemon by default as a user.
- Establishes **datadir** location and verify its permissions.

It is specifically designed to facilitate the execution environment for each node.

c) bitcoin.conf: Every Bitcoin node has a **bitcoin.conf** file that allows customization of node properties. In order to work with this parallel main-net, this file needs some adjustments.

- Adding the "router" node IP with **addnode** command allows all nodes to be interconnected.
- It is crucial to allow the node to listen through all interfaces, settle the RPC port, and allow IPs in the specific range of the docker network for internal communication.
- It must be defined as a server.
- Although a cookie system can be established, user authentication can be performed from this file.
- Without sufficient historical transaction data, Bitcoin Core cannot accurately estimate the fees required for transactions. As a result, it is essential to specify a fee rate to ensure successful transaction broadcasting over the network. If this rate is not set, problems may arise when using RPC to send transactions.
- To avoid unnecessary processes, **dnsseed** can be set to zero, as real nodes will not be useful.
- As the chain will initially not have proof-of-work, it is necessary to set **minimumchainwork** to zero for a node to consider it a valid chain.
- By enabling **txindex** all transaction data is saved in a separate index, allowing the node to keep a full record of all transactions within the blockhain. This is particularly useful in this network for easier querying and tracking specific transactions for testing.

B. HW setup

To perform the fingerprinting tests, a set of widely used hardware wallets currently available on the market were selected. All devices were initialized with the same BIP–39 [14] seed phrase to ensure consistency across tests. Each wallet was connected to Sparrow Wallet or, if not supported, to its official companion software.

Table II lists the hardware wallets included in our evaluation.

TABLE II
HARDWARE WALLETS TESTED AND THEIR FIRMWARE VERSIONS

Hardware Wallet	Firmware Version
Ledger Nano S Plus (Bitcoin App)	2.3.0
BitBox02 (Multi)	9.21.0
Foundation Passport	2.3.5
Trezor Model T Safe 3	2.8.7
Trezor Model T Safe 5	2.8.7
Coldcard Mk4	5.2.0
Blockstream Jade	1.0.35
KeepKey	7.9.1
Keystone 3 Pro (BTC)	1.1.4
SeedSigner	0.7.0

VI. TEST RESULTS

All tests were conducted using the same seed to ensure that any observable differences in the signatures could be attributed to wallet–specific behavior.

A. Methodology

The tests performed to examine signature generation on each HW followed the procedure below:

1) Generate a random seed and select a child private key derived from BIP–32 [11].
2) Import the corresponding `xpub` from the private key's derivation path into Sparrow Wallet.
3) Prepare a UTXO locked with the previously derived private key.
4) Construct a PSBT spending the previous UTXO using a script that requires an ECDSA signature.
5) Sign the PSBT multiple times with each hardware wallet[2]
6) Observe whether the resulting signatures are identical or differ, as this indicates the implementation of RFC 6979.

[2]Without broadcasting transaction

7) Check if the wallet applies *low–r grinding*.
8) Repeat the signing process using different type of SIGHASH.

To ensure robustness, signatures were collected for multiple transaction structures (e.g, 1 input – 1 output, 3 inputs – 2 outputs). This allowed us to verify whether HW behavior remained consistent in different contexts.

Table III illustrates one case in which distinct groups can be identified:

- Wallets implementing RFC 6979 without low–r grinding.
- Wallets implementing RFC 6979 with low–r grinding.
- Wallets implementing anti–exfil [15] and no low–r grinding.

TABLE III
ECDSA SIGNATURE **R** AND **S** VALUES FOR 1-IN/1-OUT PSBT

Wallet	**r (B)**	**s (B)**	**r (hex)**	**s (hex)**
Ledger Nano S+	33	32	00ab...d7b	1f4d...df5
BitBox02	33	32	0084...acd	3a7d...1be
Foundation Pass.	33	32	00ab...d7b	1f4d...df5
Trezor Safe 3	33	32	00ab...d7b	1f4d...df5
Trezor Safe 5	33	32	00ab...d7b	1f4d...df5
Coldcard Mk4	32	32	635c...8a9	5b02...90d
Jade	32	32	635c...8a9	5b02...90d
KeepKey	33	32	00ab...d7b	1f4d...df5
Keystone 3 Pro	33	32	00ab...d7b	1f4d...df5
SeedSigner	32	32	635c...8a9	5b02...90d

Although these signatures reveal whether a wallet implements RFC 6979 or not, an external observer cannot exploit this information, since broadcasting two transactions with the same PSBT is infeasible. The only persistent fingerprint is the length of the signature, which correlates with a low–r grinding.

Additionally, tests were constrained by wallet software limitations: most wallets refused to sign PSBTs with SIGHASH types other than ALL. Table IV summarizes the attainable fingerprinting methods in selected HWs.

TABLE IV
LOW-r GRINDING AND SIGHASH SUPPORT. "+A" = ANYONECANPAY.

Wallet	**Low-r**	ALL	NONE	SINGLE	ALL+A	NONE+A	SINGLE+A
Ledger Nano S+	×	✓	✓	✓	✓	✓	✓
BitBox02	×	✓	×	×	×	×	×
Foundation Passport	×	✓	×	×	×	×	×
Trezor Safe 3	×	✓	×	×	×	×	×
Trezor Safe 5	×	✓	×	×	×	×	×
Coldcard Mk4	✓	✓	×	×	×	×	×
Jade	✓	✓	×	×	×	×	×
KeepKey	×	✓	×	×	×	×	×
Keystone 3 Pro	×	✓	✓	✓	✓	✓	✓
SeedSigner	✓	✓	×	×	×	×	×

VII. CONCLUSIONS

Our analysis demonstrates that hardware wallet fingerprinting, while challenging, is not entirely infeasible. Certain implementation choices, such as low–r grinding and SIGHASH support, introduce residual traits that can serve as potential fingerprints, even though the signature itself does not reveal any information. However, these traits offer only probabilistic identification and are insufficient for definitive attribution.

In general, hardware wallets exhibit a high degree of privacy compared to software wallets, significantly reducing fingerprinting opportunities.

Future research should extend this analysis by incorporating fingerprints from software wallets to evaluate real–world feasibility. A characterization of these fingerprints could support a study focused on preventing probabilistic address clustering and transaction–level tracking, thereby mitigating critical privacy risks for Bitcoin wallet users.

ACKNOWLEDGEMENTS

We will like to thank David Corral for their help on some initial experiments of that paper. This work is funded by the projects PID2021-125962OB-C33 SECURING/NET and PID2024-156914OB-C43 SAFE-BLOCKCHAIN, funded by the Ministerio de Ciencia e Innovación, la Agencia Estatal de Investigación and the European Regional Development Fund (ERDF); the DANGER Strategic Project of Cybersecurity C062/23, funded by the Spanish National Institute of Cybersecurity through the European Union - NextGenerationEU and the Recovery, Transformation and Resilience Plan; and the Catalan Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) grant SGR2021-00643. We also thank David Corral for his participation in the project and his valuable contributions.

REFERENCES

[1] I. Misra, "Wallet Fingerprints: Detection & Analysis," *Ishaana Blog*, 2023. [Online]. Available: https://ishaana.com/blog/wallet_fingerprinting/
[2] M. Šorf, P. Švenda, and Ł. Chmielewski, "Large-Scale Security Analysis of Hardware Wallets," in *Availability, Reliability and Security (ARES 2025)*, LNCS vol. 15995, pp. 360–377, Springer, 2025. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-032-00633-2_21
[3] D. Johnson, A. Menezes, S. Vanstone: "The Elliptic Curve Digital Signature Algorithm (ECDSA)", International Journal of Information Security, vol. 1, n. 1, pp. 36–63, 2001.
[4] P. Wuille: "BIP-340: Schnorr Signatures for secp256k1", Bitcoin Improvement Proposals, 2020. Available at: https://github.com/bitcoin/bips/blob/master/bip-0340.mediawiki
[5] E. Lombrozo, J. Lau, and P. Wuille: "BIP-141: Segregated Witness (Consensus Layer)", Bitcoin Improvement Proposals, 2015. Available at: https://github.com/bitcoin/bips/blob/master/bip-0141.mediawiki
[6] P. Wuille, J. Nick, and A. Towns: "BIP-341: Taproot: SegWit Version 1 Spending Rules", Bitcoin Improvement Proposals, 2020. Available at: https://github.com/bitcoin/bips/blob/master/bip-0341.mediawiki
[7] A. Chow: "BIP-174: Partially Signed Bitcoin Transactions (PSBT)", Bitcoin Improvement Proposals, 2018. Available at: https://github.com/bitcoin/bips/blob/master/bip-0174.mediawiki
[8] ITU-T, "X.690: Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)," 2002.
[9] P. Wuille: "BIP-62: Dealing with Malleability", Bitcoin Improvement Proposals, 2014. Available at: https://github.com/bitcoin/bips/blob/master/bip-0062.mediawiki
[10] M. Brengel and C. Rossow, "Identifying Key Leakage of Bitcoin Users," in *Research in Attacks, Intrusions, and Defenses (RAID 2018)*, Lecture Notes in Computer Science, vol. 11050, pp. 623–643, Springer, Cham, 2018. [Online]. Available: https://doi.org/10.1007/978-3-030-00470-5_29
[11] P. Wuille: "BIP-32: Hierarchical Deterministic Wallets", Bitcoin Improvement Proposals, 2012. Available at: https://github.com/bitcoin/bips/blob/master/bip-0032.mediawiki
[12] T. Pornin, "Deterministic Usage of the Digital Signature Algorithm (DSA) and Elliptic Curve Digital Signature Algorithm (ECDSA)," RFC 6979, 2013. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc6979
[13] Bitcoin Core Developers, "Bitcoin," GitHub repository, [Online]. Available: https://github.com/bitcoin/bitcoin
[14] M. Palatinus, P. Rusnak, A. Voisine, and S. Bowe: "BIP-39: Mnemonic Code for Generating Deterministic Keys", Bitcoin Improvement Proposals, 2013. Available at: https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki
[15] J. Nick and A. Poelstra, "Anti-Exfil: Stopping Key Exfiltration," *Medium Blog*, 2021. [Online]. Available: https://medium.com/blockstream/anti-exfil-stopping-key-exfiltration-589f02facc2e

Explorando mejoras en el protocolo Albatross: una validación experimental en escenarios distribuidos

Cristian González García
Universidad de Oviedo
C/Federico García Lorca, nº18
gonzalezcristian@uniovi.es

Ana I. Gómez
Universidad Rey Juan Carlos
Calle Tulipán s/n
ana.gomez.perez@urjc.es

Jesus R. Abascal, Domingo Gómez-Pérez
Universidad de Cantabria
Avenida de los Castros s/n
jesus.abascal@alumnos.unican.es
domingo.gomez@unican.es

***Abstract*—Este presenta un análisis experimental de Albatross, un protocolo distribuido de generación de aleatoriedad verificable basado en técnicas de Packed Publicly Verifiable Secret Sharing (πPPVSS), y propone una mejora computacional mediante su adaptación a criptografía de curvas elípticas. El protocolo original opera sobre un grupo multiplicativo de gran tamaño, lo que implica operaciones costosas y limitaciones en escenarios con recursos restringidos. En esta investigación evaluamos una variante denominada Albatross-ECC, que sustituye las operaciones clásicas basadas en cuerpos finitos por primitivas eficientes sobre la curva elíptica Curve25519. Esta modificación reduce de forma significativa el coste computacional, el ancho de banda requerido y la memoria necesaria para la verificación de pruebas LDEI y DLEQ, lo que habilita la ejecución del protocolo en entornos distribuidos ligeros, incluyendo ≪computación edge≫ para dispositivos IoT. Parte de la mejora proviene de incorporar un mecanismo propuesto en la literatura para mapear bloques de bits a puntos de la curva llamado Elligator, optimizando la comunicación y la generación de salidas aleatorias. Presentamos una implementación funcional del protocolo, así como una evaluación comparativa entre la versión original y la propuesta basada en Elliptic Curve Cryptography (ECC).**

***Index Terms*—Albatross, ECC, Distributed Cryptography, Verifiable Randomness**

I. Introducción

En criptografía, es necesario contar con fuentes fiables de aleatoriedad para la generación de números con una determinada distribución estadística. El concepto de faro o baliza de aleatoriedad, en ingles *beacon*, se definió originalmente en 1983 [6] como un servicio que genera regularmente valores aleatorios que no son predecibles o susceptibles de ser manipulados en escenarios en que la actuación de usuarios maliciosos puedan subvertir el servicio. En la práctica esto incluye diferentes tipos de acciones: aliarse para sesgar el resultado, perturbar las fuentes de aleatoriedad, inutilizar el servicio para ciertos usuarios objetivo, etc.

La generación de aleatoriedad puede clasificarse en tres tipos principales. El primer tipo son los generadores de números pseudoaleatorios (PRNG), que son algoritmos que generan secuencias de números con buenas propiedades estadísticas. El esquema más usado en estos algoritmos es, a partir de un valor aleatorio, también llamada la semilla inicial, se genera un valor de salida y se modifica la semilla de forma que se pueden ejecutar indefinidamente. Muchos PRNGs son muy rápidos, requieren poca memoria pero pueden ser predecibles con suficientes muestras. El siguiente tipo son los generadores de números verdaderamente aleatorios (TRNG) que se basan en fenómenos físicos, como el ruido térmico. En teoría, la calidad de las secuencias de números son mejores, al basar su aleatoriedad en una teoría científica [10]. No obstante, los diseños propuestos se enfrentan a limitaciones en costos, complejidad y son sensibles a factores externos. Finalmente, los generadores de números pseudoaleatorios criptográficamente seguros (CSPRNG) emplean algoritmos especialmente diseñados para que su salida sea computacionalmente indistinguible de TRNGs. Por ello, el costo computacional de este tipo de generadores suele ser alto y limita su aplicación en algunos escenarios. Esto hace que sea necesario el estudio de mejoras en su rendimiento.

El avance de las redes de comunicaciones ha propiciado que el servicio proporcionado por estos faros de aleatoriedad se generalice a otros escenarios. La democratización de las redes de comunicaciones y el tiempo de cómputo ha dado lugar a nuevas arquitecturas. Aún con el aumento de las soluciones ad hoc, a día de hoy siguen existiendo *servicios centralizados* [11] como el popular Random.Org. La web mencionada anterior recopila aleatoriedad de mediciones de fenómenos atmosféricos y ofrece un servicio de pago para acceso a grandes volúmenes de secuencias aleatorias. Esto ha motivado el uso de otras fuentes públicas de entropía, en lo que se ha denominado *servicios implícitos*, es decir utilizar mediciones externas de fuentes públicas, procesarlas y obtener números aleatorios. Una de las fuentes más populares son los mercados de valores y las fluctuaciones de la bolsa [2]. En aplicaciones distribuidas, con presencia de usuarios maliciosos, es preferible alguna forma de poder verificar la generación de la aleatoriedad y que esta sea "consensuada", repartiendo la carga entre varios usuarios, ofreciendo más tolerancia ante fallos en nodos puntuales o pérdidas de conexión. Por ello, nacen los *servicios descentralizados*, en inglés *Distributed Randomness Beacon* (DRB) [2]. En este último caso, la confianza se basa en que los números aleatorios se generan entre varias partes de forma distribuida, con una responsabilidad horizontal de forma equitativa en cuanto a recursos.

Un protocolo DRB debe proporcionar garantías de imparcialidad aun cuando un porcentaje minoritario de las partes involucradas en el proceso de generación se comporte de forma maliciosa. Estos sistemas se consideran robustos cuando solo una porción dada de los usuarios experimentan fallos, sean intencionados o no y el sistema es capaz de recuperar su funcionamiento normal. Además también se debe poder demostrar mediante los mensajes intercambiados a un tercero que no participa en el proceso que la aleatoriedad generada proviene de una determinada ejecución

del protocolo, es decir, debe poder verificarse públicamente.

El protocolo ALBATROSS propuesto por Cascudo y David [1] es un protocolo DRB. Su nombre ALBATROSS viene del inglés publicly **A**ttestab**L**e **BAT**ched **R**andomness based **O**n **S**ecret **S**haring que genera aleatoriedad por lotes y es públicamente verificable. Su seguridad criptográfica está basada en el uso de primitivas de intercambio secretos, firmas electrónicas basadas en el logaritmo discreto y pruebas de conocimiento cero. Como indica su nombre, no se genera un único valor de salida sino que obtenemos un lote de valores aleatorios, que siguen una distribución estadística uniforme, siempre y cuando una mayoría de participantes generan datos de entrada con similar distribución probabilística. El protocolo se basa en el consenso y tiene un grado de tolerancia ante participantes malintencionados.

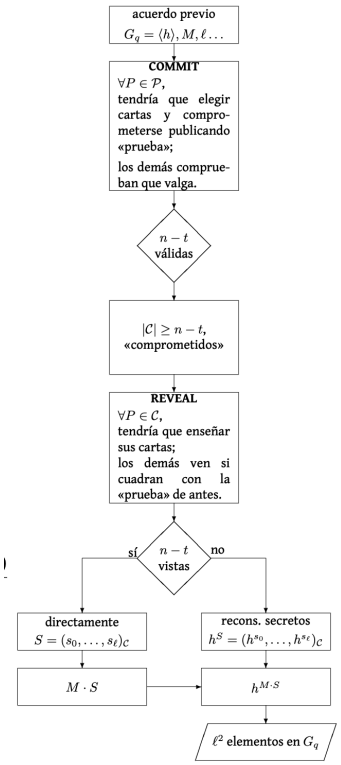


Fig. 1. Esquema del proceso

Albatross es una mejora o evolución de SCRAPE [12], un protocolo anterior de los mismos autores y para la misma tarea. El rendimiento de ambos protocolos se compara en torno a tres parámetros: el tamaño de la salida, la tolerancia a tramposos en función del número de participantes n y el número de operaciones entre todos los participantes al acabar el protocolo. En cuanto a la salida, Albatross proporciona más elementos en la salida con un tamaño de bloque que es proporcional a n^2 mientras que Scrape solo proporciona un elemento por ejecución. Respecto a la tolerancia a participantes maliciosos Albatross requiere una proporción mayor de usuarios que colaboren estrictamente con el protocolo. En Albatross, por las técnicas utilizadas para la reconstrucción, al menos la mitad de participantes más un número significativo que dependa de n deben colaborar activamente en el protocolo. Esto empeora respecto a Scrape en cuanto solo se requiere la mitad más una constante entera. Finalmente, respecto al número total de operaciones, Albatross presenta una mejora respecto a propuestas anteriores al contar con complejidad logarítmica respecto al número de exponenciaciones por cada interviniente y para cada elemento generado. Un detalle importante es que esta complejidad cubre el peor caso, referido a la deserción de algunos participantes. En el caso que todos colaboran, la complejidad se cifra en constante por el número de exponenciaciones y elemento pseudoaleatorio generado.

En este trabajo, nuestro objetivo es explorar el impacto de posibles mejoras en la complejidad del protocolo sustituyendo las operaciones criptográficas por criptografía de curva elíptica, debido a sus buenas propiedades. Nos centraremos en una curva estándar, curve25519 [7], que se eligió por su estandarización (NIST / IETF), eficiencia y resistencia a ataques conocidos, lo que la convierte en un caso representativo para análisis y optimización y porque se emplean ampliamente en protocolos modernos al ofrecen alta seguridad con claves más pequeñas, reduciendo consumo de recursos frente a RSA. Además de ser muy eficiente respecto al número de operaciones bit, la curva posee una correspondencia entre bloques de bits de tamaño constante y puntos de la curva llamada *Elligator* [8]. Aunque inicialmente fue propuesto como una forma de esconder el envío de puntos de esta curva elíptica, nosotros proponemos su uso para generar bloques de tamaño fijo, de forma que se ahorre transmisión de bits y se generen bits aleatorios en vez de números en un rango amplio. Para demostrar nuestros resultados, hemos realizado una implementación del protocolo, junto con las mejoras y siendo la primera con Elligator. Aunque los autores de este trabajo han encontrado una implementación de Albatross [1] basada en criptografía de curva elíptica, se ha desarrollado una implementación propia..

Este trabajo se organiza de la siguiente forma: En la Sección II se resumen brevemente los campos de aplicación de Albatross. A continuación, la Sección III describe los principales detalles de la propuesta Albatross con las etapas en las que se constituye y las operaciones en cada paso. La sección IV describe los cambios necesarios propuestos para la mejora de la complejidad computacional, que se implementan sobre una plataforma software descrita en la Sección VI y se comparan mediante simulaciones computaciones en la Sección VII. Finalmente se discuten resultados y trabajos futuros en la Sección VIII.

II. Aplicaciones de Albatross

La generación de aleatoriedad distribuida es un componente fundamental en numerosos protocolos criptográficos. Entre sus aplicaciones clásicas se encuentran la elección de parámetros en primitivas criptográficas, como la generación de claves, la creación de valores de relleno o *nonces*, así como su uso en protocolos presentes en sistemas de votación electrónica [3], en redes *blockchain* [4] y procesos de selección de líderes en esquemas de consenso basados en *Proof-of-Stake* [5].

En este contexto, el estudio computacional de Albatross con el uso de criptografía de curva elíptica amplía

[1] https://github.com/evapln/albatross

significativamente su aplicabilidad práctica, al reducir los costes computacionales y de comunicación y permitir su ejecución en entornos distribuidos con recursos limitados. Esto resulta especialmente adecuado para plataformas *blockchain* y sistemas de consenso para tareas como selección de validadores, formación de comités de *sharding*, generación de loterías verificables, ordenación aleatoria de transacciones, proporcionando aleatoriedad pública para operaciones como el mezclado de votos o la selección de escrutadores en sistemas de votación electrónica y protocolos de verificación descentralizada [1]. Su eficiencia lo hace compatible con arquitecturas modulares y redes modernas que dependen intensivamente de criptografía de curva elíptica.

Más allá del ámbito *blockchain*, la aleatoriedad distribuida puede ser aplicable a infraestructuras críticas y redes industriales (OT/ICS), facilitando la coordinación entre dispositivos IoT, RTUs y PLCs [17], en aprendizaje federado y protocolos de agregación segura permite inicializar con parámetros seguros de forma distribuida [18]. Finalmente, proporciona una base eficiente y verificable para protocolos de *Multi-Party Computation* (MPC), mecanismos tipo RANDAO, juegos verificables y cualquier esquema distribuido que requiera entropía colectiva resistente a sesgo.

III. Descripción Albatross

El protocolo opera sobre una red distribuida de nodos o participantes en la red, donde n es el número de participantes. Se establece una tolerancia t al numero máximo de participantes maliciosos en función de $t = \tau \cdot n$ para $0 < \tau < \frac{1}{2}$. Ambos parámetros fijan el número de secretos a cifrar por cada participante $\ell = n - 2t$, así como el tamaño del lote generado en la última etapa del protocolo.

Como parámetro público también se acuerda $G_q = \langle h \rangle$ un grupo con generador h y orden multiplicativo q , donde q es un número primo. En la propuesta original, este grupo era un subgrupo multiplicativo de un cuerpo finito de característica suficientemente grande.

Todos los parámetros se almacenan en un registro público que actúa como almacén de información sincronizada entre todos los nodos.

Un esquema general de las etapas en los que se divide el protocolo, junto con un resumen de sus principales operaciones se muestra en la Figura 1. A continuación se detallará cada etapa.

A. Commit

En esta etapa, se realiza el protocolo Packed Publicly Verifiable Secret Sharing (πPPVSS), que prueba a todos los participantes que los valores han sido fijados de antemano, sin revelar información de éstos. Esto impedirá modificaciones inadvertidas en las etapas siguientes. Para ello, cada nodo $P \in \mathcal{P}$ genera su clave privada sk_i y su clave pública $pk_i = h^{sk_i}$, que comparte en al registro público.

Entonces, cada participante elige un polinomio $p(X) \in \mathbb{Z}_q[X]$ de grado $t + \ell - 1$. Este polinomio se utiliza por el esquema de repartición de secretos, que se lleva a cabo evaluando el polinomio $p(X)$ en distintos puntos. Las evaluaciones en el rango $[-\ell + 1, 0]$ representan los secretos $s_j,\ j = 1, \ldots, \ell$, generados por el participante, mientras que las evaluaciones en el rango $[1, n]$ son repartidas a los participantes se denotan como $\sigma_i = p(i)$ como un vector de secretos. Para no revelar el contenido del vector, se publica $\hat{\sigma}_i = pk_i^{\sigma_i}$, $1 \leq i \leq n$. Cada participante recibe un fragmentos cifrados de este vector que se utilizará en la etapa de salida y, si es necesario para la reconstrucción del vector secreto.

Durante la fase de distribución del protocolo *πPPVSS*, cada participante sube al registro público una prueba llamada *Low-Degree Exponent Interpolation (LDEI)*, que demuestra que el polinomio utilizado en la distribución realmente existe. Remarcamos que no se pasa hasta la siguiente fase hasta que $n - t$ polinomios son validados.

B. Reveal

En la etapa de revelación, la verificación de las pruebas *LDEI* publicadas se reparte entre todos participantes. Se espera hasta que al menos $n - t$ participantes han validado, para que cada uno de los participantes validados publiquen su polinomio $p(x)$. Además, estos $n - t$ tienen que validar que se ha publicado los correctos a través de $\hat{\sigma}_i$. Para ello, cada participante genera los fragmentos cifrados utilizando el polinomio que se desea verificar y los compara con los fragmentos del registro público.

En esta etapa el protocolo se divide en dos ramas, si el polinomio es correcto, se continua evaluando los secretos, en otro caso es necesario realizar un paso adicional, descrito a continuación.

Recuperación de secretos: Cada participante debe descifrar y subir al registro público los $pk_i^{\sigma_i}$ recibidos de cada participante i. Estos fragmentos deben incluir una prueba de *DLEQ*, que garantiza que son válidos y coinciden con los distribuidos originalmente.

Una vez que al menos $n - t$ participantes han publicado los fragmentos descifrados validados por los participantes, se puede reconstruir el vector secreto asociado al polinomio, mediante la interpolación polinómica de Lagrange.

Finalmente, para asegurar la validez del proceso, se utiliza una prueba adicional llamada *localLDEI* que servirá para convencer de la validez de la reconstrucción a los participantes no implicados en dicha reconstrucción.

C. Output

Una vez completada la reconstrucción, si fuera necesaria, o si todos los polinomios publicados fueron correctos desde el inicio, se procede a la etapa final de salida.

En esta etapa, todos los secretos revelados por los participantes $s_j,\ j = 1, \ldots, \ell$, se organizan en una matriz T de dimensión $(n - t) \times \ell$, cada elemento s_j está operado en función h para cada participante con polinomio válido.

Esta matriz S se multiplica por una matriz M , que es la matriz de Vandermonde, construida a partir de w, un elemento aleatorio de grupo cíclico. El resultado de esta multiplicación es una nueva matriz ($\ell \times \ell$) que contiene ℓ^2 valores aleatorios que constituyen el resultado final del protocolo. Este último paso garantiza que todos los valores estén bien distribuidos, al mezclarse filas y columnas.

En caso de que se haya realizado la recuperación de secretos debido a polinomios incorrectos, el proceso de construcción de las matrices S y M , y su posterior multiplicación dan el mismo resultado.

D. Sistema de pruebas

Vamos a resumir brevemente las pruebas de conocimiento que se han mencionado en las etapas. Tanto DLEQ y LDEI son pruebas no interactivas, resultado de aplicar la heurística de Fiat-Shamir [9]. A través de Fiat-Shamir, el protocolo permite que la prueba se realice mediante un solo mensaje que puede ser verificado en cualquier momento, eliminando la necesidad de interacción.

La prueba LDEI se utiliza para demostrar una relación entre varios exponentes en un grupo cíclico. En este caso, un participante debe demostrar que un conjunto de elementos $x_1, \ldots, x_m$ puede expresarse en base a unos generadores $g_1, \ldots, g_m$ y exponentes dados por un polinomio con grado acotado y conocido. El objetivo de LDEI es garantizar que se conoce el polinomio secreto $p(X)$ sin revelarle. El protocolo $LOCAL_{LDEI}$ parte de las mismas premisas que en la prueba LDEI pero sin la intervención de un participante P que conozca el polinomio $p(X)$. Además, en este caso, todos los generadores son iguales, es decir, $g_1 = g_2 = \cdots = g_m = g$.

En el caso de DLEQ, se utiliza para probar que un participante P conoce un valor α tal que $g_1^\alpha = x_1$ y $g_2^\alpha = x_2$. La dificultad de resolver el problema está basada en la dificultad del logaritmo discreto.

IV. Albatross con curvas elipticas

Una curva elíptica es el conjunto de soluciones de una ecuación de Weierstrass:

$$y^2 = x^3 + a \cdot x^2 + b \cdot x + c.$$

Es conocido que se define de una forma natural una operación dentro de los puntos de la curva elíptica, añadiendo un punto extra, llamado *punto del infinito* que representa el elemento neutro. Recordamos informalmente la definición de la operación.

El elemento inverso respecto a la suma de un punto $A = (A_1, A_2)$ es $-A = (A_1, -A_2)$. Como ayuda al lector recordamos las operación suma sobre una curva elíptica. Si tenemos dos puntos de la curva a los que llamaremos A y B, la operación de suma consiste en trazar la recta que pasa por ambos puntos. Si esta recta corta a la curva en un punto, el inverso de dicho punto se considera (A+B). Si no corta, entonces el resultado es el punto del infinito. En el caso de operaciones sobre el mismo punto, se toma la tangente y $(A + A) = 2 \cdot A$. Esto define la multiplicación por escalar.

En particular, la curva *Curve25519* se define como las soluciones modulo $2^{255} - 19$ de la ecuación

$$y^2 = x^3 + 486662 \cdot x^2 + x.$$

El orden del grupo de esta curva elíptica es $q = 2^{252} + 27742317777372353535851937790883648493$ y el punto base del grupo se toma $h = (9, y)$, donde y es el punto de la curva con $2 \cdot y > 2^{255} - 19$.

En este caso, cada nodo $P \in \mathcal{P}$ genera su clave privada sk_i y su clave pública $pk_i = sk_i \cdot h$, que comparte en el registro público. De la misma forma, se ha publicado anteriormente el generador h y orden multiplicativo q que se considera, por la curva elegida, suficientemente grande.

De esta forma, el protocolo DLEQ se enuncia de la siguiente manera. Dados dos generadores G_1, G_2 y dos puntos X_1, X_2 , demostrar si existe un α tal que $X_1 = \alpha \cdot G_1$ y $X_2 = \alpha \cdot G_2$

De forma similar se pueden adaptar la prueba LDEI, en el que un conjunto de puntos $X_1, \ldots X_m$, una lista de enteros $\alpha_1, \ldots, \alpha_m$ puede expresarse en base a generadores $h_1, \ldots, h_m$ y un polinomio con coeficientes enteros $z(x)$, de forma que $X_1 = z(\alpha_1) \cdot h_1, \ldots, X_m = z(\alpha_m) \cdot h_m$.

La aplicación Elligator [7] toma un número $r \in \{0, \ldots 2^{255} - 19\}$. Entonces,

$$u = -486662 \cdot \left(\frac{r^2}{1 + r^2}\right) \pmod q,$$
$$\text{Calculamos } v^{(2)} = u^3 + Au^2 + u \pmod p.$$

Si $v^{(2)}$ es residuo cuadrático, entonces (u, v) es el punto en la curva. Si no, usamos $u' = -u - A$ y se realiza de nuevo el cálculo anterior. También existe una formula cerrada para aplicación inversa. Inicialmente, hemos hecho uso de la aplicación inversa pero queremos hacer notar que es posible mejorar la comunicación enviando bloques de 255 bits.

Como resultado inicial hemos utilizado para la construcción de la matriz T en la etapa de output.

V. Ventajas de Albatross basado en Curvas Elípticas

El protocolo Albatross, definido originalmente sobre un grupo multiplicativo $G_q = \langle h \rangle$ de orden primo, requiere tamaños de clave elevados (2048–3072 bits) [13] y operaciones costosas, lo que limita su eficiencia en escenarios con numerosos participantes. Sustituir este grupo clásico por una curva elíptica estándar (por ejemplo, Curve25519) proporciona mejoras significativas en términos de eficiencia computacional, ancho de banda y aplicabilidad.

1) Reducción de consumo computacional: La criptografía de curvas elípticas (ECC) ofrece un nivel de seguridad de 128 bits utilizando claves de 256 bits. Esta reducción impacta directamente en:

- el tamaño de los *commitments*,
- el coste de las pruebas DLEQ,
- el tamaño de los fragmentos πPPVSS,
- las matrices generadas en la fase final del protocolo,
- el tráfico total en la red.

En conjunto, Albatross-ECC no solo reduce el coste computacional, memoria requerida y las comunicaciones entre los nodos. Más concretamente, el uso de ECC combinado con la aplicación Elligator disminuye significativamente el número de ciclos de reloj y memoria en reconstrucciones polinómicas y verificación masiva de pruebas DLEQ y LDEI al fijar el tamaño de bloque de bits.. Esta eficiencia permitiría ejecutar Albatross en dispositivos mas ligeros como: dispositivos IoT, nodos *edge* y hardware de bajo consumo.

2) Ventajas frente a ataques cuánticos: Aunque ECC no es resistente a Shor [15] en un escenario de computación cuántica plena, presenta ventajas frente a ataques híbridos cuántico-clásicos y facilita la migración futura hacia protocolos post-cuánticos. La estructura modular permite reemplazar fácilmente la curva subyacente en caso de disponer de alternativas PQC eficientes.

VI. Implementación

Este sistema implementa el protocolo ALBATROSS, que es distribuido y permite la generación de aleatoriedad robusta y verificable. Se ha implementado en Python sobre una red simulada de nodos que interactúan en un servidor Flask. El protocolo se apoya en técnicas criptográficas de Publicly Verifiable Secret Sharing (PPVSS), con verificación mediante pruebas LDEI y DLEQ, y tolerancia a nodos maliciosos.

La **implementación** se compone de una red distribuida (Network), los nodos (Node), el Protocolo ALBATROSS, la implementación de ECC utilizando Elligator, el PPVSS, el LDEI, el DLEQ, el servidor Flask.

La Network permite crear y gestionar los nodos con claves criptográficas, asigna vecinos mediante una combinación de conexión lineal y enlaces aleatorios y propaga claves públicas mediante sincronización tipo gossip.

Los nodos tienen una clave secreta y pública, un Ledger local con datos compartidos y poseen la capacidad de ejecutar las diferentes fases del protocolo: commit, reveal, recovery, reconstruction, output. Se puede especificar el número de nodos honestos o maliciosos, permitiendo simular ataques y evaluar la robustez del sistema. Para ofrecer eficiencia, se paralelizan todo con threading.

El protocolo ALBATROSS implementado ejecuta las diferentes fases: **Commit**: distribución de secretos con PPVSS; **Reveal**: revelación de polinomios; **Output**: reconstrucción de aleatoriedad y uso de Elligator; **Recovery**: recuperación de nodos fallidos; **Reconstruction**: reconstrucción de secretos mediante matrices de Vandermonde, realizando así la interpolación en el exponente de forma segura y sin revelar valores intermedios.

El Publicly Verifiable Secret Sharing (PPVSS) es el encargado de generar los polinomios aleatorios, evaluar y cifrar fragmentos, verificar públicamente con LDEI, y reconstruir secretos con interpolación de Lagrange en el exponente. Así garantiza que los fragmentos compartidos pueden ser verificados públicamente sin revelar el secreto.

El LDEI (Linear Degree Equality in the Exponent) es una técnica criptográfica utilizada en esquemas PPVSS para verificar que los fragmentos cifrados de un secreto compartido corresponden a una evaluación de un polinomio de grado conocido, sin revelar el polinomio ni los secretos. Se utiliza en el PPVSS y en los nodos para realizar la prueba de integridad de los fragmentos cifrados evitando posible fraude y permitiendo una prueba pública por parte de los nodos.

El DLEQ se utiliza durante la fase de recuperación, cuando un nodo desencripta fragmentos de otros nodos que han fallado. Los nodos lo utilizan para realizar una prueba de corrección en la desencriptación de fragmentos, permitiendo verificar por los nodos que la desencriptación fue correcta, otorgando seguridad para evitar el falseamiento de datos y permitiendo integridad en la reconstrucción.

El servidor Flask levanta una API REST para que los nodos interactúen entre ellos, y mediante un controlador (NodeController) gestionar las operaciones de cada nodo.

El **flujo de ejecución** consta de 5 fases. **Inicialización**: crea la red con 1 nodo por participante, asigna los vecinos y se sincronizan claves públicas. El número de participantes se puede definir al inicio. **Commit**: cada nodo ejecuta PPVSS.distribute(), se publican fragmentos cifrados y se verifica con LDEI.**Reveal**: los nodos revelan sus polinomios y se verifica la consistencia de los datos. **Output**: si suficientes nodos revelan correctamente, se reconstruye la aleatoriedad. Si no, se activa la fase de recuperación. **Recovery** y **Reconstruction**: los nodos honestos desencriptan fragmentos de nodos fallidos, se verifica con DLEQ, y se reconstruye el secreto usando interpolación en el exponente, que se activa automáticamente si no se alcanza el umbral de revelaciones.

Para la **toma de tiempos** se realizan 4 medidas, sin contar el tiempo de levantamiento del servidor ni de creación de los nodos. La primera es el tiempo total de ejecución desde que comienza hasta que termina, y el tiempo separado de las tres fases: commit, reveal y output.

La implementación desarrollada por nosotros está disponible públicamente en [2].

VII. Resultados

En los experimentos realizados con el protocolo Albatross se evaluaron tres variantes: el algoritmo clásico, una implementación basada en curvas elípticas (ECC) y otra con Elligator. La tabla I muestra las pruebas realizadas en una simulación con redes diferentes con variación de los nodos totales y de si tenían nodos maliciosos, siento este el caso peor con un máximo de 1/3 y sin nodos maliciosos, para comprobar el caso mejor. Se utilizaron tamaños de clave de 1024 en el caso clásico y se midieron tiempos en la fase de output que comprende recovery y reconstruction.

En escenarios benignos, sin nodos maliciosos, los tiempos de ejecución fueron muy similares entre las tres opciones, con promedios cercanos a 2,1 segundos, siendo Elligator ligeramente más eficiente. Sin embargo, la presencia de nodos maliciosos produjo un impacto significativo en el rendimiento. El algoritmo clásico mostró una degradación extrema, en el caso peor estudiado, de hasta 488 segundos, lo que representa un incremento de más de 4 veces respecto al mejor, Elligator, y más de 214 veces más lento respecto a su versión sin nodos maliciosos. Por su parte, ECC y Elligator demostraron mejores tiempos, con incrementos de 49 veces en el caso peor respecto al caso mejor en la misma red con el mismo número de nodos. Además, respecto a estas dos implementaciones, Elligator es casi siempre ligeramente más rápida. En conclusión, aunque los tres algoritmos prácticamente equivalentes en entornos benignos, Elligator ofrece el mejor rendimiento, seguido por ECC, sobre todo cuando aumenta el número de nodos maliciosos. Mientras, el algoritmo clásico resulta altamente vulnerable en lo que respecta al tiempo necesario para la reconstrucción de la aleatoriedad poseyendo un incremento exponencial frente al polinómico de ECC y Eligator. Esto se puede observar en la Figura 2.

VIII. Conclusiones y Trabajos Futuros

Los resultados muestran mejoras computacionales cuando existen usuarios maliciosos. Tomando en cuenta los parámetros de seguridad necesarios para que el logaritmo discreto tenga suficiente seguridad computacional, es necesario tomar tamaños de clave en torno a 3000 bits [13], es de esperar que la diferencia de tiempos se

[2]https://github.com/domingoUnican/AlbatrossEllipgator

TABLE I

RESULTADOS DE LAS PRUEBAS DEL PROTOCOLO ALBATROSS CON DIFERENTES ALGORITMOS:LAS COLUMNAS DESPLIEGAN LOS TIEMPOS (SEGUNDOS) PARA DISTINTAS PROPUESTAS (CLÁSICO, ECC SIMPLE Y ECC CON ELLIGATOR PARA DISTINTOS NUMERO DE NODOS Y USUARIOS MALICIOSOS.

Nodos totales	Maliciosos	Clásico	ECC Simple	Elligator
5	0	2.10	2.05	2.06
5	1	2.09	18.67	2.05
10	0	2.15	2.10	2.09
10	3	30.47	29.49	29.55
20	0	2.16	2.13	2.09
20	6	64.89	55.82	51.77
30	0	2.19	2.16	2.12
30	10	179.19	75.65	75.90
40	0	2.28	2.15	2.15
40	13	488.05	106.63	106.43



Fig. 2. Comparación de tiempos entre las tres propuestas

incremente para mayores niveles de seguridad. La propuesta de Elligator presenta mejoras sobre la única utilización de curvas elípticas, además de ser la primera implementación realizada para Albatross. Estos resultados son prometedores respecto a su uso en escenarios en dispositivos con bajos recursos computaciones. En trabajos futuros, nuestra intención es estudiar el impacto de usuarios maliciosos en el algoritmo, además de evaluar en profundidad el desempeño computacional, junto con otros parámetros como la memoria y su desempeño sobre dispositivos de bajos recursos como dispositivos IoT [14]. Además, como la computación cuántica es un riesgo para las ECC, se plantea investigar la aplicación de las operaciones sobre curvas elípticas descritas en este trabajo en el contexto de computación cuántica, permitiendo estimar los recursos necesarios y evaluar la viabilidad práctica de ataques cuánticos. Por último, queremos explorar su aplicación en nuevos escenarios como redes industriales en conjunción con otras primitivas criptográficas.

AGRADECIMIENTOS

Domingo Gómez Pérez está financiado por la CÁTEDRA UNIVERSIDAD DE CANTABRIA-INCIBE DE NUEVOS RETOS EN CIBERSEGURIDAD, financiada por la Unión Europea NextGeneration-EU, Plan de Recuperación, Transformación y Resiliencia, a través de INCIBE.

Cristian González García y Ana I. Gómez acreditan estar parcialmente financiados mediante el proyecto PID2023-151238OA-I00, financiada por MICIU/AEI/10.13039/501100011033 y por FEDER, UE.

Jesús R. Abascal está parcialmente financiado bajo programa de doctorado industrial titulado SEGURIDAD Y PRIVACIDAD EN LA IA PARA INFRAESTRUCTURAS CRÍTICAS (SPIAFIC) y subvencionado por el Vicerrectorado de Investigación, Transferencia y Doctorado de la Universidad de Cantabria.

REFERENCES

[1] Cascudo, Ignacio and David, Bernardo. *ALBATROSS: publicly attestable batched randomness based on secret sharing*. (2020).

[2] Choi, Kevin and Manoj, Aathira and Bonneau, Joseph. *Sok: Distributed randomness beacons*. (2023).

[3] Chondros, Nikos and Zhang, Bingsheng and Zacharias, Thomas and Diamantopoulos, Panos and Maneas, Stathis and Patsonakis, Christos and Delis, Alex and Kiayias, Aggelos and Roussopoulos, Mema. *D-DEMOS: A distributed, end-to-end verifiable, internet voting system*. (2016).

[4] Kokoris-Kogias, Eleftherios and Jovanovic, Philipp and Gasser, Linus and Gailly, Nicolas and Syta, Ewa and Ford, Bryan. *Omniledger: A secure, scale-out, decentralized ledger via sharding*. (2018).

[5] Aggelos Kiayias and Alexander Russell and Bernardo David and Roman Oliynykov. *Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol*. (2017).

[6] Rabin, Michael O. *Transaction protection by beacons*. (1983).

[7] Bernstein, Daniel J. *Curve25519: new Diffie-Hellman speed records*. (2006).

[8] Bernstein, Daniel J. and Hamburg, Mike and Krasnova, Anna and Lange, Tanja. *Elligator: elliptic-curve points indistinguishable from uniform random strings*. (2013).

[9] Fiat, Amos and Shamir, Adi. *How to prove yourself: Practical solutions to identification and signature problems*. (1986).

[10] Ana I. Gomez and Markus Kiderlen and Florian Pausinger. *Improved entropy bounds for parity filtered self-timed ring based random number generators*. (2022).

[11] Kelsey, John and Brandã. *A reference for randomness beacons: Format and protocol version 2*. (2019).

[12] Cascudo, Ignacio and David, Bernardo. *SCRAPE: Scalable randomness attested by public entities*. (2017).

[13] Lenstra, Arjen K and Verheul, Eric R. *Selecting cryptographic key sizes*. (2001).

[14] Arias, Santiago and Gó. *Desarrollo de una plataforma para la evaluació*. (2024).

[15] Peter W. Shor. *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*. (1997).

[16] Ribeiro, Leonardo C. and Marcelino, Ana C. O. and Garcia, Guilherme A. and Goncalves, Desiree S. and Tarelho, Luiz V. G. and Correa, Leandro P. and Chapetta, Wladmir A. and De Oliveira Sa, Alan and Machado, Raphael C.S.. *2018 Workshop on Metrology for Industry 4.0 and IoT*. (2018).

[17] Ribeiro, Leonardo C. and Marcelino, Ana C. O. and Garcia, Guilherme A. and Goncalves, Desiree S. and Tarelho, Luiz V. G. and Correa, Leandro P. and Chapetta, Wladmir A. and De Oliveira Sa, Alan and Machado, Raphael C.S.. *2018 Workshop on Metrology for Industry 4.0 and IoT*. (2018).

[18] Behnia, Rouzbeh and Riasi, Arman and Ebrahimi, Reza and Chow, Sherman SM and Padmanabhan, Balaji and Hoang, Thang. *Efficient secure aggregation for privacy-preserving federated machine learning*. (2024).

Desarrollo Seguro de Software en BizDevOps: Incorporación Temprana de Metas y Controles

Alejandra Selva-Mora
ECCI-PPCI-CITIC Universidad de Costa Rica
San José, Costa Rica
alejandra.selvamora@ucr.ac.cr

Christian Quesada-López
ECCI Universidad de Costa Rica
San José, Costa Rica
cristian.quesadalopez@ucr.ac.cr

Antonio Santos-Olmo
Universidad de Castilla-La Mancha
Castilla-La Mancha, España
antonio.santosolmo@uclm.es

Luis Enrique Sánchez
Universidad de Castilla-La Mancha
Castilla-La Mancha, España
luise.sanchez@uclm.es

Eduardo Fernández-Medina
Universidad de Castilla-La Mancha
Castilla-La Mancha, España
eduardo.fdezmedina@uclm.es

***Resumen*—La adopción de enfoques de desarrollo de software en entornos BizDevOps pretende alinear el desarrollo de software con los objetivos empresariales, mejorando la velocidad de entrega. Sin embargo, este alineamiento plantea desafíos en ciberseguridad, puesto que no siempre se asegura la incorporación temprana de estrategias de seguridad. Este artículo propone, a partir de un análisis sistemático del estado del arte y del estado de la práctica en marcos de seguridad y BizDevOps, la incorporación de una nueva fase en el ciclo de negocio (Biz) de BizDevOps, enfocada en definir y asegurar el software desde las primeras etapas de desarrollo. El proceso planteado busca considerar sistemáticamente los objetivos de seguridad empresarial, identificando la deuda de seguridad acumulada para poder pagarla en forma oportuna. Este trabajo forma parte de una investigación en progreso que explorará actividades adicionales para fortalecer el diseño seguro de software.**

***Index Terms*—ciberseguridad, DevOps, DevSecOps, SecDevOps, BizDevOps, desarrollo de software, desarrollo ágil de software, marco de seguridad, deuda de seguridad**

I. Introducción

Los avances en tecnologías de la información (TI) han sido impulsados por la transformación digital empresarial, la cual ha consolidado la adopción de metodologías ágiles e híbridas para desarrollar software que se adapte continuamente a los cambios. Considerando que la información sensible se aloja en Internet, convirtiendo a las aplicaciones web en objetivos atractivos para los cibercriminales [1], la integración de seguridad con metodologías ágiles plantea retos importantes [2], principalmente porque las medidas de seguridad tienden a disminuir los niveles de agilidad [3] y porque factores como la falta de conocimiento en seguridad y la complejidad derivada de la evolución continua de requerimientos dificultan alcanzar niveles adecuados de seguridad [2]. Se requiere, por tanto, un enfoque proactivo que abarque el diseño, no solo la implementación y las pruebas [4].

El enfoque DevOps ha facilitado la sincronización entre los equipos de desarrollo y de operaciones, reportándose beneficios como la disminución de incidencias y el aumento de flexibilidad y automatización. Sin embargo, el desarrollo debe alinearse también con las necesidades organizacionales, por lo que surge BizDevOps, que promueve la colaboración entre equipos de negocio, desarrollo y operaciones de TI en un entorno ágil [5]. Esta colaboración favorece la transferencia de conocimiento del negocio, el aseguramiento de software de alta calidad y la integración adecuada de las aplicaciones en producción.

A pesar de los avances en colaboración, se ha identificado una carencia de seguridad, ya que ésta suele considerarse como una última comprobación cuando el software ya está desarrollado [6]. Surge entonces la necesidad de integrar estratégicamente prácticas de seguridad [2] en el flujo BizDevOps sin comprometer la agilidad, considerando las necesidades del negocio y logrando un estado de seguridad continua [6]. Este enfoque requiere reconocer el contexto organizacional para definir objetivos de seguridad [7] a nivel empresarial.

Este artículo presenta un proceso, resultado de las brechas identificadas mediante un análisis sistemático de modelos de la industria, para establecer una estrategia de desarrollo seguro de software en entornos BizDevOps con el objetivo de reducir la deuda de seguridad y mejorar la resiliencia organizacional en entornos ágiles. La propuesta forma parte de un marco más amplio que incluirá: la definición de la estrategia de seguridad empresarial (Biz); el análisis de requerimientos con un enfoque de seguridad por diseño (Dev); y la supervisión y análisis de incidentes de seguridad (Ops).

En la sección II se presentan antecedentes de BizDevOps y propuestas de implementación de seguridad. La sección III presenta el proceso para establecer la estrategia de seguridad y, para finalizar, la sección IV muestra las conclusiones de esta propuesta, así como el trabajo futuro por realizar.

II. Seguridad en Entornos BizDevOps

Según Gall [8], las metodologías ágiles tradicionalmente presentan una desconexión entre los equipos de desarrollo (Dev) y los aspectos operativos (Ops). DevOps surgió como respuesta a esta problemática, buscando equilibrar la velocidad del desarrollo con la fiabilidad y estabilidad del sistema. Esta colaboración efectiva ha permitido obtener tiempos de respuesta reducidos, así como una corrección rápida de errores mediante integración y despliegue continuos, proporcionando ventajas competitivas significativas a las empresas [8]. Sin embargo, el desarrollo de software no surge de los equipos

técnicos, sino de estrategias, metas y necesidades organizacionales. Las partes interesadas del negocio deben participar activamente en el proceso, estableciendo un flujo de extremo a extremo entre la demanda del cliente y la entrega rápida de productos o servicios [5].

II-A. Ingeniería de Software Continua y BizDevOps

La ingeniería de software continua propone un enfoque integral que aborda las necesidades organizacionales mediante la planificación, desarrollo, operación y evaluación del software. Este enfoque considera estas actividades no como fases discretas, sino como un flujo continuo de actividades relacionadas que garantiza que los requerimientos organizacionales sean considerados desde el inicio, involucrando a todas las partes interesadas y centrando las actividades de desarrollo en el cumplimiento de dichos requerimientos.

BizDevOps integra a las partes interesadas del negocio directamente en el proceso de desarrollo de software, logrando unificar las perspectivas de negocio, desarrollo y operaciones para asegurar que las iniciativas de TI estén consistentemente alineadas con los objetivos de negocio [9]. Para ello, incorpora diversas fases durante el flujo de trabajo, tal y como se muestra en la *Tabla I* [5], realizando continuamente actividades como planificación, verificación, cumplimiento y seguridad, entre otras [10]. La seguridad continua emerge como un elemento crítico que debe considerarse a lo largo de todo el ciclo de vida de las aplicaciones, apoyándose en esquemas que permitan identificar vulnerabilidades oportunamente [10].

II-B. Importancia de la Seguridad Integral

Según Williams [4], el software seguro requiere un enfoque proactivo que abarque el diseño, no solo implementación y pruebas. Esto necesita de una alta participación de los equipos de seguridad en todas las etapas del ciclo de vida del desarrollo. Además, una preocupación real por la seguridad debe vincularse con la identificación temprana y precisa de requerimientos de seguridad, tanto a nivel de aplicaciones específicas como en su interacción con otros componentes o sistemas [7].

Tabla I
FASES DEL FLUJO BIZDEVOPS [5]

Fase	Descripción
Adaptar	Analiza la retroalimentación para establecer iniciativas, hojas de ruta y planes iniciales.
Definir	Define perspectivas de las partes interesadas para luego dividirlas en componentes funcionales.
Alinear	Establece un modelo compartido del producto por entregar que alinee todas las necesidades.
Aprobar	Crea un acuerdo de las partes interesadas que indica que la solución diseñada satisface sus necesidades.
Planificar	Define requerimientos, métricas, características nuevas y mejoradas, y planifica la seguridad.
Crear	Codifica el software, incluyendo tareas como diseño, construcción y pruebas, entre otras.
Verificar	Realiza actividades relacionadas con el aseguramiento de la calidad.
Empaquetar	Agrupa archivos y directorios necesarios para el producto de software.
Liberar	Traslada el software a un ambiente en el que quede disponible para los usuarios finales.
Configurar	Ajusta parámetros para el funcionamiento del software.
Controlar	Supervisa el ambiente de producción, incluyendo métricas no funcionales.

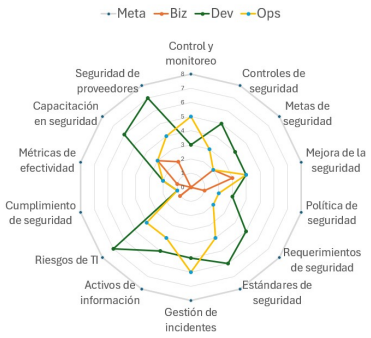


Figura 1. Cobertura de componentes BizDevOps en marcos de la industria

La industria ha intensificado su preocupación por la seguridad debido a la creciente dependencia de comunicaciones por Internet, realizadas mediante dispositivos conectados desde redes cuyas configuraciones pueden no ser confiables ni satisfacer las necesidades empresariales de seguridad. Esto genera sistemas complejos que deben considerar la seguridad desde perspectivas tanto sociales/organizativas como técnicas. En respuesta, organizaciones como ISO, Common Criteria, OWASP y NIST han desarrollado modelos, marcos de trabajo y estándares para apoyar el desarrollo seguro de software.

Con el objetivo de verificar de forma sistemática cuáles aspectos relacionados con entornos BizDevOps han sido atendidos por marcos de la industria, se realizó una comparación estructurada entre 14 de las iniciativas más utilizadas, las cuales se detallan en la *Tabla II*. El análisis sistemático de estas iniciativas revela que los aspectos más considerados son la seguridad de proveedores, la capacitación en seguridad y los riesgos de TI, seguidos por gestión de incidentes, mejora de la seguridad y estándares de seguridad, mientras que los tres aspectos menos considerados son el nivel de cumplimiento de seguridad, las métricas de efectividad y las políticas de seguridad, todos directamente relacionados con la perspectiva de negocio (Biz), evidenciando una debilidad en la relación de la perspectiva de negocio con los aspectos de seguridad.

Sin metas de seguridad establecidas, resulta difícil identificar requerimientos de seguridad en equipos con poco conocimiento en el tema [5]. La norma ISO/IEC 27014:2020 sobre gobernanza de la seguridad de la información, aunque no enfocada en desarrollo de software, proporciona guías para evaluar, dirigir y supervisar procesos de seguridad, enfatizando la necesidad de claridad en conceptos, objetivos y procesos de seguridad basados en necesidades del negocio. Para cerrar estas brechas, se propone un proceso que permita a cada organización diseñar una estrategia de desarrollo seguro de software. Esta estrategia asegura que se consideren vulnerabilidades no solo del software, sino también metas de seguridad organizacionales a lo largo de todo el ciclo de desarrollo [10], integrando efectivamente la perspectiva de negocio desde etapas tempranas del desarrollo.

III. ESTRATEGIA DE DESARROLLO SEGURO DE SOFTWARE

Con base en los hallazgos del análisis sistemático de marcos de seguridad y su enfoque en BizDevOps, descrito en la sección anterior, se propone la inclusión de una fase adicional en el flujo de trabajo BizDevOps, la cual consiste en *Asegurar*

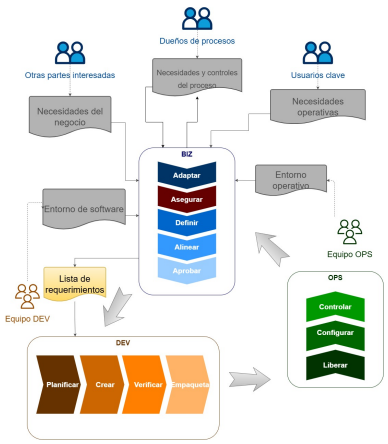


Figura 2. Propuesta de incorporación de fase Asegurar en ciclo Biz

el software mediante la integración de prácticas consolidadas de seguridad alineadas con objetivos de negocio. Esta nueva fase se ubica entre las fases *Adaptar* y *Definir*, como se muestra en la *Fig. 2*. Su ubicación es clave para asegurar la participación de personas expertas del área Biz y para que los resultados de esta fase puedan ser considerados como parte del levantamiento de requerimientos. Una actividad de este tipo favorece dos de los principios de BizDevOps [5]: la responsabilidad integral (end-to-end responsibility) de aspectos de seguridad y la estrategia de desplazamiento a la izquierda (shift-left strategy), abordando posibles problemas a partir de la prevención, no de la reacción.

Según Lohrasbinasab et al. [5], en la fase *Adaptar* se consolidan comentarios de clientes, de la empresa (incluyendo los resultados provistos por el área Ops) y del mercado, con el fin de definir las iniciativas empresariales, la hoja de ruta a seguir y los planes iniciales. Posteriormente, se encuentra la fase Definir, la cual debe partir de un modelo de solución bien entendido que incorpora los puntos de vista y necesidades de las distintas partes interesadas. Con este

Tabla II
INICIATIVAS DE SEGURIDAD DE LA INDUSTRIA

Iniciativa	Descripción
Building Security in Maturity Model (BSIMM)	Establece una línea base con actividades que deberían formar parte de iniciativas de seguridad.
Fam. ISO/IEC 27000	Incorpora la evaluación y tratamiento de riesgos y de controles de seguridad.
NIST 800-218 Secure Software Development Framework (SSDF)	Buenas prácticas para integrar seguridad en el ciclo de desarrollo de software.
NIST Cybersecurity Framework (CSF)	Directrices para gestionar y reducir riesgos de ciberseguridad.
OWASP Application Security Verification Standard (ASVS)	Gestión de seguridad con controles funcionales y no funcionales, necesarios para aplicaciones y servicios web.
OWASP Software Assurance Maturity Model (SAMM)	Analiza y evalúa la postura de seguridad, creando un programa de seguridad que muestre mejoras progresivas.
Fundamental Practices for Secure Software Development SAFECode	Recopila prácticas de seguridad para generar una cultura segura a través de un proceso de ingeniería de software seguro.
Payment Card Industry Software Security Framework (PCI SSF)	Establece requerimientos de seguridad, minimizando vulnerabilidades y ofreciendo protección frente a posibles ataques.

entendimiento, la solución puede ser dividida en componentes funcionales que determinen las distintas actividades que deben realizarse. De esta forma, al definir una solución que responda a las necesidades de la empresa, se tienen en cuenta las preocupaciones de las distintas partes interesadas. La idea de agregar la fase *Asegurar* antes de la fase *Definir* consiste, precisamente, en que, al llegar a la etapa de definición, ya se haya analizado la perspectiva de seguridad, lo que permitiría asociar los requerimientos de seguridad a los requerimientos funcionales de forma natural, considerando el entorno operativo en el que deberá trabajar el software. De esta forma, se propone la incorporación de los siguientes elementos en el flujo de trabajo BizDevOps con el fin de mejorar el proceso de desarrollo seguro de software.

III-A. Política de gestión de la deuda de seguridad

Un concepto que representa un apoyo al proceso de desarrollo seguro de software es la gestión de la deuda de seguridad (o Security Debt - SD), el cual es análogo al concepto de deuda técnica, pero hace referencia al nivel de completitud y correctibilidad de un software en términos de las características y requerimientos de seguridad necesarios para ofrecer soluciones adecuadas para el contexto operativo en que será utilizado. Una organización incurre en una SD cuando aplica soluciones que son limitadas, sea intencionalmente o no, de manera que no logra alcanzar los niveles de seguridad que son requeridos [11]. La SD se manifiesta en diversas formas, tales como vulnerabilidades identificadas pero no remediadas, controles de seguridad implementados parcialmente, configuraciones inseguras temporales que han sido perpetuadas, entre otras. A diferencia de la deuda técnica, la SD genera “intereses” en forma de incremento del riesgo de incidentes de seguridad, pudiendo generar amplias brechas de seguridad, pérdidas financieras e incumplimiento regulatorio.

En entornos BizDevOps, donde la velocidad de entrega es prioritaria, la SD puede acumularse rápidamente si las metas de negocio y las metas de seguridad no están alineadas desde etapas tempranas de desarrollo, lo cual representa la razón fundamental por la que este tema debe ser considerado como parte de la estrategia de desarrollo seguro de software. Para lograr esta alineación, las organizaciones deben definir una política de gestión de la SD que establezca, como mínimo, (i) los umbrales de aceptación (niveles máximos tolerables de SD por categoría de severidad y tipo de activo); (ii) criterios de priorización a partir de los cuales se va a ordenar el “pago” de la SD (severidad, probabilidad de ocurrencia de incidentes, esfuerzo de remediación, impacto en el negocio, entre otros); (iii) plazos para remediar la SD según la severidad; y (iv) personal responsable de la gestión de la SD, para identificar, documentar y remediar la SD. Esta política debe ser aprobada por las partes interesadas del negocio (Biz), tal y como se hace con la política de seguridad de la información, y debe ser comunicada a los equipos de desarrollo (Dev) y de operaciones (Ops), de manera que haya una base formal que permita trabajar en el desarrollo seguro de software.

Teniendo definida la política de gestión de la SD, la nueva fase *Asegurar* se compone de cinco actividades específicas: el establecimiento de metas de seguridad, la revisión de las interacciones del software con los activos de información, la identificación de la SD, la valoración de riesgos y controles

actuales y el rediseño de controles. A continuación, se explicará cada una de estas actividades.

III-B. Establecimiento de metas de seguridad

Una meta de seguridad es una forma de sugerir un problema que hay que resolver, por medio de la cual se fomenta la consideración de todas las alternativas posibles, conduciendo a soluciones de seguridad más eficaces y rentables [12]. En este proceso, se han tomado como base las metas de seguridad establecidas en el Modelo de Referencia de Aseguramiento y Seguridad de la Información (RMIAS por sus siglas en inglés [12]), las cuales se muestran en la *Tabla III* y deben ser analizadas desde una perspectiva de necesidades organizacionales. Esta actividad se divide en dos tareas: identificar las metas de seguridad y verificar su nivel de cumplimiento actual.

III-B1. Identificar metas de seguridad: Consiste en analizar la hoja de ruta y los planes iniciales resultantes de la fase previa *Adaptar*, así como la clasificación de datos organizacionales (cuando haya sido definida) con el fin de tener una perspectiva consolidada de las partes interesadas con la cual se puedan seleccionar adecuadamente las metas de seguridad que son aplicables al proyecto de software que está siendo analizado, de manera que estas metas sean consideradas en cada una de las siguientes tareas de esta nueva fase del flujo de trabajo BizDevOps.

III-B2. Verificar el cumplimiento actual: La siguiente tarea consiste en revisar los incidentes de seguridad relacionados con el proyecto de software, así como con los datos que serán utilizados, con el fin de establecer de qué forma se han incumplido las metas de seguridad relacionadas con el proyecto y cómo se solucionaron los incidentes. También incluye un análisis con respecto a si las soluciones implementadas son permanentes y aceptables o si, por el contrario, deben ser replanteadas para cumplir las metas de seguridad.

Los resultados de estas dos tareas deben ser documentados en forma concisa, con el fin de que puedan ser revisados posteriormente y sirvan como insumo a la siguiente actividad de la fase. En resumen, esta actividad recibe como insumos las iniciativas empresariales, hojas de ruta y planes iniciales, provenientes de la fase *Adaptar*, y genera como resultados las metas de seguridad aplicables para el software en análisis y el nivel de cumplimiento actual de esas metas en el contexto operativo. Estos dos resultados representan los insumos para la actividad siguiente, que se presenta a continuación.

III-C. Revisión de interacciones del software con los activos de información

En esta segunda actividad, se procederá a realizar dos tareas específicas: la primera es identificar los activos de información involucrados en el desarrollo y uso del software y la segunda consiste en relacionar las metas de seguridad con los activos de información involucrados en el proyecto de software.

III-C1. Identificar los activos de información involucrados: Esta identificación no pretende ser exhaustiva, pues no se han levantado formalmente los requerimientos, y debe basarse en el juicio experto de los equipos de desarrollo y operaciones con base en la funcionalidad que requiere la persona experta del área de negocio. Para esta actividad deben considerarse el inventario de activos de información y, bajo el conocimiento actual del software por implementar, deben identificarse los activos de información con los cuales se tendrá algún tipo de interacción. Al identificar estos activos, se podrá pensar en proporcionar seguridad no solo dentro del contexto de la aplicación, sino también considerando sus relaciones con el entorno operativo en el que se utilizará.

III-C2. Actualizar el inventario de deuda de seguridad: Como parte del proceso de desarrollo seguro de software, es necesario contar con un inventario de la SD existente, el cual consiste en documentar y analizar la SD asociada a cada activo de información, con el fin de comprender sus brechas de seguridad actuales para procurar solventarlas con el proyecto de software que está siendo analizado. Este inventario debe incluir, para cada activo de información, vulnerabilidades conocidas y sin remediar, controles de seguridad parcialmente implementados, configuraciones que incumplen las políticas establecidas, actualizaciones de seguridad pendientes y excepciones de seguridad que han sido autorizadas previamente.

III-C3. Mapear las metas de seguridad con los activos de información y la deuda de seguridad: El objetivo consiste en identificar cómo cada activo de información puede impactar las metas de seguridad seleccionadas para el proyecto de software. A partir de este conocimiento, se podrá determinar la mejor forma de implementar controles de seguridad adecuados que aseguren razonablemente el cumplimiento de las metas de seguridad y el pago posterior de la SD. La *Tabla IV* presenta un ejemplo de cómo documentar este mapeo, información que representa el insumo principal para la siguiente actividad de la fase *Asegurar*.

En resumen, esta segunda actividad recibe como insumos el inventario de activos de información, el cual corresponde a información empresarial, y las metas de seguridad prioritarias para el software, las cuales fueron generadas en la actividad anterior. El resultado es el mapeo de metas de seguridad con activos de información, el cual será uno de los insumos de la tercera actividad propuesta y que se explica a continuación.

Tabla III
METAS DE SEGURIDAD

Metas de Seguridad	Descripción
Rendición de cuentas	Capacidad de responsabilizar a los usuarios de sus acciones.
Auditabilidad	Capacidad de supervisar todas las acciones realizadas dentro de un sistema.
Autenticidad / Confianza	Capacidad para verificar la identidad y establecer la confianza en un tercero y en la información que proporciona.
Disponibilidad	Aseguramiento de que los componentes de un sistema estén disponibles y operativos al ser requeridos por usuarios autorizados.
Confidencialidad	Aseguramiento de que solo usuarios autorizados pueden acceder a la información del sistema.
Integridad	Aseguramiento de que el sistema provee información completa, precisa y ausente de modificaciones no autorizadas.
No repudio	Capacidad para demostrar la ocurrencia o no ocurrencia de un hecho o participación o no participación de una parte en un evento.
Privacidad	Aseguramiento del respeto por las leyes sobre privacidad y de la posibilidad de que las personas puedan controlar su información.

III-D. Establecimiento de la deuda de seguridad asociada al proyecto de software

La SD relacionada con el proyecto debe ser cuantificada y categorizada con el fin de integrarla eficazmente en los planes de desarrollo y operaciones de software, tal como se explica a continuación.

III-D1. Cuantificación de la deuda de seguridad actual: La cuantificación de la SD se utiliza con el fin de tener una valoración más objetiva de su estado actual. Para su inclusión en la fase *Asegurar* de BizDevOps, se propone cuantificar la SD por activo de información, de manera que, al analizar un proyecto de software, se trabaje con la SD asociada a los activos involucrados. El cálculo de la SD debe incluir la identificación de vulnerabilidades, la ponderación de los niveles de severidad (estableciendo su categorización, su valor y su peso [13]) y la antigüedad de cada vulnerabilidad, considerando que las vulnerabilidades más antiguas incrementan exponencialmente el riesgo.

La fórmula propuesta para calcular la SD en esta etapa temprana del proyecto de desarrollo de software que está siendo analizado es la siguiente:

$$SD_{\mathrm{x}} = \sum_{i=1}^{N} activoSD_{\mathrm{i}} \quad (1)$$

En esta fórmula, *X* representa el proyecto, pues se pretende conocer el estado inicial de la SD para esta iteración del proyecto; el rango *i - N* representa cada uno de los activos de información asociados al proyecto y *activoSD* representa la SD actual registrada para cada activo de información. El objetivo de calcular la SD en este punto del proyecto, en el que todavía no se han establecido formalmente los requerimientos, consiste en conocer el estado de cumplimiento actual de cada activo de información con respecto a las metas de seguridad organizacionales, así como entender sus limitaciones actuales.

III-D2. Priorización de acciones de remediación de la SD actual: Con base en la información acerca del estado actual de la SD asociada al proyecto y siguiendo los criterios de priorización para el pago de la SD establecidos en la política de gestión de la SD, se deben establecer posibles acciones de remediación de la SD que estén asociadas al proyecto, con el fin de que estas acciones sean consideradas en la fase de análisis de requerimientos. Al tener presentes estas acciones, la lista de requerimientos será más completa y el equipo de trabajo tendrá la posibilidad de incorporar componentes que ayuden a fortalecer controles que disminuyan la SD actual mientras se desarrolla la funcionalidad requerida de la aplicación.

III-E. Valoración de riesgos y controles actuales

La cuarta actividad de la fase *Asegurar* está compuesta por dos tareas específicas: identificar y valorar riesgos e identificar controles asociados.

Tabla IV
EJEMPLO DE MAPEO DE METAS, ACTIVOS Y DEUDA DE SEGURIDAD

Activo	Meta	Deuda Existente
Activo 1	Integridad	7.8 (Alta)
	Confidencialidad	4.1 (Baja)
Activo 2	Auditabilidad	7.9 (Alta)
....		

III-E1. Identificar y valorar riesgos asociados al proyecto de software: Considerando los activos de información y las metas de seguridad, deben identificarse riesgos asociados al proyecto de software y su interacción con otros sistemas. La persona gestora de riesgos de TI debería aportar conocimiento sobre riesgos conocidos y técnicas de valoración aprobadas por la alta gerencia, aplicables a riesgos emergentes. La valoración utiliza datos históricos empresariales, datos de la industria y juicio experto, respetando el apetito de riesgo institucional.

III-E2. Identificar controles asociados: Después de hacer la valoración de riesgos, se deben identificar los controles asociados, para lo cual debe incorporarse un experto en seguridad de la información que aporte conocimiento sobre el cumplimiento de metas y controles existentes, los cuales pueden ser propios del entorno empresarial o basarse en estándares internacionales como ISO/IEC 27002:2022, que agrupa controles en cuatro categorías: organizacionales, de personas, físicos y tecnológicos. Cada control identificado se valora según su cumplimiento de las metas de seguridad, considerando los activos de información involucrados. Para la valoración se deben utilizar las técnicas o herramientas aprobadas por la organización o, alternativamente, puede emplearse juicio experto. El resultado final es un mapeo que relaciona los activos de información involucrados con los controles de seguridad asociados y el resultado de su valoración por cada meta de seguridad aplicable. A partir de este mapeo, se pueden priorizar la incorporación tanto de nuevos controles como de modificaciones a los controles existentes.

La salida esperada de esta cuarta actividad de la fase *Asegurar* es una tabla que resume los resultados de la valoración de los controles según activos de información, la cual servirá como insumo principal para la última actividad, explicada a continuación.

III-F. Rediseño de controles

En esta última actividad, se realizan tareas para establecer las brechas actuales de los controles de seguridad, así como para definir guías para cerrar las brechas de cumplimiento.

III-F1. Establecer brechas actuales de los controles de seguridad: Las brechas deben establecerse comparando el nivel actual de cumplimiento de las metas de seguridad según el estado actual de los controles con las metas de seguridad aplicables identificadas en la primera actividad propuesta, llamada *Establecimiento de metas de seguridad*. Si se confirma que el entorno operativo tiene cubiertas las metas de seguridad, el proyecto de software solo debe considerar controles muy específicos a su funcionamiento. Si, por el contrario, se identifican brechas de cumplimiento, estas brechas deben ser incluidas como parte del proyecto de software, describiéndolas en forma clara y concisa para que no haya dudas con respecto a las expectativas de cumplimiento de las metas de seguridad.

III-F2. Definir guías para cerrar las brechas de cumplimiento: A partir de las brechas identificadas, se deben crear guías factibles y realistas con respecto al nivel de cumplimiento que puede esperarse en cada iteración del flujo de trabajo BizDevOps. Las guías no deben ser exhaustivas, considerando que en este punto no se ha salido del ciclo Biz,

sino que pretenden aclarar las necesidades organizacionales en cuanto a seguridad que deben considerarse en el proyecto de software, ayudando a delimitar y dirigir los esfuerzos en cuanto a la incorporación de controles de seguridad. Con este resultado, se apoyan los principios de responsabilidad integral (end-to-end responsibility) y la estrategia de desplazamiento a la izquierda (shift-left strategy) propios de BizDevOps [5] y mencionados previamente. Estas guías conformarán uno de los insumos para continuar con la fase *Definir* del flujo de trabajo BizDevOps.

III-G. Monitoreo de la deuda de seguridad

Después de estas actividades, se espera que el flujo de trabajo BizDevOps continúe como es habitual en la empresa. Nótese que, a diferencia de enfoques tradicionales que tratan la SD principalmente en fases técnicas, esta propuesta incorpora explícitamente su gestión desde la perspectiva Biz, permitiendo su alineación temprana con objetivos estratégicos y restricciones organizacionales. Siendo esta propuesta parte de una investigación en desarrollo, es esperable que se incluyan algunas otras actividades en los componentes DevOps para recalcular la SD de cada proyecto. De momento, se considera relevante dar seguimiento a la SD considerando también los incidentes relacionados con la seguridad, por lo que se deben incorporar métricas que permitan conocer el nivel de SD acumulada en cada momento. En la *Tabla V* se muestran ejemplos de métricas que podrían ser utilizadas para revisiones y análisis periódicos de los resultados y que puedan derivar en la planificación de acciones de mejora.

IV. Conclusiones

La propuesta de incluir la actividad *Asegurar* en el flujo de trabajo BizDevOps surge de la necesidad de cubrir aspectos de seguridad desde etapas tempranas del desarrollo de software. Esta propuesta se enmarca en una línea de investigación orientada al diseño y evaluación de estrategias de desarrollo seguro y se deriva de un análisis sistemático del estado del arte y del estado de la práctica. Como siguiente paso natural, el proceso será sometido a validación empírica en entornos BizDevOps reales. Si bien la propuesta se fundamenta en un análisis sistemático de marcos ampliamente utilizados, su efectividad dependerá del contexto organizacional y del nivel de madurez en BizDevOps, lo cual será abordado en futuras evaluaciones empíricas. Se espera que, durante sus primeros usos, la curva de aprendizaje genere alguna carga de trabajo y ralentice el proceso de desarrollo. Sin embargo, conforme se vaya madurando el uso de la estrategia, irá incrementando la integración de estas actividades en el flujo de trabajo y los tiempos de respuesta volverán a mejorar. Esto traerá el beneficio adicional de que el software desarrollado incluirá características de seguridad que no se consideraban previo al uso de este nuevo ciclo BizDevOps.

Como líneas futuras de investigación, se destacan tres áreas específicas: (i) La integración de métricas para validar el cumplimiento de los objetivos organizacionales de seguridad; (ii) el diseño de un marco de trabajo que incorpore esta estrategia dentro del desarrollo de proyectos de software, detallando más herramientas que apoyen la gestión de las tareas propuestas en el presente documento; y (iii) la aplicación del marco de trabajo en entornos BizDevOps reales, para evaluarlo y aplicar mejoras a partir de los resultados obtenidos.

Tabla V
Métricas de Deuda de Seguridad

Métrica	Cálculo	Meta
Puntuación total de deuda de seguridad (TSDS)	Sumatoria de toda la deuda activa	Tendencia decreciente de un sprint a otro
Deuda neta por sprint (SSD)	SD agregada menos SD pagada sprint	SSD menor o igual a 0

Agradecimientos

Este trabajo se ha desarrollado gracias al apoyo del CITIC, ECCI, PPCI, SEP y Proyecto 834-C4-157 de la Universidad de Costa Rica, así como del Grupo Alarcos de la Universidad de Castilla-La Mancha y la financiacion proporcionada por los siguientes proyectos: Proyecto Di4SPDS (CHIST-ERA grant- PCI2023145980-2) financiado por MCIN/AEI y cofinanciado por la Unión Europea, KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MCIN/AEI/10.13039/501100011033; ALBA-UCLM (TED2021-130355B-C31) financiado por CIN/AEI/10.13039/501100011033/ Union Europea NextGenerationEU/PRTR y MESIAS (2022-GRIN-34202) financiado por FEDER.

Referencias

[1] H. Villamizar, M. Kalinowski, A. Garcia, and D. Mendez: " An efficient approach for reviewing security-related aspects in agile requirements specifications of web applicacionts", en *Requir. Eng.*, vol. 25, n.4, pp. 439-468, 2020.

[2] A. Selva-Mora and C. Quesada-López: "Security Practices in Agile Software Development: A Mapping Study", en *Proc. - 2024 IEEE/ACM Int. Work. Software-Intensive Business*, IWSiB 2024, pp. 56-63, 2024.

[3] G. G. Kagombe, R. W. Mwangi, and J. M. Wafula: " Achieving Standard Software Security in Agile Developments", en *ACM Int. Conf. Proceeding Ser*, pp. 24-33, 2021.

[4] L. Williams: "Secure Software Lifecycle Knowledge Area Version", en *Cybok.Org*, 2021.

[5] I. Lohrasbinasab, P. B. Acharya, and R. Colomo-Palacios: "BizDevOps: A Multivocal Literature Review", en *LNCS Springer International Publishing*, vol. 12254, 2020.

[6] R. Kumar and R. Goyal: "Modeling continuous security: A conceptual model for automated DevSecOps using open-source software over cloud (ADOC)", en *Computers and Security*, vol. 97, 2020.

[7] R. Villalón-Fonseca: "The nature of security: A conceptual framework for integral-comprehensive modeling of IT security and cybersecurity", en *Computers and Security*, vol. 120, 2022.

[8] M. Gall and F. Pigni: "Taking DevOps mainstream: a critical review and conceptual framework", en *Eur. J. Inf. Syst.*, vol. 31, n. 5, pp. 548-567, 2022.

[9] G. Fuentes-Quijada, F. Ruiz-González and A. Caro: " Improving IT/Business Alignment in DevOps: Business Capability for Adopting BizDevOps", en *Journal of Software: Evolution and Process*, vol. 37, n. 3, p. e70016, 2025.

[10] B. Fitzgerald and K. J. Stol: " Continuous software engineering: A roadmap and agenda", en *J. Syst. Softw.*, vol. 123, pp. 176-189, 2017.

[11] J. Martinez, N. Quintano, A. Ruiz, I. Santamaria, I. Martinez, and J. Arias: "Security Debt: Characteristics, Product Life-Cycle Integration and Items", en *2021 IEEE/ACM International Conference on Technical Debt (TechDebt)*, IEEE, pp. 1-5, 2021.

[12] Y. Cherdantseva, and J. Hilton: " A reference model of information assurance and security", en *Proc. - 2013 Int. Conf. Availability, Reliab. Secur. ARES 2013*, pp. 546-555, 2013.

[13] FIRST: " Common Vulnerability Scoring System version 4.0 Specification Document", pp. 1-40, 2023.

[14] K. Rindell and J. Holvitie: "Security Risk Assessment and Management as Technical Debt", en *2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pp. 1-8, 2019.

PINO: Privacidad en Incidencias Notificadas On-Chain

Carles Anglés-Tafalla,
Jordi Castellà-Roca y Alexandre Viejo
Universitat Rovira i Virgili,
Dept. d'Eng Inf i Mat, CYBERCAT,
43007 Tarragona, Spain
{jordi.castella, carles.angles,
alexandre.viejo}@urv.cat

Jordi Soria-Comas
Universitat Oberta de Catalunya,
Est. d'Inf, Mult i Telecom,
UOC-TECH Research Centre,
CYBERCAT,
08018, Barcelona, Spain
jordi_sc@uoc.edu

Guillermo Navarro-Arribas
y Julián Salas
Universitat Autonoma de Barcelona,
Dept. d'Eng. de la Inf i les Com,
CYBERCAT,
08193 Bellaterra, Catalonia, Spain
{guillermo.navarro, julian.salas}@uab.cat

***Resumen*—El *vehicular crowdsourcing* se ha consolidado como un paradigma clave para la detección y reporte colaborativo de incidencias, generando datos compartidos geolocalizados sobre el estado de la red vial. Sin embargo, los enfoques tradicionales basados en plataformas centralizadas plantean riesgos significativos para la privacidad y la soberanía de los datos. La aparición de modelos descentralizados basados en *blockchain* ha mitigado la dependencia en entidades de confianza, pero introducen nuevos desafíos en seguridad y privacidad al hacer públicas y potencialmente trazables las contribuciones de los vehículos. Este trabajo propone un sistema de reporte de incidencias vehiculares descentralizado y respetuoso con la privacidad, combinando *Blockchain*, *Zero-Knowledge Proofs* y *differential privacy* adaptativa. El sistema garantiza autenticidad y trazabilidad sin comprometer la identidad de la persona participante, ajustando dinámicamente el nivel de perturbación de su localización. Los experimentos muestran que es posible mantener altos niveles de privacidad con una precisión aceptable en la localización agregada, lo que supone una solución prometedora para entornos vehiculares seguros y privados.**

***Index Terms*—Privacidad diferencial, Redes Vehiculares, Blockchain, Vehicular Crowdsourcing.**

I. Introducción

En la última década, el concepto de *vehicular crowdsourcing* ha emergido como un enfoque que aprovecha los datos generados por múltiples vehículos conectados para mejorar la percepción colectiva del entorno vial. En este paradigma, los automóviles actúan como nodos inteligentes capaces de recopilar, compartir o reportar información sobre accidentes, congestión o peligros en tiempo real, contribuyendo de manera activa a la generación de conocimiento compartido sobre el estado de la red vial.

Este modelo de colaboración vehicular ha inspirado la aparición de plataformas basadas en crowdsourcing como Waze[1], que permiten a los conductores compartir información instantánea sobre condiciones de tráfico y eventos en la carretera. A través de simples interacciones con la aplicación, las personas participantes pueden reportar incidentes como accidentes, atascos, obras o vehículos detenidos, generando datos que se agregan y procesan para ofrecer una visión actualizada del estado del tráfico. Más allá de las plataformas comunitarias, fabricantes de automóviles, aseguradoras y autoridades de tráfico han adoptado modelos similares mediante la transmisión periódica de datos vehiculares hacia bases de datos en la nube gestionadas de forma centralizada por terceros [1].

La recolección masiva de datos por parte de estas plataformas ha demostrado ser valiosa en aplicaciones de reporte de incidentes, mantenimiento de infraestructuras viales, análisis estadístico y forense, así como en la gestión del tráfico [2]. El uso y procesado de estos datos facilita la toma de decisiones tanto de conductores como de autoridades y servicios de emergencia, contribuyendo a una gestión más eficiente de la movilidad urbana.

A pesar de su enorme potencial, la aplicación de *vehicular crowdsourcing* introduce riesgos significativos para la privacidad y la protección de datos personales. La recopilación continua de información geolocalizada y telemétrica puede permitir inferir patrones de movilidad, trayectorias frecuentes o lugares sensibles asociados a conductores específicos [3]. Estos riesgos se intensifican en ecosistemas centralizados, donde los datos se concentran bajo el control de una sola entidad, aumentando la vulnerabilidad a accesos no autorizados, correlaciones de datos o usos secundarios no autorizados inicialmente.

Ante estas limitaciones, los modelos descentralizados basados en *blockchain* han surgido como una alternativa prometedora para mitigar la dependencia hacia entidades centrales en los sistemas de reporte de incidentes. Sin embargo, el carácter público y distribuido de esta tecnología introduce nuevos desafíos relacionados con la seguridad y privacidad: las transacciones almacenadas *on-chain* pueden permitir el rastreo sistemático de los vehículos, comprometiendo su privacidad; además, participantes deshonestos podrían intentar aprovecharse de un sistema sin una autoridad central para obtener ventaja o perjudicar a otros participantes.

Teniendo en cuenta estos desafíos, es necesario desarrollar nuevas soluciones que combinen las garantías de seguridad y descentralización propias de la tecnología *blockchain* con técnicas avanzadas de protección de la privacidad.

I-A. Estado del Arte

La preservación de la privacidad en *Location-Based Services* (*LBS*) respaldados con *vehicular crowdsourcing* ha despertado un creciente interés en los últimos años. En este ámbito, las propuestas en la literatura reciente suelen agruparse en *centralizadas* o *descentralizadas*, según su arquitectura y el grado de dependencia hacia una entidad central.

[1]Waze — https://www.waze.com

En la última década, múltiples trabajos han explorado distintas vías con la intención de evitar que proveedores de servicios centralizados accedan directamente a datos geolocalizados de las personas participantes. En [4] se propone un método colaborativo basado en *Multi-Party Computation* (*MPC*) en el que los vehículos calculan de manera conjunta su posición para LBS, evitando revelar sus coordenadas reales al servidor. En un escenario similar, PriTAEC [5] emplea *Ciphertext-Policy Attribute-Based Encryption* (*CP-ABE*) y *Oblivious Transfers* (*OT*) para ocultar a la nube las ubicaciones de los vehículos que participan en redes VANETs. En [6], [7] combinan *Homomorphic Encryption* (*HE*) con *OT* para que los vehículos soliciten información sobre incidencias viales sin revelar su ubicación exacta a terceros. En PRIS [8], lpRide [9] y PSRide [10], en cambio, aplican *Additive HE* para encontrar coincidencias en los trayectos de conductores y pasajeros sin exponer sus puntos de origen o destino al servidor central.

Aunque estos sistemas resultan efectivos para proteger datos sensibles, presentan elevados costes derivados del uso de herramientas como HE o MPC. Por otra parte, su carácter centralizado supone un único punto de fallo o ataque, además de un riesgo de confianza por parte de la entidad supervisora.

Estas limitaciones han motivado la aparición de *alternativas descentralizadas* que prescinden de entidades de confianza, distribuyendo sus operaciones entre vehículos o nodos distribuidos. Usando *MPC*, los sistemas propuestos en [11] y [12] usan protocolos de *Private Set Intersection* combinados con *HE* para que conductores y pasajeros detecten coincidencias en sus trayectos sin revelar sus ubicaciones ni depender de terceros. Aprovechando el modelo distribuido del *edge computing*, VPTS [13] utiliza las RSUs para añadir ruido, por medio de *HE* y *Differential Privacy*, a los datos agregados y garantizar la confidencialidad de los patrones de movimiento en intersecciones urbanas. Tecnologías como el Blockchain también se han incorporado en *vehicular crowdsourcing*. En [14] se presenta *PriSC*, donde los pasajeros cifran sus políticas de localización mediante *Additive HE*, y los vehículos envían sus ubicaciones cifradas a través de RSUs a una Blockchain, que verifica la elegibilidad de los participantes de forma anónima. En [15] se presenta BSDP, un framework basado en blockchain dedicado a monitorizar el tráfico colindante a zonas de parking. Las RSUs ubicadas en el *edge* usan *HE* con *Threshold Decryption* cifrar, agregar y publicar en el blockchain las ubicaciones de los vehículos participantes de forma privada.

Si bien los sistemas descentralizados mejoran la soberanía de los datos y reducen la dependencia en entidades centrales, también presentan desafíos relevantes. La distribución del cómputo entre nodos o vehículos provoca una necesidad de consenso o coordinación que puede introducir latencias adicionales. Además, ante la premisa de privacidad, muchas de estas propuestas recurren a las mismas técnicas criptográficas de agregación o cifrado homomórfico empleadas en los sistemas centralizados, heredando en el proceso sus elevados costes computacionales y de comunicación.

I-B. Contribuciones y plan del artículo

De acuerdo con las limitaciones identificadas en los sistemas de *vehicular crowdsourcing* en entornos descentralizados, este trabajo presenta una propuesta que combina blockchain, *Zero-Knowledge Proofs* y un mecanismo adaptativo basado en *Differential Privacy* para ofrecer un sistema de reporte de incidentes vehiculares descentralizado, privado y seguro.

Las principales contribuciones de este trabajo engloban los siguientes puntos:

- *Privacidad de los testigos:* Los vehículos se autentican usando credenciales verificables, garantizando que cada reporte proviene de un actor legítimo del sistema y permitiendo verificar la autenticidad de los reportes sin revelar la identidad de los conductores involucrados.
- *Privacidad adaptativa basada en densidad de reportes:* El sistema introduce un mecanismo adaptativo que ajusta el nivel de ruido aplicado a las localizaciones en función del número de reportes recibidos. De este modo, los incidentes inicialmente reportados se publican con alta perturbación espacial, acotándose progresivamente conforme más vehículos validan el mismo evento.
- *Reportes descentralizados:* El sistema permite a los vehículos reportar y publicar incidencias de forma autónoma mediante *smart contracts*, aprovechando la inmutabilidad y transparencia de la tecnología *blockchain* para garantizar la trazabilidad y verificabilidad de los eventos, sin depender de una entidad central.

El resto del artículo se estructura de la siguiente manera: la Sección II formaliza los actores y protocolos que sustentan la propuesta; en la Sección III se evalúan las garantías de privacidad y usabilidad de los datos; y finalmente, la Sección IV presenta las conclusiones y líneas futuras de trabajo.

I-C. Requisitos del sistema

Para estar acorde con las contribuciones que presenta nuestra propuesta se traducen en los siguientes detallados requisitos de Seguridad (S), Privacidad (P) y Funcionales (F):

S1) *Descentralización*: La gestión, integridad y almacenado de los reportes deben estar gobernadas por mecanismos distribuidos, evitando que una entidad central actúe como punto único de fallo o confianza.

S2) *Integridad de los reportes*: Las incidencias reportadas deben ser inmutables una vez almacenadas, garantizando que no pueden ser alteradas, eliminadas o falsificadas.

S3) *Autenticidad del vehículo reportante*: Solo los vehículos registrados por la autoridad A_R deben poder generar reportes válidos.

S4) *Resistencia a ataques Sybil*: Un atacante no debe poder crear múltiples identidades falsas para influir en la agregación de incidencias o saturar el sistema con reportes maliciosos.

P1) *No enlazabilidad*: No debe ser posible vincular los diversos reportes generados por un mismo vehículo.

P2) *Anonimato de la posición*: la ubicación desde la que se reporta una incidencia debe mantenerse protegida, y cada persona participante debe poder determinar el nivel de privacidad que desea asociar a dicha ubicación.

F1) *Utilidad*: La localización de los incidentes debe poder precisarse en función del número de veces que se reporte. Es decir, debe existir un mecanismo que permita mejorar progresivamente la precisión de la posición en aquellas incidencias que acumulen un mayor número de reportes.

II. Propuesta en detalle

En esta sección se detalla la propuesta para realizar el reporte seguro y la posterior agregación de incidencias vehiculares. En primer lugar, se describen los tres actores que intervienen en la propuesta, y a continuación se explican en detalle las distintas fases que conforman el funcionamiento del sistema:

- *Vehículo testigo (V):* Nodo participante que detecta un incidente y genera un reporte firmado y verificable. Cada vehículo dispone de una credencial verificable emitida por la autoridad de registro A_R y asociada a un par de claves criptográficas (sk_V, pk_V).
- *Autoridad de registro (A_R):* Entidad de confianza (por ejemplo, una autoridad de tráfico o un fabricante) encargada de registrar, emitir las credenciales verificables y mantener un árbol de Merkle que agrupa los identificadores públicos de los vehículos autorizados. Asimismo, A_R es responsable de desplegar y configurar el *smart contract* $RepSC$, que constituye el núcleo del sistema de gestión de reportes de incidencias.
- *Smart Contract de reporte ($RepSC$):* Componente lógico desplegado en la blockchain que almacena *on-chain* la raíz del árbol de Merkle R_V, y permite el registro y validación descentralizada de los incidentes vehiculares, eliminando la necesidad de una autoridad central o supervisión externa.

II-A. Registro del Vehículo

Este primer protocolo configura la unidad de a bordo (*OBU*) de los Vs con el fin de obtener las credenciales criptográficas para interactuar de forma segura con el resto de actores del sistema. Con este fin, la OBU genera un par de claves (sk_V, pk_V), donde la clave pública pk_V se genera a partir de una función hash segura y resistente a colisiones $\mathcal{H}(sk_V)$. Esta clave pública pk_V se usa como identificador para la credencial verificable asociada al vehículo V. A continuación, V se autentica a la A_R y, a través de un canal seguro (p.e. TLS), se realizan los siguientes pasos:

a) V envía su pk_V a A_R.
b) A_R calcula la regeneración n del árbol de Merkle a partir del conjunto de claves públicas pk_V de los vehículos registrados, generando la raíz R_V^n.
c) A_R publica la raíz R_V^n en el *smart contract* $RepSC$, de modo que sea accesible *on-chain* para posteriores verificaciones.
d) A_R envía a cada vehículo V su ruta del árbol de Merkle, denotada como $\mathrm{MerklePath}_V^n$, es decir, el conjunto de *hashes* hermanos del árbol que lo llevan hasta la raíz.
e) Con los datos recibidos, V construye su credencial verificable: $\mathsf{Cred}_V = \left(sk_V,\, n,\, \mathrm{MerklePath}_V^n\right)$.

Al finalizar el proceso, cada V implicado obtiene una credencial verificable única que puede ser empleada para demostrar, mediante *zero-knowledge proof*, su pertenencia al conjunto de vehículos registrados sin revelar su identidad.

II-B. Reporte descentralizado de incidentes

En el momento en que un vehículo V presencia un incidente, lo registra como la tupla:

$$r = (t, \tau, c, p = (\varphi, \lambda)) \tag{1}$$

donde $t \in \mathbb{T}$ es una marca de tiempo (con precisión horaria), $\tau \in \mathcal{T}$ es el tipo de incidente, $c \in \mathcal{C}$ la criticidad del incidente, y $p = (\varphi, \lambda)$ la posición. La posición se expresa en latitud y longitud, con el sistema de referencia WGS84 [16], $p = (\varphi, \lambda) \in \mathcal{L}$, tal que $\mathcal{L} \subseteq [-90, 90] \times [-180, 180]$.

Para procesar las localizaciones se utiliza una proyección al sistema métrico $\Pi : \mathcal{L} \to \mathbb{R}^2$, mediante UTM (*Universal Transverse Mercator*) dado que la extensión geográfica será relativamente acotada (p.e. a nivel de ciudad o región). Opcionalmente, V puede aportar datos adicionales (p.e. telemetría, imágenes o vídeo) del incidente, que se almacena de manera cifrada en un repositorio distribuido como IPFS, en cuyo caso el identificador de contenido (CID) se incluye en r.

Con estos datos, V genera una *non-interactive zero-knowledge proof* π_V^r, que acredita el conocimiento de los valores $(sk_V, \mathrm{MerklePath}_V^n, pk_V)$ sin revelarlos. Esta prueba garantiza las siguientes propiedades:

(I) $pk_V = \mathcal{H}(sk_V)$, i.e, V conoce la clave secreta que deriva en la clave pública registrada.
(II) $\mathsf{null}_{V,p,t}$ se ha generado a partir de $(sk_V, n, geohash(p), t)$, lo que permite identificar de forma única el evento y evitar reportes duplicados.
(III) pk_V es una hoja válida del árbol de Merkle, y el $\mathrm{MerklePath}_V^n$ reconstruye correctamente la raíz R_V^n publicada on-chain.

Una vez generada π_V, el vehículo V anonimiza su posición p aplicando una perturbación basada en *Differential Privacy*, calibrada de forma inversa a la criticidad c del evento, obteniendo así la posición perturbada $\tilde{x}$. Este proceso se detalla en la Sección II-B1.

Generados estos valores, V publica la incidencia en el Blockchain, invocando el método *submitReport()* del contrato $RepSC$ y enviando como parámetros:

$$(\tilde{x}, t, \tau, \pi_V, \mathsf{null}_{V,p,t}, [CID]) \tag{2}$$

La lógica interna de $RepSC$ realiza las siguientes operaciones para verificar la validez del incidente reportado:

a) Verifica on-chain la validez de π_V a partir de los parámetros recibidos y la raíz R_V^n, confirmando que V posee credenciales válidas, y que no ha enviado previamente un reporte correspondiente al mismo evento, identificado por $< geohash(\tilde{x}), t, \tau >$
b) Si la verificación es correcta, $RepSC$ almacena el incidente *on-chain*, junto con los siguientes atributos: $\tilde{x}, t, \tau, \pi_V, [CID]$.

II-B1. Anonimización de la posición: La posición que cada persona reporta en un incidente se protege mediante la técnica conocida como mecanismo de Laplace planar $\mathcal{M}_{\mathrm{geo}}$ que proporciona *geo-indistinguishability* [17].

Dada una ubicación p y su proyección $x = \Pi(p)$, se genera una localización perturbada $\tilde{x}$ según la siguiente densidad de probabilidad:

$$f_{\mathcal{M}_{\mathrm{geo}}(x)}(y) = \frac{\varepsilon^2}{2\pi} e^{-\varepsilon \|y - x\|_2}, \quad y \in \mathbb{R}^2.$$

El mecanismo $\mathcal{M}_{\mathrm{geo}}$ satisface ε-*geoindistinguibilidad*, es decir, para todas las ubicaciones $x, x' \in \mathbb{R}^2$ y todo conjunto medible, S se cumple:

$$\Pr[\mathcal{M}_{\text{geo}}(x) \in S] \leq e^{\varepsilon \|x-x'\|_2} \Pr[\mathcal{M}_{\text{geo}}(x') \in S].$$

Este mecanismo garantiza que la probabilidad de distinguir dos ubicaciones separadas una distancia $\|x - x'\|_2$ decrece exponencialmente con dicha distancia, proporcionando privacidad diferencial espacial con parámetro ε [17].

II-B2. Elección de ε: El sistema proporciona un marco para la elección de ε, fijando un valor máximo, y permitiendo que las personas participantes determinen el valor mínimo. Es importante tener en cuenta que este valor determinará la privacidad inicial de la localización, que se puede ver degradada al agregar el mismo incidente. Cada persona participante fijará el valor de su ε según la privacidad que desee obtener después de una estimación de su posible agregación. Dicha estimación puede ir variando durante el tiempo en función de la evolución de la densidad de tráfico o reporte de incidencias.

A continuación comentamos algunos aspectos a tener en cuenta para determinar el valor de ε.

- Número de vehículos en el sistema. Si en el sistema hay una estimación de n vehículos, una protección inicial de $\frac{\text{máx}\,\varepsilon}{n}$ fijaría una cota máxima de ε al proceso de agregación en el peor de los casos (recordamos que cada participante solamente puede reportar la misma incidencia una sola vez). Esta sería la opción más conservadora.
- Densidad estimada. Considerar todas las personas participantes del sistema puede ser una estimación poco realista. Alternativamente, podemos estimar la densidad de una localización basada en el historial de incidentes reportados. Podríamos determinar la estimación de densidad (KDE, *Kernel Density Estimation*) en función del espacio y tiempo, de manera que se aplique una ventana temporal o decaimiento a incidencias más antiguas.
- Ajuste al nivel de criticidad. La persona participante puede estar dispuesta a sacrificar su privacidad si está reportando un incidente con un nivel de criticidad muy alto. Es decir, puede determinar ε de forma proporcional al nivel de criticidad. De esta manera, obtendrá mayor privacidad al reportar incidentes poco críticos y menor privacidad en incidentes más críticos, donde la precisión de la posición puede ser de ayuda a servicios de emergencia.

II-C. Agregar incidentes

A la hora de agregar, hay que tener en cuenta que no se puede relacionar incidentes entre ellos a partir simplemente de la información reportada. En general, no se puede saber si dos incidentes reportados corresponden o no al mismo incidente real, pero sí que serán cercanos con cierto margen de error introducido por la anonimización.

Para agrupar los incidentes utilizamos un método de clustering basado en densidad relativa, como HDBSCAN (*Hierarchical Density-Based Spatial Clustering of Applications with Noise*) [18]. Al desconocer el ruido utilizado por las personas participantes es difícil establecer un umbral de distancia absoluta para agrupar localizaciones (como haríamos con DBSCAN), por lo que la distancia se determina a partir de las densidades locales de cada punto.

Dado un conjunto de localizaciones anonimizadas $\tilde{L} = \{\tilde{\ell}_1, \tilde{\ell}_2, \ldots, \tilde{\ell}_n\}$, tomamos ρ_i como la densidad local estimada en torno a $\tilde{\ell}_i$, por ejemplo, mediante el inverso del área del círculo que contiene a sus k-vecinos más cercanos:

$$\rho_i = \frac{k}{\pi\, d_k(\tilde{\ell}_i)^2},$$

donde $d_k(\tilde{\ell}_i)$ es la distancia entre $\tilde{\ell}_i$ y su k-ésimo vecino más cercano. Dos puntos $\tilde{\ell}_i$ y $\tilde{\ell}_j$ se consideran *densidad-conectados* si sus densidades son similares, es decir:

$$\frac{|\rho_i - \rho_j|}{\text{máx}(\rho_i, \rho_j)} \leq \delta,$$

para una tolerancia $\delta \in [0, 1)$. Los clusters se obtienen como las componentes conexas máximas del grafo inducido por esta relación de densidad-conectividad.

Dado un clúster $C = \{\tilde{\ell}_1, \ldots, \tilde{\ell}_k\}$ donde cada incidencia ha sido anonimizada con los valores de *epsilon* $\{\varepsilon_1, \ldots, \varepsilon_k\}$, la privacidad final de la localización se ve reducida por el teorema de composición secuencial de privacidad diferencial como:

$$\varepsilon_C = \sum_{i=1}^{k} \varepsilon_i \tag{3}$$

Esto proporciona una localización más precisa y permite situar la incidencia con mayor precisión.

Es importante remarcar que los incidentes se van reportando temporalmente, por lo que esta agrupación irá variando con el tiempo.

III. Evaluación

Como ejemplo ilustrativo hemos tomado 235 localizaciones del mismo tipo de incidencia reportadas en el área geográfica de la ciudad de Barcelona. Estos reportes se corresponden a 50 incidencias, algunas de ellas reportadas diversas veces, tal como muestra la Tabla I.

Incidencia	Número de reportes
A	100
B	50
C	25
D	10
E	5
resto de incidencias	1

Tabla I
INCIDENCIAS REPORTADAS

Las incidencias con sus posiciones reales se muestran en la Figura 1, donde podemos ver la proyección de la localización de dichas incidencias. El grosor de los puntos indica el número de reportes por incidencia.

Las personas participantes reportarían estas incidencias protegidas con el mecanismo planar de Laplace (Sección II-B1). Para simplificar hemos asumido que siempre se utiliza el mismo valor $\varepsilon = 0{,}01$. En la Figura 2 se visualizan las localizaciones reportadas.

Finalmente, al agregar estas localizaciones siguiendo la metodología descrita en la Sección II-C, podemos ver los centroides de los clústeres obtenidos en la Figura 3. Aquí el grosor del punto indica el número de puntos del clúster.

Se puede observar que los centroides de los clústeres grandes se corresponden con las incidencias con un número de reportes altos en la Figura 1.

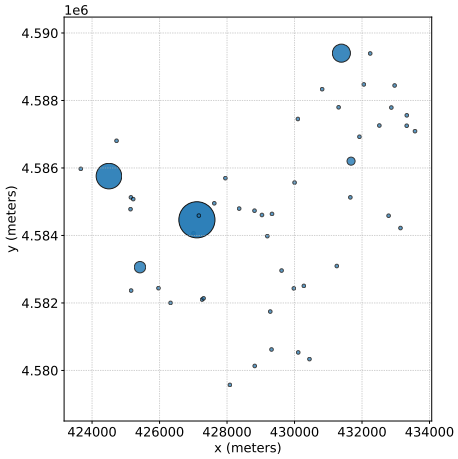


Figura 1. Incidencias reales

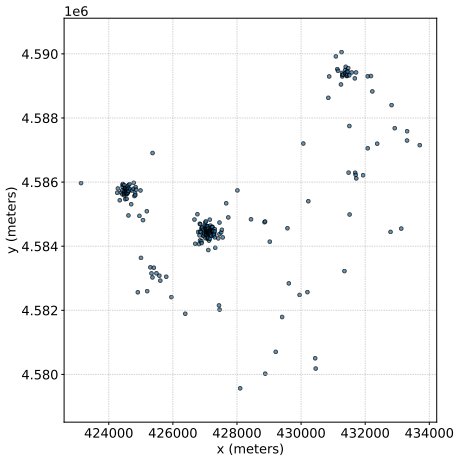


Figura 2. Incidencias reportadas (con localización protegida).

III-A. Utilidad

La utilidad de las localizaciones reportadas de forma anónima se mide cuantificando el grado en que el centroide agregado del clúster se aproxima a la ubicación real de un incidente. Una menor distancia final indica una mayor utilidad para la respuesta al incidente.

$$\mathbf{D}_{\text{Min}}(C) = \min_{i=1,\ldots,n} \|\text{Centroide}(C) - l_i\|$$

Para medir esta distancia hemos considerado la posición real de los incidentes con mayor número de reportes de la Tabla I. Para cada una hemos medido $\mathbf{D}_{\text{Min}}(C)$. Hemos considerado diferentes valores de ε, y dada la naturaleza estocástica de la anonimización hemos repetido cada una 100 veces reportando siempre el valor medio. La Tabla II muestra la distancia mínima media de los 5 clústeres mayores a su correspondiente incidencia.

Podemos ver que en general las distancias son similares y se disparan con la menor ε con una alta variabilidad. Aunque esto nos está indicando que podemos tener niveles de protección relativamente altos con una utilidad aceptable, es importante remarcar que los datos utilizados son pocos y sería necesario realizar un estudio más extenso para poder determinar mejor la influencia de ε respecto a la utilidad estimada de los datos.

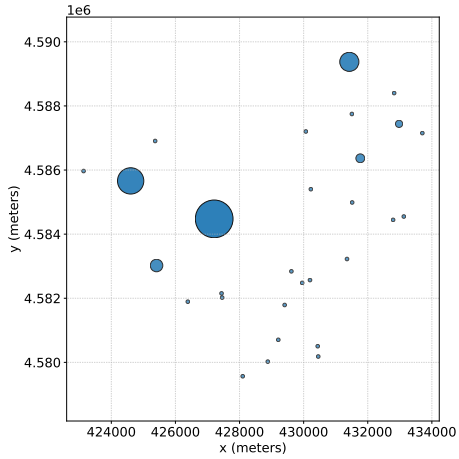


Figura 3. Incidencias agregadas

ε	Distancia media	Std
1	263.289	34.406
0.5	269.398	33.845
0.1	219.020	75.618
0.05	230.363	73.704
0.01	231.275	158.785
0.005	289.511	226.310
0.001	776.785	170.084

Tabla II
$\mathbf{D}_{\text{MIN}}(C)$ PARA LOS 5 INCIDENTES MÁS REPORTADOS.

III-B. Privacidad

El uso del mecanismo de Laplace planar asegura que la probabilidad de distinguir la ubicación real (p) de una ubicación vecina (p') decrece exponencialmente con la distancia $\|p - p'\|_2$, acotada por el parámetro de privacidad ε. Esta garantía es inherente al dato individual y no depende del conocimiento auxiliar del adversario.

Si fuéramos capaces de determinar que k reportes corresponden a la misma incidencia, el nivel de protección se degrada de acuerdo a la composición secuencial (se suma la ε de cada reporte). Ahora bien, cuando hacemos el clustering no conocemos si todos los reportes agregados corresponden al mismo incidente. Por tanto, la suma de ε es el peor caso.

Para cuantificar el impacto práctico sobre la privacidad medimos la reducción de la incertidumbre espacial correspondiente a cada reporte. Esto es, si cambiar el valor reportado por la persona usuaria por el centroide del clúster da lugar a una reducción o un incremento del ruido introducido sobre su posición original.

III-C. Discusión

Respecto a la privacidad del sistema, cada persona participante proporciona su localización con un nivel de privacidad deseado, aunque esta privacidad se verá degradada si se publican más reportes de la misma incidencia. Como se ha visto en la Sección II-B2 determinar el nivel de privacidad inicial es muy importante y debe considerarse cuidadosamente.

Nuestra propuesta protege contra adversarios pasivos globales en el caso general, siempre que exista un número de participantes y reportes mínimo. En caso contrario, se podrían producir reidentificaciones, aunque su posición no pueda llegar a ser determinada con exactitud. Dado que una

persona usuaria solamente puede reportar la misma incidencia una vez, también obtenemos protección contra atacantes internos que puedan forzar el reporte de incidencias para desanonimizar la posición de una víctima. En este sentido, el sistema proporciona una protección determinada por el nivel de privacidad inicial de cada persona usuaria. P.e., si queremos protegernos contra coaliciones de c atacantes que puedan forzar el reporte de la misma incidencia, tenemos que fijar un nivel de privacidad inicial acotado por $\frac{\varepsilon}{c}$.

También es importante remarcar que la información no protegida como el tiempo o el tipo y criticidad de la incidencia podrían dar lugar a revelación de información. En el caso del tiempo, podría protegerse introduciendo un *delay* en el envío de la incidencia; sin embargo, para que dicha protección fuese efectiva, el retraso debería ser lo suficientemente grande, lo que en muchos escenarios haría que el reporte perdiese su utilidad práctica. En cuanto al tipo y la criticidad de la incidencia, su potencial para revelar información dependerá de la granularidad con que se definan, por lo que será necesario considerarlo cuidadosamente al establecer estas categorías.

Un aspecto relevante de nuestro enfoque es que la desanonimización progresiva de la posición ocurre únicamente como consecuencia de un número elevado de personas usuarias que conocen y reportan la misma incidencia. Al mismo tiempo, este proceso incrementa el tamaño del conjunto de anonimato formado por todas las personas usuarias que han participado en el reporte. Esto no implica que la posición de una persona participante concreto no pueda verse comprometida al correlacionar incidentes previos asociados a conjuntos de anonimato más pequeños; sin embargo, sí sugiere intuitivamente que la pérdida de información individual se ve amortiguada por el crecimiento del conjunto de anonimato.

IV. Conclusiones

En este artículo hemos presentado un sistema descentralizado para reporte de incidencias en entornos vehiculares, donde se protege la localización de las personas participantes de forma adaptativa. Cada persona usuaria puede fijar un nivel de protección en su localización. Al mismo tiempo, la localización de incidentes puede desanonimizarse progresivamente a medida que dicha incidencia es reportada por mayor número de participantes. Para ello presentamos una arquitectura que integra tecnología blockchain y *geo-indistinguishability* para ofrecer un sistema de reporte vehicular descentralizado, verificable y con privacidad adaptativa. Como líneas futuras, se plantea ampliar la evaluación con escenarios más complejos y optimizar los mecanismos de agregación para reducir la pérdida de precisión sin comprometer la privacidad.

Agradecimientos

Este trabajo ha sido apoyado por los proyectos estratégicos en ciberseguridad HERMES y DANGER (C062/23), y la cátedra en ciberseguridad ARTEMISA (C057/23) financiados por el Instituto Nacional de Ciberseguridad de España a través de la Unión Europea – NextGenerationEU y del Plan de Recuperación, Transformación y Resiliencia; por los proyectos PROVTOPIA (PID2023-150098OB-I00), SECURING/CYBER (PID2021-125962OB-C31), SECURING/DATA (PID2021-125962OB-C32), SECURING/NET (PID2021-125962OB-C33), SAFE/DATA (PID2024-156914OB-C42), SAFE/BLOCKCHAIN (PID2024-156914OB-C43), SAFE/CYBER (PID2024-156914OB-C41) del Ministerio de Ciencia e Innovación, Agencia Estatal de Investigación, y European Regional Development Fund (ERDF); los proyectos SGR2021-00643, 2021SGR 00115 de la Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) de Cataluña.

Referencias

[1] S. Ebbers, S. Gense, M. Bakkouch, F. Freiling, and S. Schinzel, "Grand theft api: A forensic analysis of vehicle cloud data," *Forensic Science International: Digital Investigation*, vol. 48, p. 301691, 2024.

[2] J. Li, Z. Song, Z. Zhang, Y. Li, and C. Cao, "In-vehicle digital forensics for connected and automated vehicles with public auditing," *IEEE Internet of Things Journal*, vol. 11, no. 4, pp. 6368–6383, 2023.

[3] Z. Zhang, Z. Cao, and Y. Wang, "Forensics system for internet of vehicles based on post-quantum blockchain," *Sensors*, vol. 25, no. 19, p. 6038, 2025.

[4] S. U. Hussain and F. Koushanfar, "P3: Privacy preserving positioning for smart automotive systems," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 23, no. 6, pp. 1–19, 2018.

[5] Z. Xu, L. Wu, C. Qin, S. Li, S. Zhang, and R. Lu, "Pritaec: Privacy-preserving task assignment based on oblivious transfer and edge computing in vanet," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 4, pp. 4996–5009, 2023.

[6] Z. Tan, C. Wang, M. Zhou, and L. Zhang, "Private information retrieval in vehicular location-based services," in *IEEE 4th World Forum on Internet of Things (WF-IoT)*, 2018, pp. 56–61.

[7] C. Zhou, T. Wang, H. Tian, W. Jiang, and Z. Wang, "A top-k query scheme with privacy preservation for intelligent vehicle networks in mobile iot," *IEEE Access*, vol. 8, pp. 81 698–81 710, 2020.

[8] Y. He, J. Ni, X. Wang, B. Niu, F. Li, and X. Shen, "Privacy-preserving partner selection for ride-sharing services," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 5994–6005, 2018.

[9] H. Yu, J. Shu, X. Jia, H. Zhang, and X. Yu, "lpride: Lightweight and privacy-preserving ride matching over road networks in online ride-hailing systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10 418–10 428, 2019.

[10] H. Yu, X. Jia, H. Zhang, X. Yu, and J. Shu, "Psride: Privacy-preserving shared ride matching for online ride-hailing systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1425–1440, 2021.

[11] U. M. Aïvodji, S. Gambs, M.-J. Huguet, and M.-O. Killijian, "Meeting points in ridesharing: A privacy-preserving approach," *Transportation Research Part C: Emerging Technologies*, vol. 72, pp. 239–253, 2016.

[12] P. Hallgren, C. Orlandi, and A. Sabelfeld, "Privatepool: Privacy-preserving ridesharing," in *30th IEEE Computer Security Foundations Symposium (CSF)*, 2017, pp. 276–291.

[13] C. Zhang, L. Zhu, J. Ni, C. Huang, and X. Shen, "Verifiable and privacy-preserving traffic flow statistics for advanced traffic management systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 10 336–10 347, 2020.

[14] J. Zhang, F. Yang, Z. Ma, Z. Wang, X. Liu, and J. Ma, "A decentralized location privacy-preserving spatial crowdsourcing for internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2299–2313, 2021.

[15] C. Zhang, L. Zhu, and C. Xu, "Bsdp: Blockchain-based smart parking for digital-twin empowered vehicular sensing networks with privacy protection," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 9, pp. 7237–7246, 2023.

[16] National Geospatial-Intelligence Agency (NGA), "Department of defense world geodetic system 1984: Its definition and relationships with local geodetic systems," U.S. Department of Defense, Tech. Rep. TR8350.2, 2000, third Edition, amended 2014. [Online]. Available: https://earth-info.nga.mil/

[17] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security (CCS)*. Berlin, Germany: ACM, 2013, pp. 901–914.

[18] R. J. G. B. Campello, D. Moulavi, A. Zimek, and J. Sander, "Hierarchical density estimates for data clustering, visualization, and outlier detection," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 10, no. 1, pp. 5:1–5:51, 2015. [Online]. Available: https://doi.org/10.1145/2733381

ZWP-TRUST: Trusted Utility-Smart Telemetry Using Zero Knowledge Proofs, Reversible Watermarking, and Paillier Encryption

Tanya Koohpayeh Araghi*‡, Muhammad Yasir Masood*‡, Helena Rifà-Pous*‡, David Megías*‡,
Joan Ferré-Queralt †‡, Jordi Castellà-Roca†‡, Alexandre Viejo†‡

* Universitat Oberta de Catalunya (UOC)
UOC-TECH Research Center, Estudios de Informática, Multimedia y Telecomunicación
{tkoohpayeharaghi,mmasood,hrifa,dmegias}@uoc.edu
† Universitat Rovira i Virgili (URV)
Dept. d'Enginyeria Informàtica i Matemàtiques
{joan.ferreq,jordi.castella,alexandre.viejo}@urv.cat
‡ CYBERCAT - Center for Cybersecurity Research of Catalonia

***Abstract*—The innovation of smart grid technology poses very serious issues of data security and privacy to consumers, especially in the context of consumption data gathered by metering infrastructure. This paper proposes a high-frequency smart metering protocol that integrates homomorphic encryption, reversible watermarking, and zero-knowledge proofs to provide secure, privacy-preserving, and verifiable energy reporting. Reversible watermarking ensures the authenticity of the data while preserving the ability to fully recover the original measurements, and Paillier encryption enables aggregation directly over encrypted readings. To ensure the accuracy of meter-reported values, the protocol incorporates a Zero-Knowledge Proof (ZKP) mechanism that allows each smart meter to prove that its measurement lies within an allowed range without disclosing the actual reading. This prevents compromised meters from submitting malicious extreme values that could distort consumption data and disrupt grid operations. In general, the proposed protocol offers strong guarantees of integrity, confidentiality, and accountability while supporting the fine-grained reporting demands of modern smart grid infrastructures.**

***Index Terms*—smart grid, privacy, security, reversible watermarking, zero-knowledge proofs, Paillier encryption**

I. Introduction

The emergence of smart grid technology has reshaped how electricity is generated, distributed, and exchanged [1]. By combining power systems with advanced communication and control technologies, smart grids enable a more efficient, flexible, and resilient energy infrastructure. This evolution is driven by the growing penetration of renewable sources, the need for a stable supply, and the pursuit of sustainability and energy efficiency [4]. Beyond improving reliability, smart grids foster decentralized energy markets where consumers can actively trade for electricity.

A key enabler of this paradigm is the deployment of distributed energy resources (DERs) such as solar panels, wind turbines, and storage units located near consumption points [6]. These systems allow prosumers to both generate and consume energy, supporting bidirectional flows of power and information that make real-time trading and dynamic pricing possible [7]. A prosumer is a consumer who both consumes and produces electricity, most often using renewable energy sources such as solar panels or wind turbines. Prosumers can also store energy and inject surplus energy into the grid, becoming active energy market players as opposed to being passive consumers of energy [25]. Such interactivity enhances load balancing and the integration of renewable energy across the grid.

However, as energy trading becomes increasingly decentralized, new challenges arise concerning security, privacy, and scalability. Advanced metering infrastructure (AMI) supports real-time monitoring and optimization but also exposes detailed consumption data. Through *user profiling*, an attacker could infer private information about consumers' habits and routines. Ensuring data confidentiality is therefore vital to protect privacy and maintain trust in decentralized energy markets [9].

This paper introduces a high-frequency smart metering protocol that combines Paillier homomorphic encryption, reversible watermarking, and zero-knowledge proofs to secure and verify energy readings. The proposed protocol delivers confidentiality, integrity, and resilience for fine-grained smart grid data collection.

The rest of the paper is organized as follows: in Section II, the preliminaries on smart gird fundamentals and watermarking and cryptographic functions are explained. In Section III, recent related works in advanced metering system are described. Section IV introduces the proposed protocol and system entities. In Section V, The cryptography and data hiding fundamentals are described. The different phases of the proposed protocol are explained in Section VI. Evaluation and analysis, and comparison and discussion are described in Sections VII and VIII, and finally Section IX relates the conclusion and future work.

II. Preliminaries

This section presents the basic understanding on which decentralized energy trading is based in smart grids. We start with a definition of the smart grid itself and introduce its main elements: Distributed Energy Resources (DERs) and the place of consumers [29]. Lastly, we explain what is known as the

Advanced Metering Infrastructure (AMI) that allows the real-time, data-driven operations, which despite its advantages, raises serious security and privacy issues.

A. Smart Grid Fundamentals

A traditional power grid is taking the form of a smart grid, which is arguably defined as a modernized electrical power grid that employs information and communication technology (ICT) to detect and respond to data in an automated manner [30]. This is because this integration makes the generation, distribution, and consumption of electricity more efficient, reliable, and sustainable. The smart grid is not a technology but a web of systems, and there are two-way flows of electricity and information, which allow the introduction of new functionalities, such as real-time monitoring, dynamic pricing, and better fault handling.

B. Distributed Energy Resources (DERs)

One of the foundations of the smart grid is the spread of the DERs. DERs refer to miniature power generation/ storage devices (e.g., rooftop solar panels, mini wind turbines, battery storage) that are situated within the proximity to the point of electricity usage [29]. This decentralization enables the end-users to be consumers who consume and generate energy. Consumers are able to produce electricity that they can use themselves and feed the excess into the grid. Such an ability establishes two-way power flows, essentially altering the passivity of the traditional energy consumers and opening the door to peer-to-peer energy trading and the creation of decentralized energy markets.

C. Advanced Metering Infrastructure (AMI)

The AMI enables the flow of information in both directions required to make smart grid operations possible. AMI is an all-inclusive system of smart meters (SMs), communication networks, and data management systems that allows two-way communication between utilities and consumers [31]. SMs, the important element of AMI, are used to gather detailed and high-frequency energy consumption data (as well as generation, in the case of prosumers). This information is delivered in close real-time, and it is possible to:

- Dynamic Pricing: Implementing time-of-use rates and real-time pricing.
- Real-time Monitoring: Provide utilities with a precise view of grid conditions.
- Demand Response: Allowing utilities to manage load by incentivizing reduced consumption during peak periods.

Although privacy and confidentiality deal with the fact that the consumption data has not been revealed to an unauthorized individual, integrity addresses the fact that the metering and control information is accurate and untouched. In the further discussion, however, the threat shifts to the issues of integrity in the smart grid setting. The granular data of consumption, which AMI has accumulated, is, however, a two-edged sword. Although it is critical to optimize a grid, it may be profiled by a user, in which attackers can use the data to deduce personal information about a consumer, his or her habits, routines, and even the use of appliances. Securing this information is, hence, a serious problem with decentralized energy trading systems.

The main security concern of AMI is to make sure that no one has altered consumption data throughout the transmission, and the origin of the consumption data can be verified. The reversible watermarking has been one of the tools that pays off and addresses the issue without affecting the utility of the data [23].

D. Reversible Watermarking

Digital watermarking is essentially the process of embedding hidden information, known as a watermark, within original data to enable tamper detection, copyright protection and authentication verification. The watermark bits should be dispersed across the host data in such a way that they cannot be easily detected. The embedding process must preserve the logical integrity of the original information, and a corresponding algorithm should be capable of extracting the embedded watermark. Furthermore, digital watermarking methods do not introduce extra communication costs or processing overhead and are generally lighter than cryptographic approaches, making them suitable for resource-limited sensor devices [11].

Reversible or lossless digital watermarking refers to methods that not only verify the authenticity of protected data but also allow the exact recovery of the original sensed information. Traditional non-reversible watermarking cannot completely recover the original data once an attack has occurred. Although it introduces only slight data alterations, these modifications are permanent. Therefore, for applications that require high accuracy, such as in military, medical, or smart metering systems, non-reversible watermarking is unsuitable. Consequently, reversible digital watermarking is considered the most appropriate technique for IoT sensor networks [16].

The core strength of this approach is its perfect reversibility. After receiving the watermark data and the command for the integrity of the data, the utility can run the extraction algorithm to discard the watermark and recover the exact original consumption values [13].

This is essential for regulatory compliance and accurate billing as it prevents any long-term distortion of the foundational data. More recent research has tailored these algorithms to the smart grid environment, so that it is not only small enough to execute on the limited hardware of a standard meter, but also hardened enough to resist undetectable and non-removable tampering. Thus, through reversible watermarking, the utilities will be able to leverage the entire potential of AMI to guarantee the authenticity of the data, as well as create a two-way communication environment where they can ensure the accuracy of the consumption data as it will eventually be required in the final critical uses [15].

E. Paillier Encryption

The proposed model is based on the Paillier cryptosystem, which is a probabilistic asymmetric encryption scheme characterized by the additive homomorphic property. This special property enables certain computations to be done on ciphertexts; two ciphertexts, when decrypted, give back the sum of their plaintexts [17]. This functionality offers unique features in our proposed architecture. To begin with, it allows the lightweight SMs to concentrate on reversible watermarking only and leave the more resource-intensive

Pallier encryption to the dedicated Encryption Server (ES). Secondly, and most importantly, it allows the creation of an aggregation that preserves privacy [18]. Data Aggregator (DA) is able to calculate cumulative consumption by multiplying encrypted readings of multiple SMs without the need to decrypt them, so that the data regarding households is kept secret. Lastly, the scheme offers end-to-end confidentiality in the transmission, where the data is encrypted between the ES and the Control Center (CC), which finally decrypts the data. This process has its security based on the intractable Decisional Composite Residuality Assumption (DCRA) . The Paillier cryptosystem would make up a bottom layer together with the zero-knowledge proofs of the Billing Server (BS), which establishes the validity of the data, without showing the data, and balanced the requirement of the utility to do accurate aggregation and the requirement of consumer privacy [19].

III. Related Works

Advanced Metering Infrastructure is a technology that has radically changed the traditional electrical grids into data-driven networks. Although such a shift enables the implementation of crucial features such as demand response and remote control, it also raises complicated issues such as information protection, data integrity, and consumer privacy [34]. The consumption information that Smart Meters (SMs) gather on a fine-grained basis has the potential to depict very sensitive occupancy trends and individual habits, posing a serious risk to privacy [20]. Moreover, the data, upon which such critical operations as correct billing and grid balancing are based, should be reliably justified and guarded against malicious interference. To resolve both of these issues, integrity and confidentiality require the combined use of cryptography, and this is what contemporary texts discuss [22]. The fundamental security issue that AMI faces is the integrity of the data; that is, ensuring that the data on consumption has not been distorted in transit and that it is actually authentic. Reversible Watermarking (RW) has become a very effective method that can address this issue without hurting the usefulness of the original data negatively [23]. Techniques such as Prediction-Error Expansion (PEE) have been employed by researchers, leveraging the natural correlation between sequential meter readings to implement authentication codes directly within the data stream, particularly Hash-based Message Authentication Codes (HMACs), as seen in studies like [24] on lightweight authentication. This is the key characteristic that makes RW stand out from unidirectional approaches: having verified the integrity, it is possible to reconstruct the original meter data with 100 percent accuracy. This feature does away with the conventional trade-off between data accuracy and security fidelity, the key to billing and forensic analysis. Nonetheless, reversible watermarking is not fraud-proof against the central utility or possible third-party eavesdroppers, even though it does guarantee the integrity of the data. The threat of infringement of privacy as a result of the analysis of a detailed load profile must be investigated with regard to encryption techniques capable of maintaining functionality. The Paillier Cryptosystem is famously considered a formidable application in this field since it possesses the distinctive additive homomorphic property of a cryptography system [25]. This mathematical property permits the use of operations on the encrypted ciphertexts to be directly related to additions on the plaintext domain on decryption. This principle was modified to be used in aggregation in the Smart Grid, which was initially proposed by Garcia and Jacobs, who suggested an architecture that has SMs that encrypt their readings using a public key. The Data Aggregator can then determine the accumulated consumption by a collection of households by just multiplying the individual ciphertexts. The result of the decryption of this product is only the total value, which, in essence, hides the information about any individual home. This model is very useful in reducing the leakage of data without interfering with the computational ability of grid monitoring that is needed in grid monitoring and control systems [28]. Paillier encryption is based upon the computational hardness of the Decisional Composite Residuosity Assumption (DCRA), which offers a solid mathematical foundation to resist eavesdropping. Since reversible watermarking and Paillier encryption solve two orthogonal security concerns (integrity versus privacy), the literature implies that the two synergize well and are needed to do so. The most holistic security solutions adopt a hybrid scheme, in which data is first encrypted by Paillier to obscure consumption patterns to interceptors and a reversible watermark is added (or it can be found in the SM) to add a layer of integrity checking and authentication, still verifiable even in the encrypted space. This multi-level design can ensure data privacy as well as data integrity at every stage of the transmission channel, which meets the multifaceted and multidimensional security requirements of a smart grid of the modern world [27]. Recent studies, especially protocols like the ReWaP: Reversible Watermarking and Paillier Encryption Approach to Privacy-preserving Smart Meter [22] and similar works on Privacy-preserving protocols to high-frequency smart meters, have focused on how best these techniques can be combined to provide computationally lightweight and robust solutions that can run in resource-constrained SM environments.

IV. Proposed Model

The proposed model introduces a secure Privacy-Preserving Protocol for SMs designed to secure smart grid data through a joint application of difference expansion-based reversible watermarking and Paillier homomorphic encryption. This hybrid method aims to maintain privacy and data integrity while being computationally efficient for resource-constrained SMs that transmit readings every few seconds or minutes. In the following, the system model and each entity are introduced.

A. System Model and Entities

The system architecture consists of four main entities.

1. Smart Meters (SMs): Installed at user premises, responsible for energy measurement and initial watermark embedding. SMs can be corrupted and used to report malicious values. The typical goal of such an attack is to inject extreme (very large or very small) readings so as to significantly bias the final aggregated result and, in turn, disrupt grid operations or downstream decision processes.

2. Encryption Server (ES): A lightweight intermediary performing Paillier encryption to reduce computation on SMs.

3. Data Aggregator (DA): Collects encrypted data from multiple SMs and performs aggregation in the encrypted domain.

4. Control Center (CC): Performs decryption, watermark verification, and data aggregation analysis.

5. Billing Server (BS): The entity in charge of billing and customer invoicing. In many practical deployments the BS already collects fine-grained (per-customer) meter readings for invoicing and commercial purposes. In this extended model the BS can locally generate a cryptographic proof that a reported smart-meter reading lies within a given interval $[L, U]$ (a *zero-knowledge proof*) without disclosing the exact reading to other parties. However, this entity must not know from which exact user the reading is to further protect the privacy.

This model assumes that at most one of the entities (ES, DA, CC) could be compromised at a time, ensuring multi-layer defense even if a single layer is breached.

V. Cryptographic and Data-Hiding Foundations

In the following the cryptographic technology and type of the used watermarking scheme is explained.

A. Reversible Watermarking via Difference Expansion

Reversible watermarking ensures both authentication and integrity by embedding binary watermarks into data without loss of the original content.

Watermark Generation: A binary watermark is generated using SHA-2 hash values derived from the secret key $\mathsf{K_w}$ and timestamps.

Watermark Embedding: The watermark bit is embedded in the least significant bit of the difference between two consecutive measurements.

Watermark Extraction: After decryption, the original readings and watermark can be perfectly recovered, ensuring the accuracy and verification of the integrity of the data.

B. Paillier Cryptosystem

The Paillier encryption scheme provides additive homomorphism, allowing the DA to aggregate encrypted values directly without decryption. This ensures:

- Aggregation of consumption data without revealing individual readings.
- End-to-end confidentiality during communication.

C. Zero-Knowledge Proofs (ZKPs)

Zero-knowledge proofs allow a prover to convince a verifier that a statement is true without revealing any information beyond its validity.

Private Inputs: The prover holds a secret value, or witness, that satisfies a public statement. This value remains hidden throughout the protocol.

Public Inputs: These are the parameters and statements known to both parties, such as cryptographic commitments, encrypted values, or claimed properties to be verified.

Prover and Verifier Roles: The prover constructs a proof demonstrating knowledge of a valid witness for the public statement, while the verifier checks the proof's correctness without learning the witness itself.

Zero-knowledge proofs thus ensure verifiable correctness of computations or claims while preserving the privacy of the underlying data.

VI. Protocol Workflow

The proposed protocol consists of five sequential phases:
Primary Setup and Initialization:

- The CC distributes a secret key $\mathsf{K_w}$ and pseudo-random number generator (PRNG) seeds to all SMs via secure bootstrapping.
- Paillier key pairs are generated by the CC, while only the public key is shared with the ES.
- Each SM initializes three PRNGs to generate different random numbers for encryption and obfuscation at each timestamp.

Phase 1 — Watermark Embedding:

- Each SM embeds a watermark bit using difference expansion in two consecutive readings.
- The watermarked values are obfuscated with pseudo-random numbers.
- The readings, the watermark and the pseudo-random numbers are sent to the BS via a secure and confidential mixing protocol between the home SMs so that the BS is not able to know to whom the readings belong.

Phase 2 - Proof Generation at BS:

- BS generates a zk-proof with private inputs (readings, watermark, pseudo-random numbers) and public inputs (lower bound, upper bound). It proves that the readings are both between a certain range without revealing them.
- Sends proof back to SM.

Phase 3 - Local Encryption at SM:

- The obfuscated watermarked values are XOR-encrypted before transmission to the ES along with the proof.

Phase 4 — Paillier Encryption at ES:

- ES decrypts the XOR layer and verifies the proof.
- If correct, ES performs Paillier encryption on the watermarked data using the CC's public key.
- The ciphertexts are re-encrypted via XOR for secure transmission back to the SM.

Phase 5 — Re-encryption and Forwarding:

- SMs decrypt and re-encrypt the Paillier ciphertexts with a different PRNG key before sending them to the DA.

Phase 6 — Data Aggregation at DA:

- DA decrypts the XOR layer and aggregates the Paillier ciphertexts from all SMs homomorphically, forwarding the aggregated results to CC.

Phase 7 — Verification and Decryption at CC:

- CC decrypts the aggregated data, removes pseudo-random obfuscation, and extracts the watermark.
- If the extracted watermark matches the expected one, the data is verified as authentic; otherwise, it is flagged as tampered.

The design of the system ensures that even if any intermediary (ES or DA) is compromised, neither fine-grained user data nor valid watermarks can be reconstructed.

Figure 1 shows the details of the proposed model

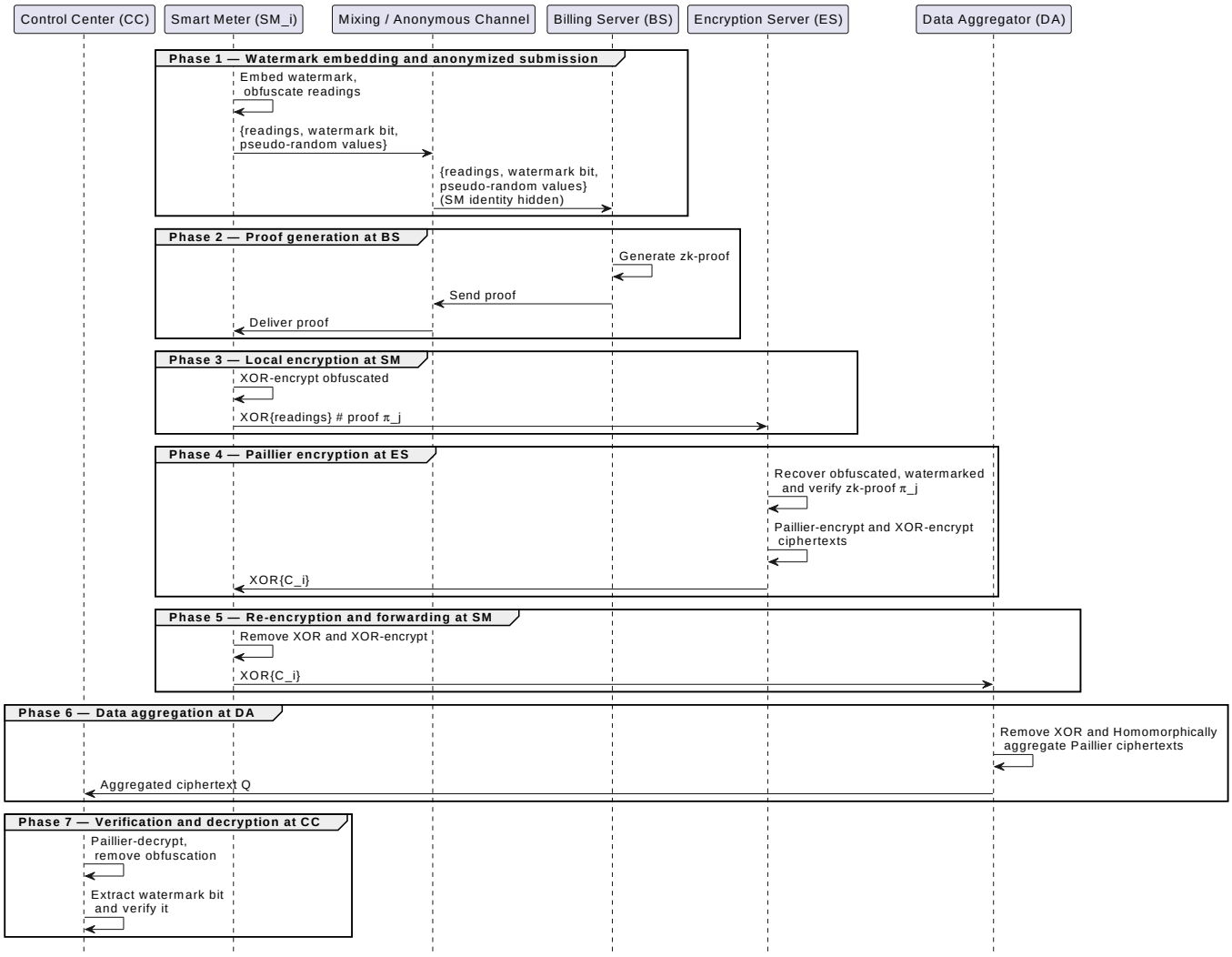


Figure 1. Protocol Phases

VII. Evaluation and Analysis

A set of theorems formally prove that the proposed protocol satisfies the essential data and privacy requirements. They are described as follows:

Privacy: Individual fine-grained data remain hidden from unauthorized entities.

Accuracy: Homomorphic aggregation guarantees that decrypted totals match the true sum of all users' consumption.

Confidentiality and Integrity: Reversible watermarking ensures tamper detection; data cannot be altered undetected.

Attack Resistance:

- *Eavesdropping and Man in the Middle (MitM) attacks:* Multi-level encryption and pseudo-random masking render intercepted data meaningless.
- *Masquerading and Replay attacks:* Unique PRNG sequences and timestamps invalidate repeated or forged transmissions.
- *False Data Injection (FDI):* Any tampering alters the watermark, triggering detection.
- *Data Poisoning:* A corrupt SM is not able to poison the aggregated sum with extreme values, the zk-proofs guarantees that the readings are between a certain range.

Table I shows the summary of defense mechanism result.

Table I
Defense Mechanism Result.

Attack Type	Defense Mechanism	Result
Eavesdropping	XOR + Paillier Encryption	Data unreadable
MITM	Watermark verification	Tampering detected
Replay	Timestamp validation	Data discarded
Masquerading	Unique PRNG seeds	Forgery prevented
FDI	Watermark mismatch	Fake data rejected
Data Poisoning	Zero-knowledge proofs	Range verified

VIII. Comparison and Discussion

In this section, the proposed model is compared with several recent models. Table II shows that the proposed model meets all the security and privacy requirements of fine-grained metering data compared to recent models.

Table II
Comparison of Security and Privacy Requirements

Methods	ReWap [22]	PFDAM [32]	EPri-MDAS [2]	NHP^3 [3]	Proposed Model
User Privacy	√	√	√	×	√
Accuracy	√	×	×	×	√
Confidentiality	√	√	√	√	√
Integrity	√	√	√	√	√

We also compared the resistance of our proposed model

Table III
COMPARISON OF ATTACK RESISTANCE

Methods	ReWap [22]	PFDAM [32]	EPri-MDAS [2]	NHP^3 [3]	Proposed Model
Eavesdropping	√	√	√	√	√
MitM	√	√	√	×	√
Masquerading	√	×	×	×	√
Replay	√	√	√	×	√
FDI	√	√	√	√	√
Data Poisoning	×	×	×	×	√

against the most common attacks that endanger the smart metering system with recent models. This comparison shows that the proposed model represents superiority in the detection and prevention of attacks compared to the other proposed models. Table III shows the comparison of the robustness of the proposed scheme with several recent methods.

IX. CONCLUSION AND FUTURE WORK

The proposed protocol provides a unified security framework for high-frequency smart that strengthens confidentiality, integrity, and verifiability in high-frequency smart metering. By combining reversible watermarking, Paillier homomorphic encryption, and ZKPs, it ensures that fine-grained readings remain private while still enabling accurate aggregation and robust tamper detection. The layered architecture enhances efficiency by delegating Paillier operations to a dedicated encryption server, reducing computational load on smart meters. Reversible watermarking guarantees integrity, authentication, and full recovery of original readings, which is essential for accurate billing and regulatory compliance. Homomorphic aggregation preserves end-to-end confidentiality while enabling accurate system-wide analytics. ZKPs generated at the billing server ensure that each reading lies within an admissible range without revealing its value, providing strong protection against data-poisoning attacks. Formal analysis confirms resistance against eavesdropping, replay, masquerading, and data-poisoning attacks, while maintaining full recoverability of original measurements for billing and regulatory needs. Future work will evaluate the scalability of the architecture in large deployments and analyse the performance overhead of ZKP generation and verification.

ACKNOWLEDGMENTS

This research is supported by Projects HERMES, DANGER and ARTEMISA, funded by the European Union NextGenerationEU/PRTR via INCIBE; by Projects PID2021-125962OB-C31 “SECURING/CYBER”, PID2021-125962OB-C32 “SECURING/DATA”, PID2024-156914OB-C41 “SAFE/CYBER” and PID2024-156914OB-C42 “SAFE/DATA”, funded by MCIN/AEI/ 10.13039/501100011033; and by Grant SGR2021-00115, funded by AGAUR, Generalitat de Catalunya. Joan Ferré-Queralt is supported by the Spanish Government under an FPU grant (ref. FPU23/01275). Alexandre Viejo is a Serra-Húnter Fellow.

REFERENCES

[1] Gellings, Clark W. *The smart grid: Enabling energy efficiency and demand response*. (2009).

[2] Zhang, Jinjiao and Zhang, Wenying and Wei, Xiaochao and Liu, Huimin. *EPri-MDAS: An efficient privacy-preserving multiple data aggregation scheme without trusted authority for fog-based smart grid*. (2024).

[3] Mohammadali, Amin and Haghighi, Mohammad Sayad. *A privacy-preserving homomorphic scheme with multiple dimensions and fault tolerance for metering data aggregation in smart grid*. (2021).

[4] Fang, Xin and Misra, Satyajayant and Xue, Guoliang and Yang, Dejun. *Smart grid—The new and improved power grid: A survey*. (2012).

[5] Araghi, Tanya Koohpayeh and Zamani, Mazdak and Manaf, A. A. and Araghi, Sagheb Kohpayeh. *An Access Control Framework in an Ad Hoc Network Infrastructure*. (2015).

[6] Liu, Chao and Zhou, Yong and Zhao, Ziang and Tian, Hui and Zhang, Yuxiang and Cai, Zhipeng. *Cyber security and privacy issues in smart grids*. (2012).

[7] Parag, Yael and Sovacool, Benjamin K. *Electricity markets design for the prosumer era*. (2016).

[8] Meng, Weiyi and Zheng, Yue and Wong, Vincent WS. *Smart grid neighborhood area networks: a survey*. (2013).

[9] Metke, Aaron R and Ekl, Randy L. *Security Technology for Smart Grid Networks*. (2010).

[10] Chen, Haiwen and Guo, Wei and Sun, Kaiqi and Wang, Xuan and Wang, Shouxiang and Guo, Luyang. *An adaptive privacy protection framework for user energy data using dictionary learning and watermarking techniques*. (2024).

[11] Araghi, Tanya Koohpayeh and Megí. *Toward the Selection of a Lightweight Authentication Technique for the Security of Smart Homes: Framework Architecture Based on a User Centric Design*. (2023).

[12] Wang, Shou-xiang and Chen, Hai-wen and Zhao, Qian-yu and Guo, Luyang and Deng, Xin-yu and Si, Wei-guo and Sun, Zhi-qing. *Preserving scheme for user's confidential information in smart grid based on digital watermark and asymmetric encryption*. (2022).

[13] Kuldeep, Gajraj and Zhang, Qi. *Multi-class privacy-preserving cloud computing based on compressive sensing for IoT*. (2022).

[14] Padma, KUV and Neelima, E. *ADVANCING IMPERCEPTIBLE SECURITY: EDWT-BASED WATERMARKING WITH BLS-HMAC ENCRYPTION*. (2025).

[15] Du, Jingqi and Dai, Chengjing and Mao, Pinshang and Dong, Wenlong and Wang, Xiujun and Li, Zhongwei. *An Efficient Lightweight Authentication Scheme for Smart Meter*. (2024).

[16] Araghi, Tanya Koohpayeh and Megí. *Evaluation and analysis of reversible watermarking techniques in wsn for secure, lightweight design of iot applications: A survey*. (2023).

[17] Zhao, Shaodong. *Smart Grids Data Aggregation Method on Paillier Homomorphic Encryption*. (2024).

[18] Qu, Bogang and Wang, Zidong and Shen, Bo and Dong, Hongli and Zhang, Xin. *Secure particle filtering with Paillier encryption–decryption scheme: Application to multi-machine power grids*. (2023).

[19] Kabir, Farzana and Araghi, Tanya Koohpayeh and Megí. *Privacy-preserving protocol for high-frequency smart meters using reversible watermarking and Paillier encryption*. (2024).

[20] Yu, Pengfei and Huang, Weicong and Li, Zhenjiang. *A Secure, lightweight, and verifiable data aggregation scheme for smart grids*. (2025).

[21] El Bouabidi, Hamid and EL Ghmary, Mohamed and Eddine Hebabaze, Salah and Amnai, Mohamed. *Enhancing Security in Edge Computing with RSA and Paillier Encryption Scheme*. (2024).

[22] Kabir, Farzana and Cabaj, Krzysztof and Araghi, Tanya Koohpayeh and Megí. *ReWaP: Reversible Watermarking and Paillier Encryption Approach for Privacy-Preserving Smart Meter*. (2025).

[23] Song, Zhen and Skuric, Antun and Ji, Kun. *A recursive watermark method for hard real-time industrial control system cyber-resilience enhancement*. (2020).

[24] Kook, Sangjin and Kim, Keunok and Ryu, Jihyeon and Lee, Youngsook and Won, Dongho. *Lightweight hash-based authentication protocol for smart grids*. (2024).

[25] Liu, Bing and Wu, Jiaming and Chai, Li. *Distributed privacy-preserving algorithm for economic dispatch and demand response of smart grid with homomorphic encryption*. (2024).

[26] Popoola, Olusogo and Rodrigues, Marcos and Marchang, Jims and Shenfield, Alex and Popoola, Jumoke and others. *Hybrid encryption for smart home healthcare: ensuring data confidentiality and security*. (2024).

[27] Bakkar, Mostafa and Bogarra, Santiago and Có. *Multi-layer smart fault protection for secure smart grids*. (2022).

[28] Li, Hongyang and Li, Xinghua and Cheng, Qingfeng. *A fine-grained privacy protection data aggregation scheme for outsourcing smart grid*. (2023).

[29] Habib, Salman. *Robust load and energy management in smart grids with Prosumer-Integrated distributed energy resources*. (2025).

[30] Alhasnawi, Bilal Naji and Hashim, Hayder Khenyab and Zanker, Marek and Bure. *The rising, applications, challenges, and future prospects of energy in smart grids and smart cities systems*. (2025).

[31] Amuthan, N and Sathya, M and Rani, Nisha C. *Empowering Consumers and Utilities for a Smarter Future: The Pivotal Role of Advanced Metering Infrastructure (AMI) in Smart Meter Technology*. (2025).

[32] Zhang, Jianhong and Wei, Jie. *PFDAM: Privacy-Preserving Fine-Grained Data Aggregation Scheme Supporting Multifunctionality in Smart Grid*. (2024).

[33] Kabir, Farzana and Megí. *RIOT-based smart metering system for privacy-preserving data aggregation using watermarking and encryption*. (2025).

[34] Sun, Jianfei and Bao, Yangyang and Qiu, Weidong and Lu, Rongxing and Zhang, Songnian and Guan, Yunguo and Cheng, Xiaochun. *Privacy-preserving fine-grained data sharing with dynamic service for the cloud-edge IoT*. (2024).

Diseño y desarrollo de un sistema de control de acceso y presencia en eventos mediante técnicas de reconocimiento facial y OCR

Sonia Solera-Cotanilla, Javier Martínez, Manuel Álvarez-Campana, Carmen Sánchez-Zas, Xavier Larriva-Novo
ETSI Telecomunicación, Universidad Politécnica de Madrid
{sonia.solera, manuel.alvarez-campana, carmen.szas, xavier.larriva.novo}@upm.es
javier.martinez.ranz@alumnos.upm.es

***Resumen*—En un mundo donde la seguridad desempeña un rol fundamental en la prevención de riesgos, el control de acceso y la gestión de la presencia en eventos son elementos clave para la detección y prevención de intrusiones. Este trabajo aborda dicha problemática mediante el diseño y desarrollo de un sistema orientado a entornos que requieren una supervisión exhaustiva y una adecuada gestión de identidades. El sistema integra mecanismos de identificación, autenticación y no repudio que permiten gestionar de manera eficaz el acceso y la permanencia de los usuarios, garantizando niveles reforzados de seguridad. Para ello, se emplea el reconocimiento facial como tecnología biométrica para la identificación de personas y el reconocimiento óptico de caracteres para la verificación de documentación, considerando, además, las implicaciones legales asociadas al tratamiento de este tipo de datos.**

***Index Terms*—control de acceso, identificación biométrica, gestión de presencia, reconocimiento óptico de caracteres**

I. Introducción

En un entorno cada vez más interconectado y con una creciente preocupación por las actuales amenazas de seguridad, resulta fundamental contar con estrategias innovadoras que garanticen la protección tanto de las personas como de los datos que manejan [1]. A medida que la tecnología avanza, estas amenazas se vuelven más sofisticadas y, por ende, los modelos tradicionales de seguridad quedan obsoletos, creando situaciones de vulnerabilidad que propician la aparición de incidentes de seguridad [2].

En la búsqueda de soluciones avanzadas que complementen los enfoques convencionales, un ámbito que ha cobrado especial relevancia en los últimos años es el control de acceso y de presencia en eventos. Cualquier usuario espera que el recinto al que acceda o el sistema con el que interactúe le brinde la seguridad y la privacidad suficientes como para despreocuparse de las posibles amenazas subyacentes.

La implementación de medidas de seguridad cada vez más estrictas puede parecer una ardua tarea para los usuarios; no obstante, supone una mejora considerable en la eficiencia y la usabilidad de los sistemas de autenticación. La adopción de estas herramientas puede transformar una tarea, inicialmente percibida como complicada, en un proceso que beneficia tanto a la seguridad institucional como a la comodidad y la confianza de los usuarios.

En este sentido, este estudio se centra en resolver una problemática real en relación con la autenticación, verificación y registro de asistencia de usuarios en entornos controlados, como punto clave para la prevención de accesos no autorizados. La pregunta de investigación que subyace al presente trabajo es: ¿Puede la combinación de diferentes tecnologías mejorar la robustez del control de acceso presencial frente a sistemas convencionales en eventos controlados?

El presente documento aborda el objetivo expuesto en las sucesivas secciones. En la sección II se exponen los antecedentes y trabajos relacionados con el estudio, los cuales sirven como base teórica para la propuesta de un sistema de control eficaz de acceso y presencia. La sección III detalla el diseño y desarrollo del sistema, que se valida en la sección IV mediante un caso de estudio específico. Al tratarse de una investigación sensible a aspectos legales, la sección V recoge las consideraciones pertinentes. Finalmente, la sección VI expone la discusión y las conclusiones del presente estudio, haciendo hincapié en los principales hallazgos y los resultados obtenidos.

II. Antecedentes y Trabajos relacionados

Con respecto al contexto de la presente investigación, es fundamental revisar los conceptos relacionados con el acceso a recursos y emplazamientos:

1. Autenticación: proceso de verificación de la identidad de un usuario. Puede realizarse mediante credenciales tradicionales, factores biométricos, tokens, etc.
2. Autorización: este proceso implica la asignación de permisos y accesos a los usuarios del sistema.
3. Registro de actividad: supone la documentación de todos los accesos a los recursos de la red, tanto los exitosos realizados por usuarios autorizados como los fallidos por no autorizados.

Estos conceptos son conocidos conjuntamente como *Authentication*, *Authorization* y *Accounting* (AAA) y se centran en asegurar registros correctos, identidades cifradas y políticas de control de acceso claras [3].

En el contexto de la evolución de los sistemas de autenticación, a lo largo de los últimos años han ido apareciendo nuevos métodos [4].

- Autenticación Biométrica Multimodal. Combina múltiples formas de autenticación biométrica para mejorar la precisión en la identificación de las personas y fortalecer la seguridad. Al integrar diferentes modalidades biométricas, como el reconocimiento facial, dactilar y de voz, se crea un sistema más robusto contra intentos de fraude, incluidos ataques sofisticados como *deepfake*.
- Two-Factor Authentication (2FA). Al requerir al menos dos formas de verificación antes de conceder acceso, el

2FA añade una capa significativa de seguridad, protegiendo contra el acceso no autorizado, incluso si alguna de las credenciales es comprometida.

- Multiple-Factor Authentication (MFA) y combinaciones de métodos. Ante la falta de un único tipo de autenticación que sea lo suficientemente robusto y barato a la vez, la tendencia actual es utilizar sistemas de autenticación mixta o combinaciones de métodos. Aportan una mayor seguridad, pero suelen ser más complejas, costosas y con un tiempo de reconocimiento considerablemente mayor.

Los controles de acceso lógicos a los sistemas de información se han ido sofisticando a lo largo de los años [5]. El uso de contraseñas y firmas como método de autenticación presenta ventajas en términos de coste y facilidad de gestión, pero conlleva riesgos significativos, por lo que se recomienda utilizar técnicas como el *hash* y *salt* para protegerlas [6].

En cuanto a los métodos basados en la pertenencia, se observa una mayor seguridad en comparación con las contraseñas. Sin embargo, este enfoque no está exento de desafíos, como el acceso múltiple, la posibilidad de cesión o robo, así como los costes asociados a la implementación y gestión de dispositivos como tarjetas inteligentes o tokens USB. En entornos de gran escala, la implementación de sistemas de control de acceso basados en roles ha demostrado ser efectiva para gestionar permisos de manera centralizada y minimizar riesgos asociados a accesos no autorizados [7].

Por otro lado, la autenticación basada en algo que se es o en rasgos biométricos ha sido objeto de numerosos avances tecnológicos y estudios académicos [8]. Las características físicas, como la huella dactilar, el reconocimiento facial y la voz, ofrecen un nivel superior de seguridad debido a su unicidad y dificultad para ser falsificadas. No obstante, su implementación implica costes elevados y complejidades técnicas, ya que requiere dispositivos especializados y algoritmos avanzados.

En este sentido, diversos trabajos han explorado el uso del reconocimiento facial como mecanismo de control de acceso en entornos físicos. Algunos diseños destacan por su eficiencia y autonomía, siendo robustos frente a cambios de iluminación y fondos complejos [9]. Revisiones generales del estado del arte en reconocimiento facial muestran tanto sus ventajas como sus limitaciones, incluyendo la necesidad de garantizar la presencialidad del usuario y la adaptabilidad a distintos entornos [10]. Por su parte, propuestas de sistemas multimodales combinan biometría con credenciales físicas, como códigos de barras, demostrando mejoras de seguridad y control de asistencia en entornos sensibles [11], [12]. No obstante, la mayoría de las propuestas se centran en entornos de alta seguridad, dejando de lado los escenarios con requisitos variables de control de acceso y registro, por tratarse de eventos puntuales.

III. Diseño y desarrollo del sistema

El sistema de control de acceso tradicional es la supervisión presencial. Al entrar al emplazamiento, los usuarios presentan una tarjeta de identificación específica y el personal de supervisión verifica manualmente la identidad. Este método asegura que se concederá el acceso solamente a las personas deseadas; sin embargo, puede resultar tedioso cuando el volumen de personas es muy elevado. Por ello, en esta sección se detalla el diseño y el desarrollo del sistema propuesto en función de la problemática descrita.

III-A. Diseño y arquitectura del sistema

Dadas las necesidades expuestas, se ha diseñado un sistema de control de acceso y presencia orientado a garantizar la autenticidad y la integridad de la información intercambiada durante los procesos de autenticación, autorización y registro. El sistema se concibe para mitigar las prácticas deshonestas más comunes realizadas por los usuarios, como la suplantación de identidad, que a su vez puede derivar en situaciones de riesgo o en vulneraciones de seguridad más graves.

Desde una perspectiva de diseño, el sistema adopta una arquitectura modular en la que cada componente aborda una parte específica del proceso de verificación de identidad. Considerando que los métodos tradicionales de autenticación son vulnerables, es crucial que la arquitectura cuente con algún mecanismo de autenticación biométrica que aporte un nivel adicional de fiabilidad. Por esta razón, la arquitectura adoptada incorpora un módulo de reconocimiento facial, seleccionado por su equilibrio entre precisión, coste y escalabilidad.

No obstante, la autenticación biométrica por sí sola no evita que los usuarios no autorizados intenten acceder de manera remota si el sistema diera esa opción. Por ello, la arquitectura cuenta con un módulo complementario de reconocimiento óptico de caracteres, cuya función es verificar la información visual presente únicamente en el entorno físico autorizado.

Finalmente, la solución incorpora un módulo de interfaz de acceso, responsable de gestionar las interacciones del usuario con el sistema. Este módulo opera como punto de entrada a la plataforma y establece la comunicación con los distintos componentes, garantizando la correcta coordinación del flujo de autenticación dentro de la arquitectura establecida.

Gracias a la modularidad de la arquitectura, cada componente del sistema funciona de manera independiente, facilitando su actualización o sustitución sin afectar al resto de los componentes. Este aspecto proporciona flexibilidad y escalabilidad a largo plazo. Además, permite a los gestores añadir o quitar funcionalidades según sus necesidades específicas. El sistema también permite seleccionar cualquiera de los tipos de autenticación descritos o una combinación de los mismos, en función del nivel de seguridad que se requiera. Además, el control de acceso a áreas restringidas se puede realizar a la entrada, a la salida o mediante un control doble en ambas.

Las tecnologías seleccionadas son Python, Flet y Firebase, por motivos de compatibilidad, escalabilidad y facilidad de integración. Python proporciona un conjunto de bibliotecas, como OpenCV, para la captura y procesamiento de imágenes; DeepFace, para la implementación de modelos de reconocimiento facial basados en aprendizaje profundo; y EasyOCR, para la extracción de texto de imágenes, lo que permite verificar la autenticidad de los códigos presentados por los usuarios. Flet se seleccionó por su capacidad de crear interfaces de usuario multiplataforma de manera rápida y escalable. Mientras que Firebase facilita la gestión de datos y la autenticación segura en la nube.

La decisión de emplear DeepFace y EasyOCR se fundamenta en su robustez, flexibilidad y existencia de implementaciones preentrenadas, lo que reduce los tiempos de desarrollo sin sacrificar la precisión. Esta elección se justifica frente a

alternativas como sistemas de desarrollo propio de modelos basados en biometría o el uso exclusivo de reconocimiento óptico de caracteres (Optical Character Recognition, OCR), que implicarían mayor complejidad, menor fiabilidad y mayores costes de mantenimiento.

La aplicación se divide principalmente en dos perfiles de usuario, que pueden identificar diferentes niveles de autorización y, por tanto, tienen funcionalidades específicas que facilitan sus tareas respectivas. El rol con mayor acceso a recursos dispone de una serie de herramientas para gestionar y verificar la identidad de los usuarios:

- Creación de eventos que requieran acceso controlado
- Configuración de controles asociados a dichos eventos
- Selección de los métodos de autenticación requeridos (facial, código, ambos o ninguno)
- Generación de códigos únicos para los usuarios con diferentes roles
- Verificación de resultados y estadísticas de acceso y control

El rol con menor acceso a recursos básicamente hace uso del sistema para pasar por los procesos de autenticación requeridos, los cuales pueden incluir la identificación facial y/o el proceso de verificación con código, según proceda.

De esta manera, se consigue un enfoque de arquitectura fiel al objetivo principal del estudio. El diseño descrito permite al usuario interactuar con el sistema de manera eficiente y segura a través de la interfaz y ofrece una alta personalización de las funcionalidades deseadas en función de los requisitos del evento.

III-B. Desarrollo del sistema

A continuación, se detallan los principales algoritmos implementados, que son el reconocimiento facial y la identificación de códigos, mostrados en la Figura 1. Ambos algoritmos están diseñados para mejorar la autenticación y el control de acceso en emplazamientos y eventos de cualquier índole.

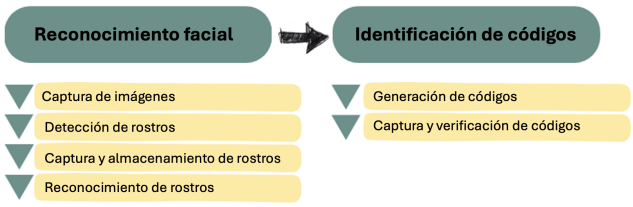


Figura 1. Flujo lógico del sistema.

III-B1. Reconocimiento facial: El sistema de reconocimiento facial se basa en el procesamiento de imágenes y en el aprendizaje profundo, implementados en Python. Este sistema ha sido diseñado para garantizar una alta precisión en la detección y el reconocimiento de rostros, incluso en diversas condiciones de iluminación y ángulos de captura. A continuación, se detalla la implementación del algoritmo:

1. Captura de imágenes: La captura de imágenes en tiempo real se realiza utilizando la biblioteca OpenCV (cv2). Este paso implica acceder a la cámara del dispositivo y obtener *frames* de vídeo, que luego se procesan para la detección de rostros.
2. Detección de rostros: La detección de rostros utiliza el clasificador Haar Cascade, una herramienta preentrenada disponible en OpenCV. El clasificador detecta regiones de la imagen que contienen rostros y devuelve las coordenadas de estos.
3. Captura y almacenamiento de rostros: Una vez detectados, los rostros se recortan y se guardan en una carpeta específica.
4. Reconocimiento de rostros: La etapa final implica el uso de DeepFace para verificar la identidad de los rostros capturados, comparándolos con una imagen base. Se hace uso de modelos incluidos en la biblioteca DeepFace, puesto que permiten trabajar con diferentes detectores, métricas y modelos.
   - Detectores: herramientas utilizadas para localizar y extraer rostros en una imagen base. Concretamente, se utilizó el detector de “OpenCV” (Haarcascades) para detectar los rostros.
   - Métricas: funciones matemáticas que cuantifican la similitud entre dos vectores de características. Las métricas disponibles en DeepFace incluyen la distancia euclidiana, el coseno y la distancia euclidiana L2.
   - Modelos: redes neuronales preentrenadas que extraen características faciales a partir de imágenes. Los modelos disponibles en DeepFace son “ArcFace”, “DeepFace”, “Facenet”, “Facenet512”, “OpenFace” y “VGG-Face”.

Para evaluar el rendimiento del sistema, se probaron distintas combinaciones de modelos y métricas, destacando “VGG-Face” con “cosine” y “Facenet512” con “euclidean”. La combinación VGG-Face + Cosine proporcionó la mayor precisión y la latencia más baja, permitiendo la verificación en tiempo real durante el control de acceso. Facenet512 + Euclidean mostró también un desempeño adecuado, aunque con menor eficiencia y sensibilidad a variaciones en gesto, ángulo o accesorios. Los falsos negativos y positivos se minimizaron bajo condiciones óptimas, especialmente con ángulo frontal, gesto sonriente y sin mascarilla, mejorando la fiabilidad del sistema. No obstante, se probaron otras variables como diferentes ángulos (frontal, lateral, desde arriba, desde abajo), gestos (serio, sonriente, triste) y accesorios (gafas, mascarilla y sin accesorios).

En concreto, DeepFace compara cada rostro capturado con la imagen de referencia que tiene de cada usuario y devuelve un resultado indicando si las imágenes corresponden a la misma persona, basándose en la métrica de distancia “cosine”. El veredicto definitivo se obtiene evaluando todos los resultados individuales de las comparaciones de los rostros capturados.

Es importante destacar que el sistema asume que los usuarios que acceden al evento están previamente registrados, por lo que el reconocimiento facial se implementa como un sistema de identificación de conjunto cerrado.

III-B2. Identificación de códigos: El sistema de identificación de códigos consta de dos partes fundamentales: la generación de códigos y la captura y verificación de los mismos. Este proceso permite que cada usuario tenga un código único asociado al evento, el cual se utiliza para validar

su identidad. Cabe mencionar que, para este algoritmo, no se ha considerado relevante indicar las partes del código más que la función encargada de la extracción del texto.

El sistema de identificación de códigos consta de dos partes fundamentales: la generación de códigos y la captura y verificación de los mismos. El organizador del evento genera códigos aleatorios de cuatro dígitos para cada usuario y se guardan en un archivo de texto. Se genera un archivo PDF para cada código (utilizando la biblioteca FPDF) y se combinan todos los archivos PDF individuales en un solo archivo (utilizando PdfMerger de PyPDF2) para facilitar su distribución.

La elección de un código de cuatro dígitos se justifica principalmente por el equilibrio que ofrece entre la complejidad y la facilidad de manejo, así como por la probabilidad de colisiones. Los códigos de tres dígitos (000-999) ofrecen solo mil combinaciones posibles, lo cual es suficiente para identificar a los usuarios. Por otro lado, los códigos de cinco dígitos (00000-99999) ofrecen cien mil combinaciones, lo cual es excesivo y puede complicar la gestión y verificación de códigos sin aportar una ventaja significativa en términos de seguridad y/o funcionalidad.

Durante el evento, el sistema captura la imagen del código mediante la cámara (con OpenCV), extrae el texto con OCR (gracias a la biblioteca easyocr) y verifica su validez frente a la base de datos, asegurando así la autenticación presencial. Para este algoritmo, no se considera relevante detallar más las partes del código que la función de extracción de texto.

IV. Validación de resultados

IV-A. Caso de estudio

Esta sección pone a prueba el sistema desarrollado mediante su aplicación a un caso de estudio real con requerimientos de control de acceso restringido a una zona delimitada a través de la gestión de identidades. En concreto, la validación del sistema se realizó durante una sesión piloto de evaluación para la obtención de una certificación profesional interna organizada por la institución. El proceso contempla las fases mostradas en la Figura 2.

Las fases de color azul son realizadas por el organizador del evento (rol con mayor acceso a recursos) y las fases de color naranja las realiza cada uno de los usuarios. A su vez, las etapas con un color más claro indican que son fases opcionales específicas de este caso de estudio. Estas fases no estarían en un evento como, por ejemplo, un congreso con control de aforo y registro de asistencia.

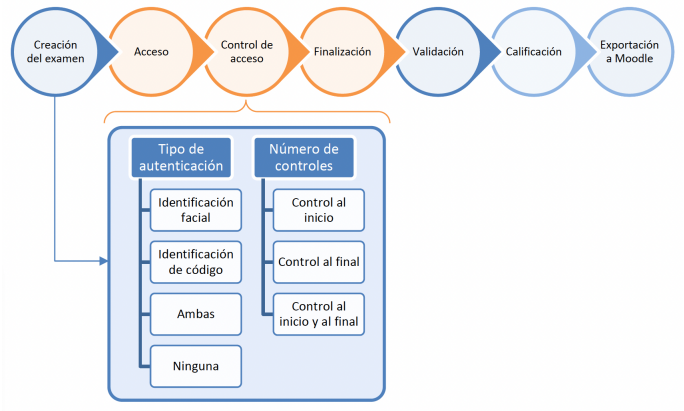


Figura 2. Fases del caso de estudio en validación.

1. Creación del evento
   El organizador crea un evento con la configuración de seguridad que desee; esto incluye identificación facial y/o código, así como otros aspectos generales del evento, como la fecha, la duración, el contenido, etc. La flexibilidad del sistema le permite ajustar el evento a la robustez que se requiere. En el caso de selección "identificación de código", se le permitirá descargar un documento PDF con tantos códigos como usuarios admitidos en el evento; los cuales serán asignados junto con la invitación del evento a cada usuario en particular; en este caso, junto al enunciado para la obtención de la certificación.
2. Preparación y acceso al evento
   Una vez creado el evento, todos los usuarios verán la información correspondiente en la vista de la aplicación.
3. Controles de acceso
   El primer paso es la identificación facial. Los usuarios toman tantas fotos de su rostro como necesiten para estar satisfechos con tres de ellas y el sistema inicia la verificación de los rostros. Las imágenes, junto con sus veredictos individuales y el veredicto definitivo del proceso de identificación facial, se almacenan en la base de datos, ocultas para el usuario.
   El segundo paso es la identificación del código. Se abre de nuevo la cámara y los usuarios toman una foto del código escrito en su hoja. De igual manera, se les da la posibilidad de repetir la captura y, cuando lo decidan, verifican el código. La aplicación verifica si el código reconocido es uno de los códigos creados para este evento. Tras lo cual, el sistema redirige a la vista de eventos.
4. Finalización del evento
   Pasado el tiempo destinado, se repiten los procesos de identificación facial y de código (si así ha sido seleccionado por el organizador); y, finalmente, el usuario puede, en este caso, entregar el examen de manera física y abandonar el emplazamiento.
5. Validación por parte del organizador
   El sistema ofrece todos los registros de los usuarios que han accedido, independientemente de si han terminado o no el proceso en el intervalo estipulado para ello.
6. Calificación
   El sistema permite a la organización corregir y asignar calificaciones de manera eficiente, con el único objetivo de ofrecer las certificaciones. Este proceso varía según el tipo de autenticación escogido (identificación facial y/o identificación por código).
   - Identificación de código: Esta opción proporciona completa transparencia en el proceso de corrección, ya que los usuarios no necesitan poner su nombre en la prueba de obtención de certificación. El organizador corrige introduciendo el código de cada prueba (escrito o mediante OCR) en la aplicación

para deliberar sobre la certificación.
   - Identificación facial o sin identificación de código: En este caso, la corrección se realiza escribiendo el correo electrónico del usuario, lo cual identifica al alumno y no es tan transparente. Esta opción también puede resultar tediosa, ya que requiere introducir manualmente el correo electrónico de cada persona. En cambio, es un proceso que, después, ahorra tiempo a la hora de subir las notas al portal correspondiente.
7. Exportación de calificaciones
   Para facilitar el proceso de subida de notas, una vez corregidas todas las pruebas y asignadas las calificaciones, el sistema permite exportar un archivo de Excel con las notas. Este archivo se puede importar en la sección de calificaciones del curso o asignatura, simplificando la actualización de las calificaciones en la plataforma educativa. En este caso, se ha realizado la validación con la plataforma educativa Moodle.

IV-B. Metodologías de validación

Para validar la eficacia del sistema, se han implementado diferentes metodologías de prueba que aseguran una evaluación exhaustiva y objetiva del mismo. Estas son:

- Pruebas unitarias para evaluar de manera independiente cada módulo.
- Pruebas de integración para evaluar la correcta comunicación entre los componentes.
- Pruebas de rendimiento para evaluar la respuesta del sistema en diferentes condiciones.
- Pruebas de robustez, simulación de fallos y propuesta de soluciones.

Las pruebas unitarias permiten verificar la funcionalidad de componentes individuales del sistema, como los algoritmos de reconocimiento facial y de código. Estas pruebas se enfocan en asegurar que cada módulo y función opere correctamente de manera aislada. Para el reconocimiento facial, se probaron variaciones en el ángulo de la cara, el gesto o la presencia de accesorios como gafas o mascarillas. Además, se verificó que el reconocimiento óptico de caracteres podía identificar correctamente los códigos en diversas condiciones de calidad de imagen y tipos de fuente.

Por otro lado, también se llevaron a cabo las pruebas de integración mencionadas para garantizar que los diferentes módulos del sistema trabajaran juntos de manera que la aplicación estuviera bien cohesionada. Se evaluó la interacción entre el módulo de autenticación, la base de datos y la interfaz de usuario mediante la simulación completa del caso de estudio. Durante estas pruebas, se verificó que los datos se transfirieran correctamente entre los módulos y que las respuestas del sistema fueran las esperadas. Además, se evaluó el rendimiento del sistema para manejar múltiples usuarios simultáneamente, asegurando que la aplicación fuera robusta y escalable.

IV-C. Resultados y validación

Los resultados obtenidos tras poner a prueba el sistema en un caso de estudio práctico real arrojan información contrastada con el objetivo propuesto al inicio del artículo. En concreto, se valida la efectividad del sistema frente a la suplantación de identidad. La validación no solo verifica la funcionalidad básica, sino también la experiencia del usuario y la robustez frente a posibles fallos.

Las pruebas simulan situaciones reales para asegurar que el sistema responde adecuadamente bajo diversas condiciones y tienen en cuenta posibles fallos del sistema para garantizar la robustez y fiabilidad de la solución propuesta. Los principales tipos de fallos están relacionados con la identificación facial (por iluminación o ángulo), con la lectura de códigos (por calidad de impresión o captura), con la autenticación (por contraseñas incorrectas) y con problemas de interfaz de usuario (por usabilidad e interacción). En todos los casos, el sistema proporciona retroalimentación al usuario y permite reintentos o asistencia técnica.

V. Consideraciones legales

La protección de la privacidad y de los datos personales constituye un elemento fundamental en el diseño y desarrollo del sistema propuesto. El tratamiento de información sensible en entornos de evaluación exige el cumplimiento estricto del Reglamento (UE) 2016/679, relativo a la protección de datos personales (RGPD) [13], y, en su caso, de la Ley Orgánica 3/2018 de Protección de Datos Personales y garantía de los derechos digitales (LOPDGDD) [14]. A pesar de esta normativa existente, diversos estudios señalan que, en la práctica, la información personal no siempre se gestiona conforme a los niveles de protección requeridos.

En el contexto de este sistema, y según el artículo 9 del RGPD, los datos biométricos destinados a identificar de manera unívoca a una persona se consideran categorías especiales de datos y están sujetos a condiciones estrictas para su tratamiento. El antiguo Grupo de Trabajo del Artículo 29 (WP29), antecedente del Comité Europeo de Protección de Datos (CEPD) [15], ha subrayado que estos sistemas deben cumplir criterios reforzados de transparencia, proporcionalidad y minimización del tratamiento.

Además, diversas autoridades de protección de datos de la UE han emitido resoluciones en las que se restringe o prohíbe el uso del reconocimiento facial cuando las organizaciones no pueden acreditar el cumplimiento de dichas normativas. Por tanto, con el fin de garantizar su cumplimiento y proteger adecuadamente los derechos de los usuarios, se adoptaron las siguientes medidas:

- Comunicación clara y previa de la finalidad del tratamiento, especificando el uso de las imágenes en los procesos de acceso, verificación y control de permanencia.
- Obtención del consentimiento explícito, informado, libre y verificable para el tratamiento de datos biométricos, conforme al artículo 9.2.a del RGPD.
- Provisión de una alternativa equivalente que no requiera el uso de dichas tecnologías, sin afectar la equidad, la duración ni la dificultad de la evaluación.
- Aplicación de los principios de minimización y limitación del plazo de conservación, restringiendo el almacenamiento de las imágenes a un máximo de un mes antes de su eliminación.
- Almacenamiento cifrado de los datos y empleo de medidas de seguridad técnicas y organizativas adecuadas,

incluyendo control de acceso, pseudonimización y registro de operaciones de tratamiento, conforme al artículo 32 del RGPD.

Estas medidas tienen como objetivo asegurar no solo el cumplimiento normativo, sino también la confianza, la transparencia y la protección efectiva de los usuarios durante el uso del sistema. En el caso que concierne a este trabajo, los patrones faciales utilizados para la identificación se almacenan cifrados en la base de datos, garantizando que no se conserven en claro.

VI. Discusión y Conclusiones

Se ha desarrollado un sistema integral para la gestión del acceso y la permanencia de usuarios en emplazamientos que tiene un control de seguridad específico. Este sistema tiene como objetivo principal proporcionar una herramienta robusta y eficiente que permita prevenir y detectar delitos relacionados con la suplantación de identidad.

Como fortaleza del estudio se encuentra la integración de múltiples métodos de autenticación, incluyendo la identificación facial y por código. Esta dualidad mejora la seguridad del sistema y permite adaptarse a diferentes escenarios. La identificación facial asegura que el usuario presente es el registrado, mientras que la identificación por código confirma que el acto se lleva a cabo de manera presencial y en el entorno delimitado con acceso controlado. Además, la arquitectura modular del sistema facilita el mantenimiento y la escalabilidad, permitiendo futuras mejoras sin rediseñar la arquitectura del sistema.

En concreto, en el caso de estudio con el que se ha decidido validar el sistema, se permite prevenir el fraude y garantizar la confidencialidad del usuario identificado. Esto constata que adoptar medidas de seguridad puede resultar incómodo, pero si se aplican correctamente, son capaces de contribuir positivamente a valores transversales en cualquier ámbito, como la equidad, la integridad y la transparencia.

El sistema ha sido puesto a prueba en un caso real de simulación de una evaluación de certificación profesional, donde finalmente se ha identificado un posible caso de suplantación de identidad. Durante esta simulación, el sistema logró detectar la discrepancia entre las imágenes faciales capturadas y las registradas previamente, evitando así un intento de fraude. Este tipo de metodologías de validación ha demostrado la efectividad del sistema en contextos reales.

No obstante, el sistema también presentaba inicialmente algunas debilidades que, en el mismo proceso de validación, han sido abordadas en diferentes iteraciones del desarrollo del prototipo. Las debilidades identificadas en el sistema no suponen factores críticos que comprometan el funcionamiento continuado del mismo.

Si bien este sistema ha sido validado con un caso de estudio basado en el entorno académico, su aplicabilidad se extiende a otros contextos donde el control de acceso y la validación de identidad son cruciales. En eventos de alta seguridad o en entornos laborales, como conferencias o reuniones corporativas, la identificación de empleados mediante métodos biométricos puede mejorar la seguridad y la eficiencia en la gestión del acceso a instalaciones sensibles, como lo constatan algunos de los trabajos mencionados. El sistema también puede adaptarse a escenarios de salud, como hospitales o clínicas, donde es crucial asegurar que solo el personal autorizado tenga acceso a áreas específicas, protegiendo así la privacidad de los pacientes. Otro posible servicio que puede beneficiarse de este sistema es la justicia, al aplicar el control de acceso a salas de audiencias o áreas restringidas. El sistema también tiene un potencial significativo en el ámbito del ocio, como en conciertos o eventos deportivos, para evitar la reventa ilegal de entradas.

Con respecto al diseño del sistema, otra posible extensión es convertirlo en reactivo en lugar de proactivo. Se podría integrar con mecanismos de control de acceso físico, como puertas automáticas que solo se abren si la autenticación es exitosa, o alarmas que se activan ante intentos de acceso no autorizados. Este conjunto de aplicaciones en distintos sectores subraya la versatilidad de la solución propuesta.

Referencias

[1] Yadigar N. Imamverdiyev: “Analysis of cybersecurity problems in process control systems”, en *Problems of Information Technology*, 2021, n. 2 pp 16-29

[2] M. Tissera1, S. Thelijjagoda1, J. Goonathilake2: “User-centric privacy preservation solution to control third party access in digital databases”, en *International Journal of Advances in Engineering & Technology*, 2017, vol. 10, n. 1 pp 30-45

[3] C. A. Ardagna, M. Cremonini, S- De Capitani di Vimercati y P. Samarati: “A privay-aware access control system”, en *Journal of Computer Security*, 2002, vol. 16, n. 4

[4] D. Shevchuk, O. Harasymchuk, A. Partyka, and N. Korshun: “Designing Secured Services for Authentication, Authorization, and Accounting of Users Cybersecurity Providing” en *Information and Telecommunication Systems*, 2023, pp 207-225

[5] M. Jong-Sik, L. Im-Yeong: “A Study on Ticket-Based AAA Protocol for Mobile User Authentication”, en *Korea Information Processing Society*, 2006, pp 987-990

[6] H. A. Hasan, H. F. Al-Layla, and F. N. Ibraheem: “A Review of Hash Function Types and their Applications” en *WJCMS*, 2022, vol. 1, no. 3, pp. 75–88

[7] A. Schaad, J. D. Moffett y J. Jacob: “The Role-Based Access Control System of a European Bank: A Case Study and Discussion”, en *Proceedings of the Sixth ACM Symposium on Access Control Models and Technologies*, 2001, pp. 3–9

[8] R. Ryu, S. Yeom, D. Herbert, and J. Dermoudy: “The design and evaluation of adaptive biometric authentication systems: Current status, challenges and future direction” en *ICT Express*, 2023, vol. 9, pp. 1183-1197

[9] H. Lee, S.-H. Park, J.-H. Yoo, S.-H. Jung y J.-H. Huh: “Face Recognition at a Distance for a Stand-Alone Access Control System”, en *Sensors*, 2020, vol. 20, n. 3

[10] P. Kaur, K. Krishan, S. K. Sharma y T. Kanchan: “Facial-recognition algorithms: A literature review”, en *Medicine, Science and the Law*, 2020, vol. 60, n. 2

[11] J. L. Center: “Practical Application of Facial Recognition: Automated Facial Recognition Access Control System”, en *Face Recognition (Springer Berlin Heidelberg)*, 1998, pp. 402–411

[12] A. M. Ayub, R. Kolandaisamy y K. H. Keoy: “Getting Smarter with Fatrix: A Facial Recognition Access Control System”, en *Proceedings of the 2023 IEEE 3rd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA)*, 2023, pp. 149–153

[13] Parlamento Europeo y Consejo de la Unión Europea, “Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos (RGPD)” Diario Oficial de la Unión Europea, L 119, 2016.

[14] Jefatura del Estado, “Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales”, Boletín Oficial del Estado, núm. 294, 2018.

[15] Grupo de Trabajo del Artículo 29 (WP29), “Dictámenes y orientaciones sobre protección de datos”, Comisión Europea, 2018.

Helm-ET: Reducing Exposure to Lateral Movement in Kubernetes Artifacts

Jacopo Bufalino
CNAM, France
Aalto University, Finland
jacopo.bufalino@lecnam.net

Jose Luis Martin-Navarro
Aalto University, Finland
Universitat Politècnica de València, Spain
jose.martinnavarro@aalto.fi

Aleksi Peltonen
CISPA Helmholtz Center for Information Security, Germany
aleksi.peltonen@cispa.de

Tuomas Aura
Aalto University, Finland
tuomas.aura@aalto.fi

***Abstract*—Este es un resumen extendido de la publicación [1]. Modern cloud applications consist of containerized microservices deployed to a virtual computing environment. Policies are needed to block attackers from using discovery and lateral movement to exploit the infrastructure. However, interactions between the components are not clearly defined, and enabling them requires manual inspection of the configuration and source code of the applications. This paper proposes a novel approach to creating network policies, based on the principles of modularity and information hiding, and the identification of service composition patterns. We implemented out solution as an open-source tool, Helm-ET, which we evaluate on 451 applications across three datasets. The results show that the proposed approach can significantly reduce the internal attack surface (92.71% less allowed connections), achieving comparable results to state-of-the-art tools. Still, compared to other solutions, Helm-ET is faster (<100ms vs 79 seconds) and applies the policies before the application deployment.**

***Index Terms*—Kubernetes, Helm, Service Composition, Network Policies, Security, Microservices**

I. Introduction

Modern cloud-native infrastructures are built as collections of independently deployable microservices that communicate through application programming interfaces (APIs) [2]. Similarly to how software libraries provide reusable, self-contained components, microservices are integrated to compose intricate applications. Cloud platforms, such as Kubernetes [3], maintain the state of the containerized services based on declarative specifications, which are also modular and reusable. Deployment tools like Helm [4] introduce an additional abstraction layer with a template language for automating and customizing the application deployment and configuration.

Despite advances in cloud infrastructure technologies, recent research [5, 6, 7] shows that declarative infrastructure and application configurations often do not follow best security practices. One of the core challenges in securing cloud platforms is applying network isolation, which is a technique for limiting lateral movement and attacker capabilities at the network layer [8, 9]. While physical networks are often highly segmented, virtual cloud environments frequently have all services running in one open network with no internal boundaries. remains a challenge.

In this paper, we develop a new policy configuration method to restrict access between microservices in the Kubernetes cluster using the natural boundaries of the declarative application specification. To achieve this goal, we employ a three-stage study. First, we review publicly available cloud applications and identify their network access patterns. Building upon these patterns, we define access rules to isolate the applications. Second, we implement a policy generation process based on the service composition patterns. Finally, we evaluate the solution by comparing it with state-of-the-art policy generation tools and with user-defined policies.

Our main contributions are the following:

1) We propose a new methodology to automatically generate network policies based on the naturally occurring composition patterns of cloud applications.
2) We introduce Helm-ET, an open-source implementation of the proposed approach for Helm declarative specifications.
3) We conduct a quantitative and qualitative study of our methodology using 451 publicly available Kubernetes applications and large benchmark applications. We compare the results against other state-of-the-art policy generation tools and show that our solution achieves considerable improvements.

In this extended abstract we include the background information on cloud technologies (Section II), the implementation (Section III). and the security evaluation (Section IV). We refer the reader to the full version of this work for more details on the proposed solution, the related work, motivation, performance evaluation and extended discussion [1].

II. Background

This section provides an overview of Kubernetes and its networking model, the Helm package manager, and

the concepts of modularity and information hiding as used in this paper.

A. Kubernetes and its networking model

Kubernetes [3] is an open-source orchestration tool for managing containerized applications. Its primary virtual compute units are *Pods*, which comprise one or more containers. They are deployed into a virtual environment called *cluster*. Pods are transient entities with dynamic names and addresses. These resources can be grouped into *namespaces* for project management and resource allocation purposes. Every Kubernetes resource may also be given *labels*, which are key-value pairs. An abstraction called *Service* is used to give permanent names and, optionally, static IP addresses to sets of compute resources. Microservices in the cluster discover each other by resolving the Service name using DNS.

The Kubernetes networking model creates a *flat virtual cluster network* where, by default, all Pods can connect to each other and all Services with IP-based protocols. The easy integration of microservices over the flat cluster network is arguably one of the reasons behind the success of Kubernetes [10]. As a result, developers do not need to include network configuration for the application to work.

B. NetworkPolicy resource

Network-layer access between the compute resources in the Kubernetes cluster network can be restricted by defining a native Kubernetes resource called *NetworkPolicy*. It is typically used for network segmentation, i.e., isolating applications from each other, but it can also be used for more fine-grained access control policies between microservices. The network policy is expressed as *ingress* or *egress* rules on the individual Pods or collections. The allowed connections in these rules can be defined based on namespaces, labels, IP addresses, and port numbers.

The NetworkPolicy resource has several limitations. For instance, policies are defined on the network and transport layers and cannot be filtered by domain names or HTTP headers. Rules identify Pods based on labels but Services based on their IP addresses, which creates a confusing mixture of two different identifier spaces. Furthermore, only *allow* rules can be specified. The policies are additive, meaning that when multiple allow rules exist for the same Pod, all those connections are allowed. Somewhat unexpectedly, when there are no ingress or egress rules for a Pod, all traffic is allowed in that direction. Only when the first ingress or egress rule is added, the default policy for the specific Pod and direction changes to deny. These limitation affect the complexity of developing policies incrementally or composing policies from different sources.

III. Implementation

This section explains the policy generation process and introduces Helm-ET, its open source implementation.

A. Policy generation process

The policy generation process takes a Helm Chart as input and produces a set of Kubernetes network policies and a configuration file as the output. The process has three steps.

Step 1 – Resource enumeration and nodes generation
We start by enumerating the dependencies, the compute resources, the corresponding Services, labels and selectors. We derive the dependency information from the Helm Chart structure and use selectors and labels to link compute units to services.
The outcome of this process is a list of nodes for the dependency and configuration graphs.

Step 2 – Dependency and Configuration graph construction
This step generates the dependency graph from the chart specifications and the connectivity graph based on the application configuration and environment variables. CC denotes the set of all components included in the chart. For every component in $c \in CC$, we collect the list of ancestors A_c and descendants D_c in the component graph. We generate the Configuration graph from environment variables, container information, ConfigMaps and Secrets objects. We match the variables with services and compute unit names. These connections form the edges in the Configuration graph.

Step 3 – Connectivity graph generation
The dependency and configuration graphs are merged into the final connectivity graph. The final connectivity graph contains the union of the edges of both graphs. This graph is used to generate the network policies in the next step.

Step 4 – Policy generation
The detailed process is implemented as follows:

1) The policy generation requires as an input the chart and its dependencies CC, the mappings for the ancestors A and descendants D, calculated in Step 1, and the Configuration graph from Step 2 (lines 1-4) .
2) The first policy created for every chart is the default deny rule $Allow(c, \emptyset)$ (line 9).
3) *Allow*(c,c) enables access from each chart to itself to allow computing units within the same chart to communicate (line 10).
4) The following rules are applied to the descendants D_c (lines 11-16) and ancestors A_c (lines 17-21) of every chart component c:
   a) Access is allowed from the chart to its descendants (*AllowEgress*), both to the compute units and the services (lines 13-14).
   b) Mirroring rules allow ingress (*AllowEgress*) from the compute units of the ancestor to the chart c (line 19).

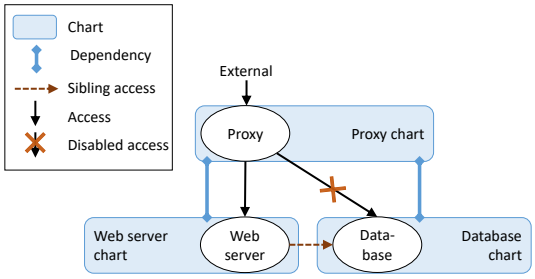
5) Default Egress connections are allowed to DNS port 53 in the `kube-system` namespace, *ClusterDNS*, and addresses outside the cluster, *Internet* (lines 22-23).
6) Finally, rules from the Configuration graph are computed (lines 24-30). All edges are included in the policy (line 28), but only Egress is required to access the Kubernetes APIs (lines 25-26).

The set of rules allowing outbound connections, is a necessary tradeoff between limiting connectivity and avoid breaking applications. For example, microservices use DNS to discover each other in the cluster. Many applications also need access to the Internet and other external networks for security updates, software license validation, or similar functionality. As the focus of this paper is internal attack surfaces, selectively disabling the outbound access is outside the scope of the current discussion.

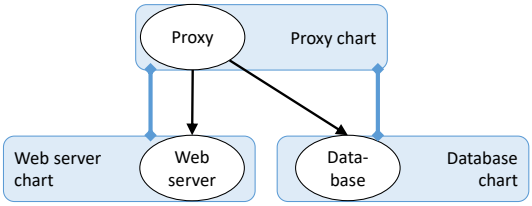
B. Helm Edge-Trimmer (Helm-ET)

Helm-ET[1] is an open source implementation of the policy generation process, written in Go. It runs as an extension of the Helm software, modifying the rendered chart based on the policy configuration. Helm-ET automatically performs all the steps of the proposed approach, including resource labeling and creating the NetworkPolicy objects.

1) Policy generation



(a) Chart with rule customization.



(b) Chart with dependencies.

Fig. 1: Comparison of chart configurations.

During the rendering process, Helm-ET generates NetworkPolicy objects and a policy configuration template. The network policies are automatically added to the cluster, whereas the template is saved to a separate file and may be edited by the administrator. The configuration template includes information about the dependencies and interactions between the system components. Each dependency includes a list of allowed incoming connections, and the higher-level component orchestrates sibling access between its dependencies.

[1]Available at: https://github.com/kubesonde/helmet

TABLE I: Dataset description

Dataset	No. Charts w/ Net. Policies	No. Charts w/o Net. Policies	Total
D1 Microservices	0	2	2
D2 ArtifactHub	41	394	446
D3 DeathStarBench	0	2	2
$\sum$	41	398	450

2) Policy customization

Manual customization is an optional step tailored to chart developers and administrators who want to tune the results of automatic policy generation. For example, the cloud platform provider may require ingress to specific IP subnets for service discovery and metadata. The configuration template allows users to (i) enable sibling access between selected charts, (ii) restrict access to selected recursive dependencies, and (iii) restrict outbound access from selected charts to services outside the cluster or to DNS in the cluster. Fig. 1 has an example of an application with rule customization to allow sibling access and disable access from the proxy to the database. An example of policy customization is available in Appendix **??**.

IV. Evaluation

This section evaluates Helm-ET and compares it with the state of the art. First, we introduce the dataset and experimental setup. Then, we present the results of a security (**E1**). The performance (**E2**) evaluation can be found in the full version of the paper. All experiments are run in a cluster of three virtual machines, each equipped with an 8-core SkyLake processor and 32 GB RAM. We use K3s [11] version v1.28.8, and Calico version v1.27.3 as the CNI.

A. Dataset

Table I summarizes the three datasets used in our evaluation. The first dataset (**D1**) is used to compare Helm-ET with the state of the art. It comprises two OSS applications: *Bookinfo* [12] (v1.18.0) and *Online Boutique* [13] (v0.9.0), which we adapt to be deployed as Helm Charts. These applications and the experimental setup were chosen to allow comparing our tool with AutoArmor[2] [14]. For traffic generation, we use the load testing tool Locust [15], with each load test running for 1000 seconds with 10 concurrent users.

[2]AutoArmor is not publicly available. We contacted the authors but received no response. Thus, only the published policy examples could be used for the comparison.

TABLE II: Dataset D3 details

Application	No. Pods	No. Open Ports	
		Defined	Undefined
Media	33	39	54
Social Network	27	33	35
$\sum$	60	72	89

The second dataset (**D2**) is used to evaluate the security improvement and policy generation time of Helm-ET. It comprises 446 Helm charts obtained from Artifact Hub [16]. We only consider applications that have received at least five stars (user endorsement indicating interest on the application) and that have been updated at least once in the year 2024.

The third dataset (**D3**, Table II) is used to evaluate scalability of Helm-ET against real-world deployments. For that, we employ the Death Star [17] benchmark, which is a collection of Helm charts that deploy large-scale microservices applications. The dataset comprises three charts: (1) Social Network, (2) Hotel Reservation, and (3) Media Microservices. We left out of the experiments Hotel Reservation because the application crashed when deployed.

Consistency

We use the same cluster setup and configuration for all tests to ensure consistency and repeatability. Each experiment begins by installing one chart into an empty cluster. Once the application is running, we measure the internal attack surface. We then repeat the same process by installing the application with the policies generated by Helm-ET. When comparing our results with the state of the art, we also make sure that the microservices are deployed to the same Kubernetes nodes on each execution to avoid unnecessary variation. Whenever possible, the charts are installed with their default configuration. Otherwise, we manually set the minimal configuration parameters required for running the application. The settings are the same when the application runs alone and when it runs with Helm-ET generated policies. The automatically generated policies are deployed without any editing.

False positives

We verify that each chart is correctly running by inspecting the *liveness and readiness* probes of each Pod before and after running the experiment. We assume that the probes have been correctly configured by the chart creators. We store the real number of allowed connections using Kubesonde [18]. Over a total of 451 Helm charts, we were able to successfully install and run 387 of them. The remaining 64 applications — all of which belonged to the D2 dataset — failed to run due to missing dependencies, misconfigured Helm charts, or other issues. This means that Helm-ET was able to generate policies for over 85% of the applications. The remaining 15% of the applications were excluded from the evaluation.

*B. Security evaluation (**E1**)*

The security analysis consists of three parts. First, we evaluate the case when *no policies are available in the chart* (**E1a**). Then, we compare the results of Helm-ET with state-of-the-art tools such as Neuvector [19] and AutoArmor [14] (**E1b**). Finally, we evaluate Helm-ET with charts that have predefined network policies (**E1c**).

*1) Applications without network policies (**E1a**)*

This experiment measures the reduction in the number of allowed connections when deploying Helm-ET on applications that do not have pre-configured network policies (dataset D2). The results are summarized in Fig. 2. It shows the reduction in lateral movement and the number of allowed connections before and after applying Helm-ET. The results show that Helm-ET reduces the number of allowed connections on average by 10% for applications with one dependency, and by more than 50% for applications with four dependencies.

*2) Comparison with other tools (**E1b**)*

This experiment compares the attack surface reduction of Helm-ET with other state-of-the-art tools (dataset D1). Specifically, we compare Helm-ET with AutoArmor [20] and Neuvector [19] by analyzing L3 connections (reachability of ports) for all test cases, and L7 connections (access to application-layer interfaces) for a subset of them. L3 connections are measured with Kubesonde [18], and L7 connections are obtained from the server responses to the HTTP or gRPC requests. The results are summarized in Fig. 3a.

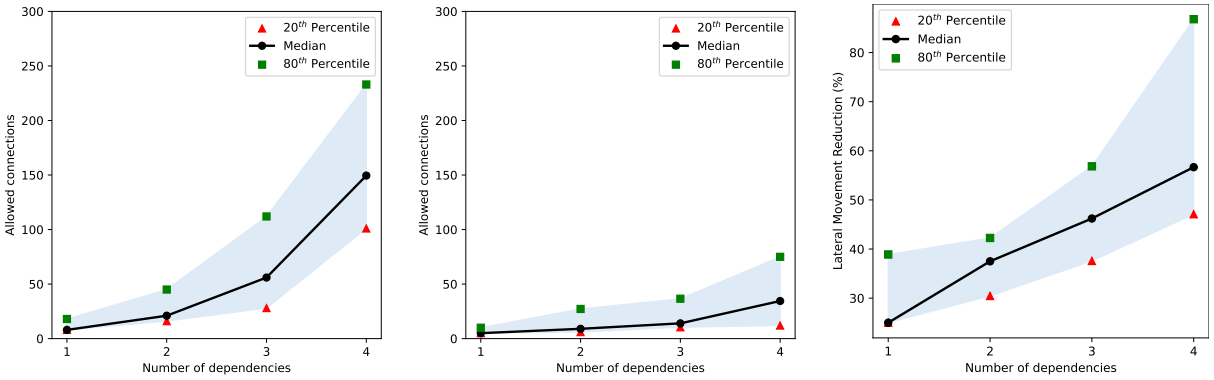
AutoArmor controls L7 connections effectively but does not prevent L3 access. This allows an attacker who is already in the cluster to scan ports and discover the services. The main reason why Helm-ET blocks fewer connections than AutoArmor is that it uses the standard Kubernetes NetworkPolicy resource, which cannot block direct access to ports that implement a Service. The direct access to Services is not considered a security weakness because the client could connect to the same destinations via the Service. However, not being able to block the direct access complicates security posture monitoring by creating seemingly unnecessary open ports.

For L3 connections, Neuvector generates network rules while running a load test for five minutes. We use a list of endpoints from AutoArmor test data [14] for consistent comparison. As can be seen in Fig. 3a, Neuvector restricts the number of connections more than Helm-ET. However, this happens at the cost of blocking valid connections, due to the rule generation being based on load testing. For example, in the BookInfo test case, an endpoint[3] is missing from the load test, which causes it to be blocked by both AutoArmor and Neuvector.

*3) Comparison with pre-defined policies (**E1c**)*

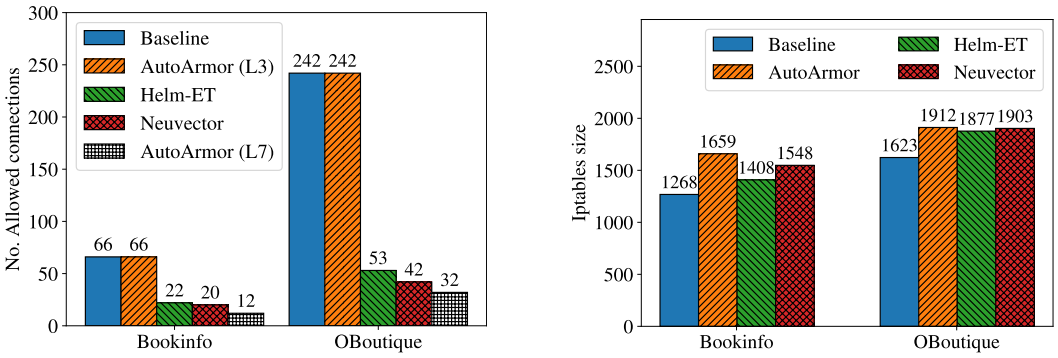
This experiment compares the network policies generated by Helm-ET with those already included in

[3] `GET /api/v1/products/<product_id>/ratings`



(a) Before applying Helm-ET (b) After applying Helm-ET (c) Lateral movement reduction (%)

Fig. 2: Comparison of allowed connections before and after applying Helm-ET for applications without network policies (**E1a**).



(a) Allowed L3 and L7 connections (**E1b**). (b) Number of iptables rules.

Fig. 3: Number of allowed connections and iptables rules.

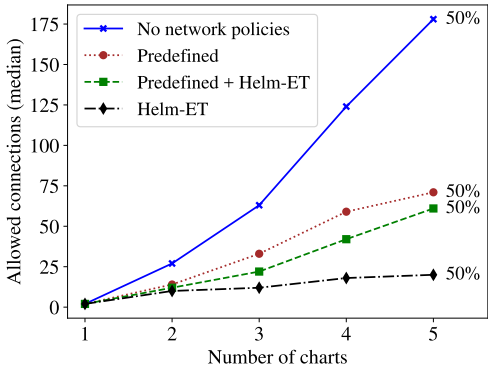


Fig. 4: Median number of allowed connections for charts with NetworkPolicy (**E1c**).

applications (dataset D2), containing 41 applications from Artifact Hub [16] that include a pre-defined network policy (see Table I). We compare the number of connections when the application (1) does not have any network policy, (2) uses the predefined policies, (3) uses Helm-ET policies, and (4) uses a combination of both predefined and Helm-ET policies.

Fig. 4 shows the median number of possible connections for the different policies. The horizontal axis is the number of charts in the application. The reason for the pre-defined policies not being as strict as Helm-ET policies is that the policies typically only block incoming traffic to the chart, and do not include a cluster-wide default deny policy. Helm-ET creates default deny policies for all compute resources. Note that combining Helm-ET policies with pre-defined policies results in a less strict policy because, after setting a default deny policy everywhere, the network policy rules are additive allow rules.

The number of charts with predefined policies is relatively small (41 charts), which explains the variation of the results. Nevertheless, the policies generated by Helm-ET are able to restrict the number of connections better than the chart or application developer policies. Since the combination of both policies is additive, this experiment supports the usage of Helm-ET both for charts that have missing network policies or that include them.

V. Conclusion

This paper presents a novel approach to creating network policies for Kubernetes clusters. The work was motivated by the lack of isolation boundaries in cloud applications, with less than 10% of the tested applications providing a network policy. We developed a methodology for generating network-layer

isolation policies from the compositional structure of the application. We implemented the methodology for Helm charts and evaluated it with an extensive set of published charts. The approach can be used in any Kubernetes cluster, without installing new resources and with negligible latency. The evaluation results show that our approach can automatically apply policies to over 85% of the top downloaded Helm charts and reduce the number of unnecessary connections by 50% on average.

References

[1] J. Bufalino, J. L. Martin-Navarro, A. Peltonen, and T. Aura, "Helm-et: Reducing exposure to lateral movement in kubernetes artifacts," in *2025 IEEE 18th International Conference on Cloud Computing (CLOUD)*. IEEE, 2025, pp. 109–120.

[2] N. Dragoni, S. Giallorenzo, A. L. Lafuente, M. Mazzara, F. Montesi, R. Mustafin, and L. Safina, *Microservices: Yesterday, Today, and Tomorrow*. Springer, 2017.

[3] D. K. Rensin, *Kubernetes - Scheduling the Future at Cloud Scale*. O'Reilly, 2015. [Online]. Available: http://www.oreilly.com/webops-perf/free/kubernetes.csp

[4] Helm, "Helm: The package manager for Kubernetes," 2024, Accessed on April 17, 2024. [Online]. Available: https://helm.sh

[5] A. Blaise and F. Rebecchi, "Stay at the Helm: secure Kubernetes deployments via graph generation and attack reconstruction," in *2022 IEEE 15th International Conference on Cloud Computing (CLOUD)*. IEEE, 2022, pp. 59–69.

[6] A. Rahman, C. Parnin, and L. Williams, "The seven sins: Security smells in infrastructure as code scripts," in *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*. IEEE, 2019, pp. 164–175.

[7] M. Chiari, M. De Pascalis, and M. Pradella, "Static analysis of infrastructure as code: a survey," in *2022 IEEE 19th International Conference on Software Architecture Companion (ICSA-C)*. IEEE, 2022, pp. 218–225.

[8] OWASP, "OWASP: Cloud-native application security top 10," 2022, Accessed on April 17, 2024. [Online]. Available: https://web.archive.org/web/20250428105129/owasp.org/www-project-cloud-native-application%2Dsecurity-top-10/

[9] National Security Agency (NSA) and Cybersecurity and Infrastructure Security Agency (CISA), "Kubernetes hardening guidance," 2022, Accessed on April 17, 2024. [Online]. Available: https://www.nsa.gov/Press-Room/News-Highlights/Article/Article/2716980/nsa-cisa-release-kubernetes-hardening-guidance

[10] Kubernetes, "Kubernetes design proposals," 2017, Accessed on April 17, 2024. [Online]. Available: https://github.com/kubernetes/design-proposals-archive/blob/main/network/networking.md

[11] Rancher Labs, "K3s: Lightweight Kubernetes," 2024, Accessed on April 18, 2024. [Online]. Available: https://k3s.io

[12] Istio, "Bookinfo application," 2024, Accessed on April 18, 2024. [Online]. Available: https://istio.io/latest/docs/examples/bookinfo

[13] GoogleCloudPlatform, "Microservices demo," 2024, Accessed on April 18, 2024. [Online]. Available: https://github.com/GoogleCloudPlatform/microservices-demo

[14] X. Li, Y. Chen, and Z. Lin, "Towards automated inter-service authorization for microservice applications," in *Proceedings of the ACM SIGCOMM 2019 Conference Posters and Demos*, ser. SIGCOMM Posters and Demos '19. ACM, 2019, pp. 3–5.

[15] Locust, "Locust," 2024, Accessed on April 17, 2024. [Online]. Available: https://locust.io

[16] The Linux Foundation, "Artifact Hub," 2024, Accessed on April 17, 2024. [Online]. Available: https://artifacthub.io

[17] Y. Gan, C. Waldspurger, S. Venkataraman, Q. Li, N. Satish, and C. Delimitrou, "An open-source benchmark suite for microservices and their hardware-software implications for cloud and edge systems," in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. ACM, 2019, pp. 3–18.

[18] J. Bufalino, M. Di Francesco, and T. Aura, "Analyzing microservice connectivity with Kubesonde," in *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE 2023. ACM, 2023, pp. 2038–2043.

[19] Suse, "NeuVector: Full lifecycle container security," 2024, Accessed on April 17, 2024. [Online]. Available: https://www.suse.com/neuvector

[20] X. Li, Y. Chen, Z. Lin, X. Wang, and J. H. Chen, "Automatic policy generation for inter-service access control of microservices," in *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, August 2021, pp. 3971–3988. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/li-xing

Inteligencia Artificial aplicada al análisis de códigos convolucionales y sus representaciones ISO

Lucía Mallo-Fernández,David Escudero García
Research Institute of Applied Sciences
in Cybersecurity (RIASC)
Universidad de León.
Campus de Vegazana s/n 24007 León
lmalf@unileon.es,descg@unileon.es

Noemí DeCastro-García,Ángel Luis Muñoz Castañeda
Departamento de Matemáticas.
Universidad de León
Campus de Vegazana s/n 24007 León
ncasg@unileon.es,amunc@unileon.es

***Resumen*—En este trabajo se realiza una aproximación exploratoria al análisis de códigos convolucionales, fundamentales en comunicaciones y criptografía post-cuántica, basada en el descubrimiento automático de conocimiento mediante técnicas de inteligencia artificial. La base de datos se construye mediante la vinculación existente entre códigos convolucionales y sus representaciones ISO, sistemas dinámicos lineales. Empleando métodos de minería de reglas de asociación y medidas de información mutua, se identifican relaciones y conjeturas entre propiedades algebraicas de los sistemas y las propiedades que verifican los códigos convolucionales asociados. Los resultados confirman la validez de varios teoremas conocidos y sugieren nuevas direcciones para la generación automatizada de hipótesis en teoría de la codificación.**

***Index Terms*—códigos convolucionales, inteligencia artificial, sistemas dinámicos, descubrimiento de conocimiento**

I. Introducción

Los códigos convolucionales son una clase de códigos de detección y corrección de errores que desempeñan un papel fundamental en garantizar la transmisión confiable de datos, tanto sobre canales ruidosos, como de borrado. Se utilizan ampliamente en las tecnologías de comunicación y son cada vez más reconocidos por su potencial aplicación en el criptosistema de McEliece, un candidato para la criptografía post-cuántica. Esta característica posiciona a los códigos convolucionales como una herramienta altamente relevante en el campo de la ciberseguridad, donde garantizar la confidencialidad, integridad y fiabilidad de la información es crítico ante amenazas cada vez más sofisticadas. Sin embargo, el diseño de códigos convolucionales con buenas características (observabilidad, algoritmos de decodificación eficientes y distancias óptimas) presenta desafíos significativos. Por estas razones, comprender las conexiones entre los códigos convolucionales y otras estructuras matemáticas constituye un área de investigación activa y esencial. Este trabajo está contextualizado en la estrecha relación entre los códigos convolucionales y los sistemas dinámicos lineales a través de las representaciones ISO (Input/State/Output) de los códigos convolucionales.

Como perspectiva innovadora, nuestro objetivo es emplear *Descubrimiento Automático de Conocimiento* o *Descubrimiento Guiado por Inteligencia Artificial* (IA, por sus siglas en inglés) para detectar y revelar relaciones y propiedades de los objetos mencionados. El descubrimiento automatizado de conocimiento es el término que se utiliza para describir el uso de técnicas de IA para identificar patrones o generar hipótesis novedosas a partir de los datos de estudio [1]. La ciencia basada en descubrimiento automatizado busca emplear algoritmos de IA para formular hipótesis científicas, diseñar experimentos y analizar datos de manera automática [2]. No se trata simplemente de explorar conjuntos de datos para sugerir explicaciones para un conjunto de observaciones, o para buscar relaciones que puedan utilizarse para generar predicciones. En este caso, se usaría la IA como un asistente en la formulación de conjeturas que luego pueden ser probadas mediante técnicas de matemática formal. La generación de conjeturas mediante IA fue introducida en [3] en el contexto de la teoría de cuerdas, pero es un campo emergente en la aplicación de IA en lo que se consideran ciencias puras (véase, por ejemplo, los trabajos incluidos en [4], [5], [6], [7]). Aunque el aprendizaje automático ha sido utilizado en el campo de los códigos convolucionales para mejorar varios aspectos clave, como la decodificación, la optimización de parámetros y la detección de errores, hasta el momento, no se ha aplicado IA para la generación de hipótesis en el campo de los códigos convolucionales. Al incorporar IA, buscamos generar hipótesis en el estudio de las condiciones que las matrices de la representación ISO deben verificar para obtener códigos convolucionales con buenas propiedades. Esta metodología, que diverge de los métodos tradicionales en la disciplina, requiere construir una base de datos integral que capture las características más relevantes de ambos tipos de objetos matemáticos: los códigos convolucionales y las representaciones ISO. De manera transversal a la generación de hipótesis, disponer de un conjunto de datos de calidad de códigos convolucionales y sus representaciones ISO, puede ser útil para toda la comunidad científica que se dedica a la disciplina, para la investigación en sistemas dinámicos, y para la relación entre ambos objetos matemáticos. Esta aproximación permitiría generar hipótesis, pero también proponer ejemplos y contraejemplos de manera más eficiente. Eventualmente, también se podrían estudiar nuevos enfoques para construir códigos convolucionales óptimos a través de las propiedades de los sistemas lineales [8]. En este trabajo, presentamos nuestros resultados preliminares. Se muestra que el análisis de datos es capaz de detectar comportamientos teóricos conocidos, y se pone de manifiesto el potencial de la IA para formular nuevas hipótesis matemáticas en el ámbito de la codificación.

Este documento está organizado de la siguiente manera: En la Sección 2, se aportan los preliminares teóricos del tópico de investigación. En la sección 3, se desarrollan los resultados preliminares obtenidos. Finalmente, se incluyen las conclusiones y las referencias.

II. Códigos convolucionales y sistemas dinámicos

Definición 1: Un (n, k, δ) - código convolucional $\mathcal{C}$ sobre un cuerpo finito es un submódulo de $\mathbb{F}[z]^n$, donde $\mathbb{F}$ es un cuerpo finito, n es el número de símbolos de salida, k es el número de símbolos de entrada, y δ es la complejidad del código.

Definición 2: Un codificador convolucional $G(z)$ representa la transformación de una secuencia de entrada $u(z)$ a una secuencia de salida $v(z)$, dada por

$$v(z) = G(z) \cdot u(z),$$

donde $G(z)$ es una matriz sobre el anillo $\mathbb{F}[z]$ (polinomios sobre el cuerpo finito $\mathbb{F}$).

Una propiedad esencial de los códigos convolucionales es la observabilidad del código, característica que implica que el código no sea catastrófico.

Definición 3 (Lema 3.3.2. [9]): Si $G(z)$ es un codificador de un (n, k, δ)- código convolucional $\mathcal{C}$ sobre $\mathbb{F}$, $\mathcal{C}$ es observable si y sólo si existe un síndrome de Former ψ tal que la siguiente sucesión es exacta:

$$0 \to \mathbb{F}[z]^k \xrightarrow{G(z)} \mathbb{F}[z]^n \xrightarrow{\psi} \mathbb{F}[z]^{n-k} \to 0.$$

Otro área de estudio relevante en la teoría de los códigos convolucionales es el proceso de decodificación. Sea $u(z)$ una palabra de información, y $G(z)$ un codificador para un (n, k, δ)- código convolucional . La palabra código $v(z) = G(z) \cdot u(z)$ se transmite a través de un canal, donde pueden añadirse errores o producirse borrados. El objetivo del proceso de decodificación es recuperar $v(z)$ a partir de la palabra código errónea. La búsqueda de algoritmos de decodificación para códigos convolucionaes no es trivial. Por ese motivo, una línea de investigación actual en códigos convolucionales es construir códigos que tengan buenas propiedades de decodificación o *Good Decodable Property* (GDP, por sus siglas en inglés); es decir, para los que existan algoritmos de decodificación eficientes. En este contexto, los códigos convolucionales observables admiten métodos de decodificación algebraicos que mejoran en eficiencia al algoritmo de Viterbi cuando es aplicado sobre ciertas familias de códigos estructurados (por ejemplo, Reed-Solomon [10]). Este hecho resulta especialmente interesante en el ámbito de la criptografía basada en códigos convolucionales [11], [12] ya que permiten reducir el tamaño de clave en el criptosistema de McEliece a la vez que admiten algoritmos de decodificación eficiente.

Otro punto fundamental en la teoría de códigos convolucionales es la distancia del código. La distancia de un código convolucional es una medida que evalúa su capacidad para detectar y corregir los errores de los datos. Se pueden definir varios tipos de distancia para códigos convolucionales: la distancia libre (distancia mínima de Hamming entre dos secuencias de código cualesquiera), y las llamadas distancias columna. La distancia libre y las distancias por columnas tienen cotas cuyo alcance sirve para definir códigos considerados óptimos en términos de distancias. Sea $\mathcal{C}$ un (n, k, δ)-código convolucional sobre $\mathbb{F}$, entonces :

Definición 4 ([13]): Un (n, k, δ)- código convolucional se denomina *MDS* (*maximum distance separable*), si la distancia libre es igual a la cota de Singleton generalizada; $d_{\text{free}}(\mathcal{C}) = (n-k)\left(\left\lfloor \frac{\delta}{k} \right\rfloor + 1\right) + \delta + 1$.

Definición 5: Si la distancia de columna es igual a su cota superior, $d_j^c(\mathcal{C}) = (n-k)(j+1)+1$ para $j = 0, \ldots, \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lfloor \frac{\delta}{n-k} \right\rfloor$, el código convolucional se denomina *MDP* [14], [15].

La construcción de códigos convolucionales que verifiquen simultáneamente las condiciones de observabilidad, óptima distancia y eficiencia en la decodificación constituye un desafío. Por ello, las investigaciones recientes se centran en buscar aproximaciones y métodos constructivos alternativos que permitan tratar algunas de estas características y encontrar nuevas familias de códigos convolucionales que mantengan un rendimiento cercano al óptimo sin sacrificar la viabilidad práctica de su implementación. Una de las aproximaciones más conocidas para estudiar los códigos convolucionales es a través de las representaciones ISO.

Definición 6: Section 5.2, [9], Remark 4.1, [16] Sea $\mathcal{C} \subset \mathbb{F}[z]^n$ un (n, k, δ)-código convolucional . Una representación ISO de $\mathcal{C}$ es una tupla de matrices $\Sigma = (A, B, C, D) \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{(n-k)\times\delta} \times \mathbb{F}^{(n-k)\times k}$ tal que

$$\mathcal{C} = \left\{ \begin{array}{l} v(z) = \begin{pmatrix} y(z) \\ u(z) \end{pmatrix} \in \mathbb{F}[z]^n : \exists x(z) \in \mathbb{F}[z]^\delta \\ \\ \text{que satisface} \left\{ \begin{array}{rcl} x_{t+1} & = & Ax_t + Bu_t \\ y_t & = & Cx_t + Du_t \\ v_t & = & \begin{pmatrix} y_t \\ u_t \end{pmatrix} \\ x_0 & = & 0 \end{array} \right. \end{array} \right\}.$$

donde $x_t \in \mathbb{F}^\delta$ denota el vector de estado, $u_t \in \mathbb{F}^k$ representa el vector de control/entrada/información, y $y_t \in \mathbb{F}^{n-k}$ es el vector de salida/verificación de paridad para cada instante de tiempo t. El vector v_t es el vector del código transmitido a través del canal de comunicación. Por simplicidad, diremos que $\Sigma = (A, B, C, D) \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{(n-k)\times\delta} \times \mathbb{F}^{(n-k)\times k}$ forma la representación I/S/O de $\mathcal{C}(\Sigma)$.

Para poder obtener una representación ISO para un (n, k, δ)-código convolucional dado sobre $\mathbb{F}$ se requiere primero obtener una representación de primer orden mínima (Teorema 5.1.1, [9], Teorema 3.1, [16]): Sea $\mathcal{C} \subset \mathbb{F}[z]^n$ un código convolucional (n, k, δ). Las matrices $(\mathcal{K}, \mathcal{L}, \mathcal{M})$ se denominan representación de primer orden del código $\mathcal{C}$ si satisfacen las siguientes condiciones:

(i) $\mathcal{C} = \{v(z) \in \mathbb{F}[z]^n : \exists x(z) \in \mathbb{F}[z]^\delta$ tal que $z\mathcal{K}x(z) + \mathcal{L}x(z) + \mathcal{M}v(z) = 0\}$.
(ii) $\mathcal{K}$ tiene rango máximo.
(iii) $(\mathcal{K}, \mathcal{M})$ tiene rango máximo.

Además, si $(\mathcal{K}, \mathcal{L}, \mathcal{M})$ también satisface la propiedad adicional:

(iv) $(z\mathcal{K} + \mathcal{L}, \mathcal{M})$ tiene rango máximo,

entonces la representación de primer orden se denomina mínima.

A partir de $(\mathcal{K}, \mathcal{L}, \mathcal{M})$, y mediante una permutación apropiada de las palabras código, si es necesario, podremos obtener Σ:

$$\mathcal{K}' = \begin{pmatrix} -I_\delta \\ 0 \end{pmatrix}, \quad \mathcal{L}' = \begin{pmatrix} A_{\delta\times\delta} \\ C_{(n-k)\times\delta} \end{pmatrix},$$
$$\mathcal{M}' = \begin{pmatrix} 0 & B_{\delta\times k} \\ -I_{n-k} & D_{(n-k)\times k} \end{pmatrix}.$$

Recientemente, se han diseñado algoritmos para poder transformar directamente el codificador en Σ sin necesidad de calcular la representación mínima de primer orden [17].

Desde el punto de vista de sistemas dinámicos, el sistema obtenido de la manera descrita, Σ, es un sistema accesible; es decir, rk(Φ_δ) =δ siendo

$$\Phi_\delta(A,B) = \begin{pmatrix} B & AB & \cdots & A^{\delta-2}B & A^{\delta-1}B \end{pmatrix}$$

la matriz de accesibilidad. La accesibilidad de Σ proviene de la propiedad de minimalidad de la representación ISO obtenida del código convolucional por el proceso descrito (Lemma 5.3.5, [9]).

Por otra parte, y dado que podemos obtener un codificador $G(z)$ calculando una base mínima del $\mathbb{F}[z]$-módulo libre

$$\text{Ker}(z\mathcal{K} + \mathcal{L} \mid \mathcal{M}) = \left\{ v(z) \in \mathbb{F}[z]^n \mid \exists x(z) \in \mathbb{F}[z]^\delta : \right.$$
$$\left. (z\mathcal{K} + \mathcal{L})x(z) + \mathcal{M}v(z) = 0 \right\},$$

las representaciones ISO nos permiten construir códigos convolucionales observables basados en las propiedades del sistema asociado (Lema 5.3.5, [9]): $\mathcal{C}(\Sigma)$ es observable (no catastrófico) $\Leftrightarrow$ Σ es accesible y observable $\Leftrightarrow$ rk(Φ_δ) = rk(Ω_δ) = δ, donde

$$\Omega_\delta(A,C) = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{\delta-1} \end{pmatrix}.$$

es la matriz de observabilidad. En el caso en el que Σ es accesible y observable, se dice que Σ es una es una representación ISO mínima del código.

Desde el punto de vista de la decodificación, la existencia de representaciones ISO mínimas permite el desarrollo de nuevos algoritmos de decodificación. Cuando el canal es ruidoso, los primeros algoritmos de decodificación que utilizan representaciones ISO fueron introducidos en [10] y se basan en una relación algebraica entre la secuencia de entrada u_t y la secuencia de salida y_t que proporciona la ISO. Si tenemos un (n,k,δ)- código convolucional $\mathcal{C}(\Sigma) \subseteq \mathbb{F}[z]^n$, cualquier secuencia de código v_t, debe satisfacer la dinámica del sistema (Proposición 2.6, [10]):

$$\begin{pmatrix} C \\ CA \\ \vdots \\ CA^L \end{pmatrix} x_t + \left(\begin{array}{c|cccccc} & D & 0 & 0 & \dots & 0 & 0 \\ & CB & D & 0 & \dots & \vdots & \vdots \\ -I_{L+1} & CAB & CB & D & \dots & 0 & 0 \\ & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ & CA^{L-2}B & CA^{L-3}B & CA^{L-4}B & \dots & D & 0 \\ & CA^{L-1}B & CA^{L-2}B & CA^{L-3}B & \dots & CB & D \end{array} \right) \begin{pmatrix} y_t \\ y_{t+1} \\ \vdots \\ y_{t+L} \\ u_t \\ u_{t+1} \\ \vdots \\ u_{t+L} \end{pmatrix} = 0$$

lo cual denotamos por:

$$\Omega_{L+1}(C,A) \cdot \mathbf{x_t} + (-I_{L+1} \mid F_L) \cdot \mathbf{v_t} = 0 \qquad (1)$$

donde F_L es una matriz de bloques de tipo Toeplitz. A partir del resultado descrito, podemos determinar diferentes propiedades sobre Σ para poder usar la representación ISO con el objetivo de construir códigos convolucionales GDP:

- En canales ruidosos: Si Σ es accesible, observable y A es invertible, entonces $\mathcal{C}(\Sigma)$ es GDP_{ruido}.
- En canales de borrado: Si Σ es accesible, observable y *output observable*; es decir, si el rango de la matriz .

$$\mathbf{T_L} := [\Omega_{L+1}(A,C) \mid \mathbf{F_L}] = \left(\begin{array}{c|cccccc} C & D & & & & \\ CA & CB & D & \vdots & \vdots & \\ CA^2 & CAB & CB & D & \vdots & \\ \vdots & \vdots & & \vdots & & \\ CA^L & CA^{L-1}B & CA^{L-2}B & \cdots & CB & D \end{array} \right),$$

es máximo, donde $L := \left\lfloor \frac{\delta}{k} \right\rfloor + \left\lfloor \frac{\delta}{n-k} \right\rfloor > 0$, entonces $\mathcal{C}(\Sigma)$ es GDP_{borrado}.

Por último, $\mathcal{C}(\Sigma)$ es MDP $\Leftrightarrow$ cada menor de F_L, que no sea trivialmente nulo, es distinto de cero. Por lo tanto, si F_L es superregular, el código convolucional asociado es un código MDP [Teorema 2.4., [15]]

III. Materiales y métodos

III-A. Conjunto de datos

Las *features* (variables) con las que se ha trabajdo para generar la base de datos son:

- Parámetros del código/sistema: q, n, k, δ.
- Generación de matrices: $\Sigma = (A,B,C,D)$, donde cada elemento de las matrices toma valores de 0 a $q-1$.
- Cálculo de matrices: Φ_δ, Ω_δ, T_L y F_L. Se incluye también el cálculo de la representación de primer orden asociada.
- Variables: A invertible, F_L superregular, Σ accesible, observable, output-observable, rangos de las matrices que forman Σ, $\mathcal{C}(\Sigma)$ es MDP, $\mathcal{C}(\Sigma)$ es observable y $\mathcal{C}(\Sigma)$ es GDP (en canal de borrado y con ruido).

III-B. Análisis

1. El primer análisis realizado ha sido porcentual.
2. Tras haber realizado un estudio exploratorio, con los resultados obtenidos, se realiza un estudio de la relación entre que el código sea GDP y otras propiedades de su representación ISO asociada. Con este fin, se utiliza la función de información mutua [18]. La Función de Información Mutua (normalizada) de dos variables aleatorias X y Y, MIF(X,Y), es una función basada en la entropía que determina la cantidad de información compartida por ambas variables. Se utiliza la implementación normalizada de la librería *sklearn* [19].
3. El otro tipo de análisis que se ha realizado es Association Rules Mining (ARM, por sus siglas en inglés). El minado de reglas de asociación es un proceso que tiene como objetivo descubrir relaciones o reglas de asociación entre conjuntos de elementos en grandes bases de datos [20]. Sea $\mathcal{I} = \{i_1, \dots, i_d\}$ un conjunto de elementos, y $\mathcal{T} = \{T_1, \dots, T_n\}$, con $T_j \in \mathcal{P}(\mathcal{I})\, \forall j = 1, \dots, n$ un conjunto de datos de transacciones. Las reglas de asociación se extraen de $\mathcal{T}$ y son reglas de la forma $X \Rightarrow Y$, donde $X \subseteq \mathcal{I}$ es el antecedente e $Y \subseteq \mathcal{I}$ es el

consecuente. Este tipo de reglas nos permitirán identificar combinaciones frecuentes de atributos que aparecen juntos, lo cual puede sugerir interacciones importantes entre características que no serían evidentes mediante otro tipo de estudios. Para determinar la validez de una regla, se suelen utilizar tres métricas: soporte (*support*), confianza (*confidence*) y elevación (*lift*).

$$\begin{aligned}\text{supp}(A) &= \frac{|\{T_k \in \mathcal{T} : A \subseteq T_k\}|}{|\mathcal{T}|},\\ \text{supp}(A \Rightarrow B) &= \text{supp}(A \cup B),\\ \text{conf}(A \Rightarrow B) &= \frac{\text{supp}(A \cup B)}{\text{supp}(A)},\\ \text{lift}(A \Rightarrow B) &= \frac{\text{supp}(A \cup B)}{\text{supp}(A) \cdot \text{supp}(B)}.\end{aligned}$$

donde $A, B \subseteq \mathcal{I},\ A \cap B = \emptyset$. Se han considerado todas las variables lógicas de la base de datos y se han construido variables booleanas adicionales que toman valor `True` cuando se cumplen las siguientes condiciones: $rk(\Phi) = rk(\Omega)$, $rk(A) = \delta$, $rk(B) = \delta$, $rk(C) = \delta$, $rk(D) = \delta$, $A_eigenvalues_zero = 0$, $\mathcal{C}(\Sigma)$ es GDP_{borrado} y $\mathcal{C}(\Sigma)$ es GDP_{ruido}.
Se han extraído todas las reglas cuyas antecedentes tienen un único elemento y todas las reglas cuyo consecuente tiene un único elemento. En este caso, como el objetivo es la obtención de relaciones demostrables, únicamente se han considerado las reglas con $confidence = 1$. Se han dividido las reglas obtenidas en dos clases: por un lado, aquellas que se corresponden exactamente con las normas empleadas para el etiquetado de los datos y que, por lo tanto, no aportan infomación nueva alguna, y, por otro lado, aquellas que sí son informativas.
Para ARM, se ha utilizado implementación del algoritmo Apriori para ARM [21] del módulo de Python *mlxtend.frequent_patterns* [22].

IV. RESULTADOS PRELIMINARES

IV-A. Base de datos de códigos convolucionales y representaciones I/S/O

El primer estudio realizado ha sido la complejidad de la creación de la base de datos. Dado un cuerpo finito $\mathbb{F}_q$ y los parámetros k, n y δ, el espacio

$$\mathbb{F}_q^{\delta\times\delta} \times \mathbb{F}_q^{\delta\times k} \times \mathbb{F}_q^{(n-k)\times\delta} \times \mathbb{F}_q^{(n-k)\times k}$$

tiene cardinalidad $q^{\delta^2+\delta n+(n-k)k}$. Por lo tanto, la creación de una base de datos adecuada requerirá estrategias de optimización utilizando resultados de equivalencia de sistemas y códigos. Para este trabajo, se ha optado por recolectar los datos mediante algoritmos de búsqueda por fuerza bruta con parámetros con valor bajo. Así, se han creado todos los sistemas posibles para $q = 2$ y combinaciones de parámetros k, n, δ en los rangos $k = 1, 2; n = 2, 3;\ \delta = 1, 2, 3$. En total se han generado y analizado 2,171,536 sistemas y las caracterizaciones de los mismos vinculadas a los códigos convolucionales asociados considerándolos como ISO. El conjunto de datos está disponible en [23].

IV-B. Análisis estadístico exploratorio

En la Tabla I se incluye un resumen del análisis frecuencial realizado.

Como podemos observar en la Tabla II, los resultados encontrados mediante análisis de datos concuerdan con los resultados esperados:

- La probablidad de que un (n, k, δ)-código convolucional $\mathcal{C}(\Sigma)$ con Σ accesible sea no catastrófico es $1-t^{(n-k)}+O(t^{(n-k)+1})$ [Teorema 58, [24]]. En este caso, $t = \frac{1}{|\mathbb{F}|} = 0{,}5$.
- La probablidad de que un código convolucional escogido aleatoriamente con parámetros $k = 1, n = 2, \delta = 1$ sea MDP es $1-5t+O(t^2)$. En particular, no existen códigos binarios MDP con estos parámetros [Teorema 84, [24]]. En este caso, $t = \frac{1}{|\mathbb{F}|} = 0{,}5$.

Como para $q = 2$ no han salido códigos convolucionales MDP, en este trabajo sólo nos centraremos en propiedades de observabilidad y decodificación.

IV-C. Información mutua

La Tabla III muestra los valores obtenidos de la función de información mutua cuando el objetivo es que el código convolucional asociado tenga la propiedad GDP_{ruido}. Las relaciones que aparecen están completamente vinculadas a la forma de etiquetado de la base de datos: existe una coincidencia perfecta, MIF = 1, entre GDP_{ruido} y las variables que representan los autovalores de A iguales a 0, rk(A)= δ y rk(Φ)= rk(Ω).

La Tabla IV muestra el valor de la función de información mutua de la propiedad GDP_{borrado} con otras variables. En este caso, no hay ninguna dependencia fuerte con otras variable excepto en el caso en que el rk(Φ)=rk(Ω), que está relacionado de forma indirecta con el propio etiquetado de GDP_{borrado}.

IV-D. Aplicación de ARM

En la Tabla V se muestran las reglas que se han encontrado y que se corresponden con etiquetado de las variables correspondientes. Por lo tanto, no aportan información nueva. En la Tabla VI se muestran las reglas que muestran relaciones que no derivan del etiquetado de los datos y, por lo tanto, son informativas.

De las tablas anteriores se pueden extraer diferentes reglas de asociación. Algunas de ellas, son resultados extensamente conocidos y demostrados.

1) A invertible $\Leftrightarrow rk(A) = \delta \Leftrightarrow$ Todos los autovalores de A son no nulos. Demostrado en [25].
2) Sea $\Sigma \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{(n-k)\times\delta} \times \mathbb{F}^{(n-k)\times k}$. Si $rk(B) = \delta \Rightarrow \Sigma$ es accesible. Demostrado en [26]. La condición es suficiente pero no necesaria, hecho que también detecta la aplicación del ARM introduciendo $rk(B)$ como antecedente y no como consecuente.
3) Sea $\Sigma \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{(n-k)\times\delta} \times \mathbb{F}^{(n-k)\times k}$. Si $rk(C) = \delta \Rightarrow \Sigma$ es observable. Demostrado en [26]. La condición es suficiente pero no necesaria, hecho que también detecta la aplicación del ARM introduciendo el $rk(C)$ como antecedente y no como consecuente.
4) Sea $\Sigma \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{(n-k)\times\delta} \times \mathbb{F}^{(n-k)\times k}$. Si Σ es accesible y observable $\Rightarrow rk(\Phi_\delta) = rk(\Omega_\delta)$. Se deduce

Tabla I
RESUMEN DE LA BASE DE DATOS

Condición sobre Σ	**Cantidad**	**% /Total**	**Representación ISO para $\mathcal{C}(\boldsymbol{\Sigma})$**
Accesible	1,015,192	46.75 %	Representación ISO para $\mathcal{C}(\boldsymbol{\Sigma})$
Observable	1,015,192	46.75 %	
Output observable	1,311,960	60.42 %	
A invertible	712,968	32.83 %	
Accesible + Observable	529,300	24.37 %	ISO para código observable
Accesible + Observable + Output observable	399,688	18.41 %	ISO para código observable y GDP_{borrado}
Accesible + Observable + A invertible	302,774	13.94 %	ISO para código observable y GDP_{ruido}
Accesible + Observable + A invertible + Output observable	219,752	10.12 %	ISO para código observable, GDP_{borrado} y GDP_{ruido}
F_L superregular	0	0.00 %	Posible ISO para código MDP

Tabla II
PROPORCIÓN DE CÓDIGOS OBSERVABLES SOBRE AQUELLOS CON Σ ACCESIBLE

n	k	δ	Proporción real	Teórica
2	1	1	0.5000	0.5+O(0.25)
2	1	2	0.5000	0.5+O(0.25)
2	1	3	0.5000	0.5+O(0.25)
3	1	1	0.7500	0.75+O(0.125)
3	1	2	0.7500	0.75+O(0.125)
3	1	3	0.7500	0.75+O(0.125)
3	2	1	0.5000	0.5+O(0.25)
3	2	2	0.4286	0.5+O(0.25)
3	2	3	0.4000	0.5+O(0.25)

Tabla III
INFORMACIÓN MUTUA DE LA PROPIEDAD GDP_{RUIDO} CON OTRAS VARIABLES DEL CONJUNTO DE DATOS.

Variable	MIF
Autovalores de A iguales a 0	0.3473
Rk(A)	0.3473
Rk(Φ)=Rk(Ω)	0.2874
Rk(C)	0.0179
Rk(B)	0.0179
Σ output-observable	0.0097
Rk(D)	2.127 x 10^{-6}

de la propia definición: si Σ es accesible y observable, entonces $rk(\Phi_\delta) = \delta = rk(\Omega_\delta)$.

Así, se prueba que el análisis ARM es capaz de deducir relaciones matemáticas ciertas y, por lo tanto, tiene potencial para descrubir otras nuevas. Por otro lado, analizando las conjeturas detectadas, se puede proponer la siguiente:

Proposición 1: Sea $\Sigma \in \mathbb{F}^{\delta\times\delta} \times \mathbb{F}^{\delta\times k} \times \mathbb{F}^{(n-k)\times\delta} \times \mathbb{F}^{(n-k)\times k}$. Si $rk(C) = rk(B) = \delta$, entonces $\mathcal{C}(\Sigma)$ es un (n, k, δ)- código observable sobre $\mathbb{F}$.

Proof: Si $rk(B) = \delta$, entonces Σ es accesible [26]. Por lo tanto, se puede considerar Σ como una representación ISO para un (n, k, δ)- código convolucional [9]. Por otro lado, si $rk(C) = \delta$, Σ es observable [26]. Entonces, Σ es una representación ISO minima para $\mathcal{C}$ y $\mathcal{C}(\Sigma)$ es observable [9], [14] ■

Tabla IV
INFORMACIÓN MUTUA DE LA PROPIEDAD GDP_{BORRADO} CON OTRAS VARIABLES DEL CONJUNTO DE DATOS.

Variable	MIF
Rk(Φ)=Rk(Ω)	0.3737
Rk(A)	0.0514
A es invertible	0.0428
Rk(B)	0.0381
Rk(C)	0.0309
Autovalores de A iguales a 0	0.0301
Rk(D)	0.0080

Tabla V
REGLAS DE ETIQUETADO

Antecedente	**Consecuente**	***Lift***
A invertible, Σ accesible, Σ observable	$\mathcal{C}(\Sigma)$ es GDP_{ruido}	7.1721
Σ es output_observable, Σ accesible, Σ observable	$\mathcal{C}(\Sigma)$ es GDP_{borrado}	5.4331
$\mathcal{C}(\Sigma)$ es GDP_{ruido}	A invertible	3.0458
$\mathcal{C}(\Sigma)$ es GDP_{ruido}	Σ accesible	2.1390
$\mathcal{C}(\Sigma)$ es GDP_{borrado}	Σ accesible	2.1390
$\mathcal{C}(\Sigma)$ es GDP_{ruido}	Σ observable	2.1390
$\mathcal{C}(\Sigma)$ es GDP_{borrado}	Σ observable	2.1390
$\mathcal{C}(\Sigma)$es GDP_{borrado}	Σ output_observable	1.6551

Por otro lado, a la vista de los resultados obtenidos, las reglas de asociación no establecen una equivalencia directa entre las propiedades de decodificación GDP_{ruido} y GDP_{borrado}, sino que ponen de manifiesto las condiciones adicionales necesarias para que una de ellas implique a la otra. Estas reglas, asociadas a los mayores valores de *lift* (7.1721 y 5.4331, respectivamente), no reflejan una relación de equivalencia entre ambas propiedades, sino una dependencia condicional: cada una puede derivarse de la otra únicamente mediante la incorporación de condiciones adicionales sobre la estructura del sistema. Así, la base de datos construida también presenta la funcionalidad de facilitar la búsqueda de contraejemplos. Es el caso del $(2, 1, 3)$-código convolucional generado mediante la representación ISO

$$A = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, D = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Este código es GDP_{ruido} pero no es GDP_{borrado}. Por el contrario, el $(3, 2, 3)$-código convolucional generado mediante la representación ISO

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 1 \end{pmatrix},$$

$$C = \begin{pmatrix} 0 & 1 & 1 \end{pmatrix}, D = \begin{pmatrix} 0 & 1 \end{pmatrix},$$

es GDP_{borrado} pero no es GDP_{ruido}.

Tabla VI
REGLAS INFORMATIVAS

Antecedente (Σ)	**Antecedente** ($\mathcal{C}(\Sigma)$)	**Consecuente**	***Lift***
A invertible, Σ output_obs., Σ accesible, Σ observable	A invertible, $\mathcal{C}(\Sigma)$ observable y GGDP_{borrado}	$\mathcal{C}(\Sigma)$ GDP de ruido	7.1721
Σ output_obs., A invertible, Σ accesible, Σ observable	Σ output_obs., $\mathcal{C}(\Sigma)$ observable y GDP_{ruido}	$\mathcal{C}(\Sigma)$ GDP_{borrado}	5.4331
A invertible, Σ accesible, Σ observable	$\mathcal{C}(\Sigma)$ observable y GDP_{ruido}	$rk(A) = \delta$	3.0458
A invertible, Σ accesible, Σ observable	$\mathcal{C}(\Sigma)$ observable y GDP_{ruido}	Autovalores de A no nulos	3.0458
A invertible		$rk(A) = \delta$, Autovalores de A no nulos	3.0458
Autovalores de A no nulos		A invertible, $rk(A) = \delta$	3.0458
$rk(A) = \delta$		A invertible, Autovalores de A no nulos	3.0458
Σ accesible, Σ observable	$\mathcal{C}(\Sigma)$ observable	$rk(\Phi) = rk(\Omega)$	2.6228
$rk(B) = \delta$, Σ observable		$rk(\Phi) = rk(\Omega)$	2.6228
Σ accesible, $rk(C) = \delta$		$rk(\Phi) = rk(\Omega)$	2.6228
$rk(B) = \delta$, $rk(C) = \delta$		$rk(\Phi) = rk(\Omega)$	2.6228
$rk(B) = \delta$		Σ accesible	2.1390
$rk(C) = \delta$		Σ observable	2.1390
$rk(\Phi) = rk(\Omega)$, Σ observable		Σ accesible	2.1390
$rk(\Phi) = rk(\Omega)$, $rk(C) = \delta$		Σ accesible	2.1390
$rk(\Phi) = rk(\Omega)$, Σ accesible		Σ observable	2.1390
$rk(\Phi) = rk(\Omega)$, $rk(B) = \delta$		Σ observable	2.1390

V. CONCLUSIONES

La construcción de códigos convolucionales con buenas propiedades estructurales y de decodificación sigue siendo un desafío en la teoría moderna de la codificación. En este trabajo se presenta una aproximación al estudio de los códigos convolucionales basada en el descubrimiento automatizado de conocimiento mediante técnicas de IA. A través de la generación y análisis masivo de representaciones ISO y sus propiedades estructurales, se han identificado relaciones y conjeturas que conectan características algebraicas y dinámicas de los sistemas con los códigos convolucionales. Este enfoque aporta nuevas estrategias eficientes para la construcción de códigos convolucionales con propiedades deseables de observabilidad, decodificación y distancia.

AGRADECIMIENTOS

Esta investigación resulta del Proyecto Estratégico *Ciencia de datos para un modelo de inteligencia artificial en ciberseguridad* (C073/23), fruto del convenio de colaboración suscrito entre el Instituto Nacional de Ciberseguridad (INCIBE) y la Universidad de León. Esta iniciativa se realiza en el marco de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation).

REFERENCIAS

[1] Akhavian, R. y Behzadan, A. H.: "Automated knowledge discovery and data-driven simulation model generation of construction operations", en *Proceedings of the 2013 Winter Simulation Conference*, pp. 3026–3037, 2013.

[2] Gukov, S., Halverson, J. y Ruehle, F.: "Rigor with machine learning from field theory to the Poincaré conjecture", en *Nature Reviews Physics*, vol. 6, pp. 310–319, 2024.

[3] Wu Y., et al.: "Autoformalization with large language models", en *Advances in Neural Information Processing Systems*, vol. 35, pp. 32353–32368, 2022.

[4] Hales, T. C.: "Exposé Bourbaki 1086 : Developments in formal proofs", en *Astérisque*, vol. 367, pp. 387-410, 2015.

[5] Alama J., Heskes T., Kühlwein D., Tsivtsivadze E. y Urban J.: "Premise selection for mathematics by corpus analysis and kernel methods", en *Journal of Automated Reasoning* , vol. 52, pp. 191–213, 2014.

[6] Ruehle, F.: "Data science applications to string theory", en *Physics Reports*, vol. 839, pp. 1–117, 2020.

[7] He, Y. H.: "Machine Learning in Pure Mathematics and Theoretical Physics", por *World Scientific*, 2023.

[8] DeCastro-García, N., Muñoz Castañeda, A. L. y Carriegos, M. V.: "Construction of observable and MDP convolutional codes with good decodable properties by I/S/O representations", to appear in *International journal of Applied Mathematics and Computer Science*, Vol. 36, No. 1,2026.

[9] York, E.V.: "Algebraic description and construction of error correcting codes, a systems theory point of view", en *Ph. D. dissertation Univ. of Notre Dame*, 1997.

[10] Rosenthal, J.: "An Algebraic Decoding Algorithm for Convolutional Codes", en *Dynamical Systems*, pp. 343–360, 1999.

[11] McEliece, R. J., "A public-key cryptosystem based on algebraic coding theory", DSN Progress Report, 42–44, 114–116. Jet Propulsion Laboratory, California Institute of Technology, 1978.

[12] Almeida, P., Beltrá, M., Napp, D., and Sebastião, C.: "Smaller Keys for the McEliece Cryptosystem: A Convolutional Variant with GRS Codes", arXiv:2104.06809 [cs.IT], 2023.

[13] Rosenthal, J. y Smarandache, R.: "Maximum distance separable convolutional codes", en *Applicable Algebra in Engineering, Communication and Computing*, vol. 10, n. 1, pp. 15-32, 1999.

[14] Gluesing-Luerssen, H., Rosenthal, J. y Smarandache, R.: "Strongly-MDS convolutional codes", en *IEEE Transactions on Information Theory*, vol. 52, n. 2, pp. 584-598, 2006.

[15] Hutchinson, R., Rosenthal, J. y Smarandache, R.: "Convolutional codes with maximum distance profile", en *Systems & Control Letters*, vol. 54, n. 1, pp. 53-63, 2005.

[16] Rosenthal, J., Schumacher, J.M. y York, E.V.: "On behaviors and convolutional codes", en *IEEE Transactions on Information Theory*, vol. 42, n. 6, pp. 1881-1891, 1996.

[17] Climent, J.-J., Napp, D. y Requena, V.: "An algorithm to compute a minimal input-state-output representation of a convolutional code", en *Linear Algebra and its Applications*, vol. 721, pp. 715-735, 2025.

[18] Kozachenko, L.F., Leonenko, N.N.: "Sample estimate of the entropy of a random vector", en *Problemy Peredachi Informatsii*, vol. 23, n. 2, pp. 95–101, 1987.

[19] Pedregosa et al.: "Scikit-learn: Machine Learning in Python", en *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.

[20] Agrawal, R., Imieliński, T. y Swami, A.: "Mining association rules between sets of items in large databases", en *ACM SIGMOD Record*, vol. 22, n. 2, pp. 207-216, 1993.

[21] Agrawal, R. y Srikant, R.: "Fast Algorithms for Mining Association Rules", en *Proceedings Of The 20th International Conference On Very Large Data Bases (VLDB)*, pp. 487-499, 1994.

[22] Raschka, S.: "MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack", en *Journal Of Open Source Software*, vol. 3, n. 24, pp. 638, 2018.

[23] Escudero García, D., DeCastro-García, N. y Muñoz Castañeda, A. L.: "total_F2", en *https://drive.google.com/file/d/1MKfGiXL2yU-hqsmkcmlrOWZ_oR9Y1lHa/view?usp=drive_link*, 2025.

[24] Lieb, J.: "Counting Polynomial Matrices Over Finite Fields: Matrices with Certain Primeness Properties and Applications to Linear Systems and Coding Theory", por *BoD–Books on Demand*, 2017.

[25] Bernstein, D. S.: "Matrix Mathematics: Theory, Facts and Formulas", por *Princeton University Press*, 2009.

[26] Kailath, T.: "Linear Systems", en *Prentice-Hall*, Englewood Cliffs, NJ, 1980.

Reimagining Proof-of-Work with Elliptic Curve Discrete Logarithms

Ghazaleh Keshavarzkalhori, Biel Castellarnau, Jordi Herrera-Joancomartí, Cristina Pérez-Solà
Departament d'Enginyeria de la Informació i les Comunicacions
Universitat Autònoma de Barcelona (UAB)
Ghazaleh.Keshavarzkalhori@uab.cat, Biel.Castellarnau@uab.cat, Jordi.Herrera@uab.cat, Cristina.Perez@uab.cat

***Abstract*—Proof-of-work (PoW) forms the foundation of blockchain technology and plays a pivotal role in securing decentralized networks. Most existing PoW schemes rely on hash-based computations, which, while effective for achieving consensus, perform work that has no intrinsic utility beyond the protocol itself. In this paper, we first review the fundamental properties required of PoW mechanisms and then propose a new scheme based on the Elliptic Curve Discrete Logarithm Problem (ECDLP) that satisfies these criteria. We further analyze the computational complexity of our approach and compare its performance with conventional hash-based schemes. Our results show that it is feasible to design a secure PoW mechanism grounded in a well-studied mathematical problem, offering an alternative to purely hash-based work while preserving the properties essential to blockchain consensus.**

***Index Terms*—proof-of-work, ECDLP, blockchain**

I. Introduction

Proof-of-work (PoW) mechanisms enable a prover to present evidence of having performed a prescribed amount of computational effort. Asymmetry is at the core of proof-of-work design: verification of the performed work is highly efficient, whereas generating it requires substantially greater computational cost.

Although PoW is now predominantly associated with blockchain consensus protocols, the concept predates blockchain applications and was originally proposed in the context of mitigating abuse in distributed systems, such as spam prevention and access throttling [1].

Most existing PoW constructions employ cryptographic hash functions as their core computational task. These designs benefit from well-understood security properties and straightforward verification. However, hash-based schemes inherently require computations with no value beyond the proof itself. This characteristic has motivated research into PoW mechanisms that introduce additional structure or potential utility, while retaining the asymmetry required for practical deployment.

In this work, we examine the abstract properties that a generic PoW scheme must satisfy and explain how these properties manifest in the context of blockchain systems. Based on this analysis, we propose a new PoW construction derived from the Elliptic Curve Discrete Logarithm Problem (ECDLP). The scheme retains efficient verification and satisfies the properties necessary for blockchain mining.

The contributions of this paper are thus the enumeration of the requirements for PoW mechanisms in blockchain-specific contexts, the design of a scheme instantiated from ECDLP that meets these requirements, and the analysis of the limitations of this design.

The remainder of the paper is organized as follows. Section II reviews the necessary background, including the ECDLP and standard mechanisms for hashing to elliptic-curve points. Section III introduces proof-of-work and outlines the properties required of PoW schemes in blockchain settings. Section IV then presents our ECDLP-based PoW construction and Section V analyzes its computational complexity. Then, Section VI surveys related work and highlights the main differences relative to our proposal. Finally, Section VII offers concluding remarks and directions for future research.

II. Background

This section presents the essential background for building our proposal, focusing on the Elliptic Curve Discrete Logarithm Problem (ECDLP) and the RFC 9380 procedures for generating elliptic-curve points from arbitrary data.

A. The Elliptic Curve Discrete Logarithm Problem

Cyclic groups over elliptic curves may be defined by selecting a generator point, G, and taking all the powers of G under the elliptic-curve group law:

$$\mathbb{G} = \{P \text{ such that } P = G^x, \forall x \in \mathbb{Z}_q\}$$

where q is the order of the generator.

Solving the ECDLP in this context refers to the problem of finding the element x used to generate a given point $P \in \mathbb{G}$. While this multiplicative notation aligns more directly with the terminology of the discrete logarithm problem, we adopt additive notation for the remainder of this work.

Since ECDLP is known to be a computationally hard problem, many cryptographic schemes hide secret values as the value used to generate a public point to make it computationally infeasible for any entity to get access to the secret.

The **naive method** of breaking the ECDLP is to try all possible $x \in \mathbb{Z}_q$, computing $x \cdot G$, up until one coincides with P.

Shanks' algorithm [2] enhances the efficiency of this attempt by searching for points that would add up to P to which we know the field elements. Doing so reduces the search for x to only $\sqrt{q}$, making the search much more efficient. Although this effort reduces the search to only the square root of the size of the field, the storage needed for the algorithm is of order $O(\sqrt{q})$.

Pollard's rho-algorithm [3], on the other hand, removes the need for such large storage and guarantees the search for x with $\sqrt{q}$. This is done by choosing a random function that has the same domain and range. In each iteration, a field element

is transformed into another using this function. After each transformation, the algorithm checks whether the resulting element satisfies a predefined condition that qualifies it as a distinguished point (for example, having a fixed number of zero-leading bits). If the condition is met, the point, along with its associated data, is stored in a shared list accessible to all parallel processes. The storage needed is $O(d)$ where d is distinguished points in the list. A collision is detected when a newly computed distinguished point matches one already in the list. This collision can then be used to solve for the discrete logarithm.

Given G and $Q = k \cdot G$ points on the elliptic curve, where k is a positive integer and the discrete algorithm we want to find (which must be less than order q), the Pollard's rho-algorithm is defined as:

1) Each point R in the sequence is represented as: $R = aG + bQ$
2) If R is a distinguished point, we store the tuple (R, a, b) in the shared list.
3) We compute the next point using $(R, a, b) = nextPoint(G, Q, R, a, b)$ and start again until a distinguished point is found.
4) Let the elliptic curve group $E(\mathbb{Z}_p)$ be partitioned into three subsets: S_1, S_2, S_3. The $nextPoint$ function is defined as:

$$f(R_i) = \begin{cases} Q + R_i, & \text{if } R_i \in S_1 \\ 2R_i, & \text{if } R_i \in S_2 \\ G + R_i, & \text{if } R_i \in S_3 \end{cases}$$

5) When a collision is found, where $R = aG + bQ$ is the current point and $R^* = cG + dQ$ is a previous point saved on the shared list, we can solve for k as: $k = \frac{a-c}{d-b} \mod q$, provided $d \not\equiv b \pmod q$. This yields the discrete logarithm k such that $Q = kG$.

A variety of alternative methods have been developed to enhance the efficiency of solving the ECDLP in classical computation, as comprehensively discussed by Hankerson et al. [4]. These algorithms are still computationally heavy for processing practical secure curves.

Pollard's lambda method (also known as the ***kangaroo method***) is more efficient than Pollard's rho method in cases where the discrete logarithm x is known to lie within a restricted range, typically $x \in [0, b)$ with $b < q$. In practical implementations, such a constraint allows for faster computation. The method can be visualized as a number line of points $G, 2G, 3G, \ldots$. Imagine two kangaroos on this line: a *tame kangaroo* and a *wild kangaroo*. The goal is for the wild kangaroo to catch the tame one.

1) The *tame kangaroo* starts at a known point $R_0 = bG$ and performs a sequence of jumps. The next location of the kangaroo is determined by $R_{i+1} = R_i + f(R_i)P$ where the function $f(R_i)$ determines the jump distance.
2) The *wild kangaroo* starts at $R'_0 = Q$ and performs jumps using the same function $f(R)$. If the wild kangaroo's total distance exceeds that of the tame kangaroo, it restarts from a new point $R'_0 = Q + zG$, for a small known vale of z.

The process continues until the wild kangaroo lands on the same point as the tame kangaroo, at which point the discrete logarithm can be calculated from the difference of the distances traveled by the two kangaroos.

B. Hashing to Elliptic Curves (RFC 9380)

In order to ensure that a calculated point belongs to a curve without excessive computation, the mechanism described in *RFC 9380: Hashing to Elliptic Curves* [5] is used.

This RFC defines a `hash_to_curve(msg: bytes)` function that maps a message to a point P on a curve. The process begins with `hash_to_field(msg, count)`, where `count=` 2. It returns two field elements in a finite field $\mathbb{F}_p$. This is done using an `expand_message(msg,DST,l` function. `msg` is a byte string containing the message to hash, `DST` is a byte string that acts as a domain separation tag and `l` is the number of bytes to be generated. The function produces a uniformly random byte string using a cryptographic hash function H that outputs b bits.

Next, each field element is passed to `map_to_curve`(v), which returns a point Q on the curve. The RFC focuses on short Weierstrass curves, allowing the use of the Simplified Shallue–van de Woestijne (sSWU) [5] method. This requires curve parameters A, B with $AB \neq 0$, and a constant Z that must be a non-square in $\mathbb{F}_q$. Since two field elements were returned in the previous function, a point for each element will be created. These two points are added to get the final desired point R.

Finally, the `clear_cofactor(R)` function ensures the point lies in the prime-order subgroup G. In our case, since the curve has prime order n, the cofactor is $h = 1$, so $G = E(\mathbb{F}_q)$ and no clearing is needed.

The `hash_to_curve` algorithm is thus used to convert an arbitrary-length byte string `msg` into P, a point in a curve E:

1) Extract the field elements:

$$\mathbf{u} = \texttt{hash_to_field}(\texttt{msg}, 2)$$

2) Transform the field elements into points in the curve[1]:

$$R_0 = \texttt{map_to_curve_sSWU}(u[0]),$$
$$R_1 = \texttt{map_to_curve_sSWU}(u[1])$$

where sSWU is the Simplified Shallue–van de Woestijne mapping for $y^2 = x^3 + Ax + B$.

3) Add the points:

$$P = R_0 + R_1$$

4) Return the point of the curve P.

III. Proof-of-work for blockchain

Proof of Work (PoW) was designed to ensure that a certain amount of computational effort has been expended to complete a task that is easy to verify. This asymmetry between expensive computation and cheap verification is central to its security model, especially in blockchain systems like Bitcoin.

The formal description of a PoW scheme comprises three algorithms:

[1]The RFC generates two points and sums them, rather than producing a single point directly, to ensure that the resulting output is statistically close to a uniform distribution.

- `c = Gen(`d`)`: A deterministic algorithm with unpredictable output that generates a challenge c for a given difficulty d.
- `s = Solve(`c`)`: A computationally expensive algorithm that attempts to solve the challenge, producing a solution s.
- `TRUE/FALSE = Verify(`c, s`)`: A fast algorithm that checks whether s is a valid solution to c.

The key requirement is that `Gen` and `Verify` must be fast, while `Solve` must be computationally expensive. This ensures that the solution s shows that a certain amount of work was done. Moreover, `Verify` must be able to confirm that c was generated in a way that required this effort, preventing trivial or precomputed challenges.

When deployed in a blockchain environment, a proof-of-work mechanism is required to have the following properties:

- **Difficulty adjustment:** When additional computational power enters the network, the average time required to solve a challenge decreases; conversely, when mining power drops (because miners leave or hardware becomes unavailable) the average time increases. A reliable difficulty-adjustment mechanism is therefore required to keep the solving times of the challenge c near their target value, thereby ensuring stable block production despite fluctuations in total computational capacity. Moreover, the adjustment must be sufficiently granular to keep actual block times close to the theoretically defined target.
- **Non-reusability and input uniqueness:** The solution s must be intrinsically linked to a specific instance of the problem (e.g., a specific block header). This prevents reuse or precomputation of solutions, ensuring that each PoW is fresh and tied to the current state of the chain.
- **Linkability to the block:** The challenge c must include the previous block's hash and current transaction data, making each PoW cryptographically linked to the block it secures. This chaining prevents miners from predicting or preparing future blocks in advance.
- **ASIC resistance (desirable):** Ideally, PoW should be resistant to specialized hardware (ASICs), promoting decentralization by allowing general-purpose CPUs to compete fairly. This supports the "one-CPU-one-vote" [6] philosophy and reduces the risk of mining centralization.

Together, these properties make PoW a robust mechanism for securing decentralized systems, ensuring fairness, verifiability, and resistance to manipulation.

In Bitcoin, PoW involves finding a nonce such that the hash of the block header meets a difficulty target (e.g., has a certain number of leading zeros):[2]

$$hash(\text{ header } || \text{ nonce }) < \text{difficulty_target}$$

Generating such a hash is computationally expensive. However, once a valid hash is found, it can be efficiently and independently verified by any node, incurring minimal verification cost.

Although Bitcoin's proof-of-work effectively demonstrates that significant computational effort has been expended and allows for rapid verification, the work itself consists solely of performing hashing operations, which have no function beyond satisfying the requirements of the proof-of-work protocol.

[2]Technically, the nonce is included in the header, but we have opted for this formulation to simplify the explanation.

IV. ECDLP-BASED PROOFS-OF-WORK

The overall goal of the PoW algorithm presented in this paper is to generate ECDLP instances whose difficulty can be adjusted while remaining cryptographically bound to the specific block being mined (see Section III). Adjustable difficulty is achieved by varying the underlying cyclic group in which the ECDLP is defined, accomplished through the careful selection of the elliptic curve on which the problem is instantiated. The binding to a particular block is ensured by deriving the target point for the discrete logarithm computation directly from the current block header.

In general, PoW blockchains define epochs (of length Δ_{Epoch} blocks) during which the difficulty d of the PoW is constant. At the end of each epoch, difficulty is adjusted to account for computational power changes in the network (see Fig 1).

The rest of this section covers how blocks are generated, solved, and verified. Phase 0 first elaborates on how the difficulty is set at the start of each epoch, while phases 1-3 dig deeper into how a challenge is set for the miner to solve, how it solves it, and how the rest of the miners can verify the block upon receiving it.

Phase 0: Difficulty Setting

This phase is only carried out at the start of a new epoch. The goal of this phase is to select a, b and p defining an elliptic curve over which the ECDLP is going to be solved:

$$E : y^2 = x^3 + ax + b \mod p$$

In our protocol, selected curves are always of prime order. This guarantees that all points on the curve lie in a single cyclic group, and that any non-identity point can serve as a generator. Consequently, every ECDLP instance is bounded by the order of the curve, which is chosen to reflect the desired difficulty.

In this phase, we employ the algorithm proposed by Bröker et al. [7] to generate a Hilbert class elliptic curve [3] containing exactly q points, where q is prime number whose size is defined by the current difficulty:

1) Let `data` be the previous block's header.
2) $h = hash(\texttt{data})$
3) $n = 2^{2d} + (h \mod 2^{2d})$
4) $q = ComputeNextPrime(n)$
5) $(a, b, p) = ConstructEllipticCurve(q)$

The protocol then selects a point on the curve to be used as the generator for the ECDLP instance. This is accomplished via the `hash_to_curve` algorithm (Section II-B), using the block header of the last block from the previous epoch as the `msg` and `ECDLP_NEW_EPOCH` as the domain-separation tag.

Once the elliptic curve and generator for the epoch are defined, miners derive the challenge for the current block.

[3]To the best of our knowledge, no algorithms more efficient than the classical approaches—such as Shanks's method or Pollard's rho algorithm—are currently known for solving the ECDLP on these curves, thereby supporting their suitability and security for our intended application.

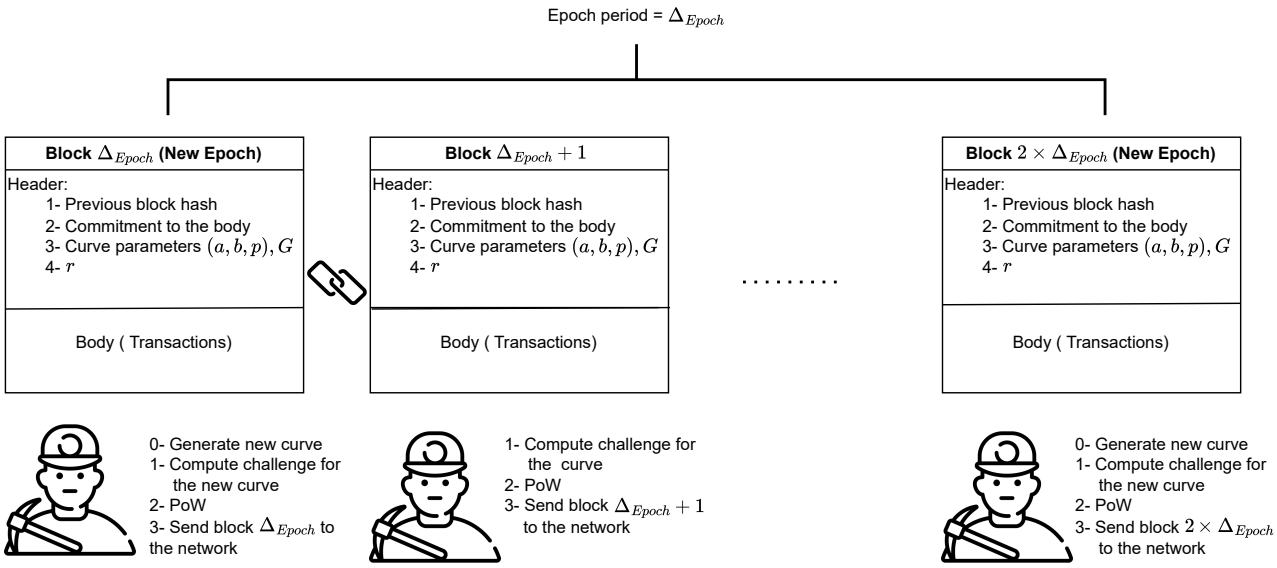


Fig. 1. The structure and types of blocks existing in the blockchain.

Phase 1: Challenge Computation

In the proposed PoW, the challenge for each block is computing the discrete logarithm of a point P in the current epoch's elliptic curve. The point P is deterministically derived from the block to ensure likability and non-re-usability (as defined in Section II-A).

Specifically, the point P is derived from calling the `hash_to_curve` algorithm (Section II-B), using the current block header as the `msg` and `ECDLP_NEW_BLOCK` as domain separation tag:

1) Let `msg` be the current block's header, with the field r set to 0.
2) $P = hash_to_curve(\texttt{msg}, \texttt{ECDLP_NEW_BLOCK})$

Phase 2: Proof of Work

In this phase, the miner solves the discrete logarithm of the point P over the epoch's curve, that is, determines the integer r such that $P = r \cdot G$ over the curve E.

To do so, he may use one of the algorithms for solving the ECDLP presented in Section II-A.

The first miner to solve the challenge will then update the header of the block to include the solution of the challenge r and broadcast its block to the network.

Phase 3: Verification

Each miner receiving a new block must validate if the transactions are valid as well as check if the challenge of the block has been solved and created correctly. Miner M' receiving the block from M will carry out the following checks:

1) Check that the curve over which the ECDLP is solved, is correctly derived from the difficulty adjustment algorithm.
2) Check that the challenge P is correctly derived from the current block header. To recompute P, the validators set $r = 0$ in the header when calling the `hash_to_curve` function.
3) Check that the original r is a valid solution of the ECDLP instance, that is, $P = r \cdot G$.

If any of the checks fail, the validation process is terminated and the block is rejected. If all checks succeed, the block is accepted into the network, and mining proceeds on top of it.

V. Complexity analysis

We now analyze the computational complexity of the proposed PoW algorithm and its dependence on the difficulty parameter. Let E be the elliptic curve selected for the current epoch, with $q = |E(\mathbb{F}_p)|$ denoting the number of group elements; the ECDLP is solved over this group. Under the protocol definition, q is chosen as the smallest prime greater than $2^{2d} + (h \mod 2^{2d})$ where h denotes the hash of the last block in an epoch.

Solving the ECDLP for a point $P \in E$ incurs a time complexity of $O(\sqrt{q})$. Under the simplifying assumption that $q \approx 2^{2d}$, block generation therefore requires $O(2^d)$ time and uses constant space $O(1)$.

For comparison, in Bitcoin the computational complexity of the proof-of-work mechanism is given by $O(2^d)$ where d denotes the network difficulty parameter. The dominant operation consists of repeated evaluations of the double SHA-256 hash function, as opposed to elliptic curve scalar multiplications used in the proposed algorithm.

It is important to note that the computational complexity of auxiliary algorithms employed during block generation, such as `HashtoCurve` and `ConstructEllipticCurve` is negligible compared to the PoW. These algorithms exhibit complexities on the order of $O((\log q)^\alpha)$ where $\alpha \in \{1, \ldots, 4\}$, which is insignificant relative to the exponential cost of PoW.

VI. Related work

Several mechanisms have been proposed in the literature to demonstrate the consumption of different types of computational resources, including proof of memory [21], proof of storage [22], and proof of bandwidth [23].

Mechanism	Family	Problem	Example
Classic hashing [1]	Hash-based	Find a hash below a target value	D: Bitcoin (SHA256) [6]
Memory-hard hashing	Hash-based	Find a hash below a target value	D: Litecoin (Scrypt) [8], [9]
Equihash [10]	Hash-based	Solve generalized birthday problem	D: Zcash [11]
Prime chains	Number-theoretic	Find chains of prime numbers	D: Primecoin (Cunningham/Bi-twin chains) [12], Riecoin (prime constellations) [13]
Integer factorization	Number-theoretic	Factor a composite number	D: Factorn [14], [15]
Discrete logarithm	Number-theoretic	Solve a discrete logarithm	T: Meneghetti et al. [16]
Repeated squaring	Iterative function-based	Sequential modular arithmetic	D: Chia [17]
Cuckoo Cycle [18]	Graph-based	Find a cycle of fixed length in a bipartite graph	D: Grin [19]
Dominating set	Graph-based	Find valid dominating set	T: ScaloWork [20]

TABLE I
EXISTING PROOF-OF-WORK MECHANISMS. IN THE EXAMPLE COLUMN, D STANDS FOR A DEPLOYED PROJECT, WHEREAS T STANDS FOR A THEORETICAL PROPOSAL (NOT YET DEPLOYED IN AN EXISTING BLOCKCHAIN PROJECT).

In this work, we focus exclusively on proof-of-work (PoW), which requires a participant to perform a computationally intensive task that can be efficiently verified. Existing PoW constructions can be broadly categorized according to the nature of the underlying computational problem (see Table I for a summary). Hash-based schemes remain the most widely deployed due to their simplicity and well-understood security properties. Other approaches include number-theoretic schemes, which rely on arithmetic problems such as finding large primes; iterative function–based schemes, which require repeated application of a function with certain cryptographic properties; and graph-based schemes, which use hard problems defined over graph structures. Our work builds on this landscape by proposing a PoW mechanism based on the elliptic curve discrete logarithm problem.

A. DLP-based proof of work

Although relatively few, some prior works have explored the design of proof-of-work schemes based on the discrete logarithm problem.

Hastings et al. [24] propose a PoW algorithm to motivate research be done towards advancing the computation of DLPs. In their work, they take advantage of Pollard's rho algorithm to find distinguished points with a specific number of starting zeros in their bit representations. Hastings suggests to set the starting point of the Pollard's algorithm to the hash of the previous block and a nonce to avoid reusability and provide freshness for the miners as well as setting an upper bound for the number of steps carried out for each hash and nonce. This will assure that the validation of the PoW will be fast and thorough.

The ECDLP-Based PoW Model [16] proposes having two types of blocks. Epoch blocks select elliptic-curve parameters. In addition to the header and transactions, the miner computes a prime p, an elliptic curve E, and a base point $G \in E(\mathbb{F}_p)$. As part of the proof-of-work, the miner must provide integers N_1 and N_2 such that N_iG are valid points on E. These blocks appear every 2016 blocks, mirroring the difficulty-adjustment schedule of Bitcoin. Standard blocks contain only the usual header and transactions and inherit p, E, and G from the last epoch block.

The validation conditions are based on a deterministic function $\mathsf{P_Gen}$ mapping a hash to a point on E. A block is valid only if two integers (N_1, N_2) satisfy

$$\mathsf{P_Gen}(H(h_{prev}), E) = N_1G,$$
$$\mathsf{P_Gen}(H(M), E) = N_2G.$$

where h_{prev} is the previous header and M the Merkle root.

The parameter p is generated so that the resulting curve depends on the current difficulty d. Starting from the hash of the previous header h, a candidate prime is computed as $p = h \bmod 2^{2d}$; then the function $\mathsf{Next_Prime}$ is applied to obtain the next smallest prime at least this large. Primes with undesirable structure (e.g., Crandall, generalized Mersenne, or Montgomery-friendly forms) are rejected, and the process is repeated with re-hashing until an acceptable p is found.

Given p and h, the curve coefficients are defined by $A_E = H(h + i)$ and $B_E = H(A_E)$ for an integer counter i, which produce the curve

$$E : y^2 = x^3 + A_Ex + B_E \pmod p.$$

The curve is accepted only if it meets the standard security properties, including a prime group order distinct from p, embedding degree exceeding 20, and a complex multiplication discriminant larger than 2^{40}. If any condition fails, i increases and the coefficients are recomputed.

The generator point is finally obtained as

$$G = \mathsf{P_Gen}(H(p \parallel A_E \parallel B_E),\, E)\,.$$

B. Comparison with existing protocols

Previous works have explored alternatives to hash-based designs by using elliptic curve structures, with the objective of making the computational effort more meaningful. However, these protocols are significantly different in how they define mining tasks, manage curve generation, and adjust difficulty.

The proposed approach in [4] does not require miners to solve the Elliptic Curve Discrete Logarithm Problem (ECDLP); instead, the objective is limited to identifying distinguishable points on the curve. To achieve this, the system targets large elliptic curves while adjusting the difficulty so that the challenge remains computationally feasible—miners

only need to locate unique points for each block. This design is appealing because, if applied at scale, it would enable the network to collectively work on curves of near-practical size. However, the authors assume that a curve will eventually be solved after sufficient cumulative effort, yet they do not specify a mechanism for transitioning to a new curve once the current one becomes unusable, leaving a gap in the protocol's adaptability.

In [16] curve generation is performed by deterministically constructing a secure elliptic curve over a prime field $\mathbb{F}_p$, where the prime p is computed as the next prime greater than $h \mod 2^{2d}$, h being the hash of the previous block. Although this process is deterministic, the resulting prime may be relatively small, and the procedure iterates until a curve satisfying security requirements is obtained—without any explicit upper bound on the number of iterations. On the other hand, each miner receiving an epoch block is required to validate both the primality of p and the security properties of the curve, which introduces significant computational overhead.

Furthermore, while the design assumes that network difficulty is tied to operations on elliptic curve points, it incorrectly associates this difficulty with the size of the prime field rather than the actual group order of the curve. According to Hasse's theorem, the group order lies within the interval $(p+1-2\sqrt{p},\ p+1+2\sqrt{p})$, meaning that the field size and group size are not equivalent hence difficulty is not the same as the scheme is setting it to be.

VII. Conclusions and future work

In this paper, we have introduced an ECDLP-based proof-of-work mechanism as an alternative to traditional hash-based schemes. Using elliptic curve operations and the RFC9380 deterministic hash-to-curve mapping, the proposed design ensures that the blocks are linked and maintain the asymmetry required for a PoW (a challenge hard to solve and a verification easy to execute).

By introducing an elliptic curve–based challenge, the computational effort serves a dual purpose: not only to secure the network, but also to generate new elliptic curves that could have value in other domains[4]. Unlike traditional hash-based schemes, which produce results with no utility beyond consensus, elliptic curve challenges create mathematical artifacts that could help evaluate the hardness of elliptic-curve groups (or the security margins of specific parameters) and the performance of algorithms.

With respect to the design, future work will first focus on evaluating the granularity provided by our scheme to determine whether it is sufficient for blockchain applications, particularly in maintaining stable block production under varying network conditions. Additionally, subsequent research will involve implementing and testing a prototype of the protocol under real-world conditions to assess its practical performance, efficiency, and robustness.

Acknowledgements

This work is funded by the projects PID2021-125962OB-C33 SECURING/NET and PID2024-156914OB-C43 SAFE-BLOCKCHAIN, funded by the Ministerio de Ciencia e Innovación, la Agencia Estatal de Investigación and the European Regional Development Fund (ERDF); the DANGER Strategic Project of Cybersecurity C062/23, funded by the Spanish National Institute of Cybersecurity through the European Union - NextGenerationEU and the Recovery, Transformation and Resilience Plan; and the Catalan Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) grant SGR2021-00643.

[4]For example, this effort could be applied to the exploration of prime gaps, such as investigating intervals that might violate Bertrand's postulate while selecting prime parameters for each difficulty adjustment.

References

[1] A. Back *et al.*, "Hashcash-a denial of service counter-measure," 2002.
[2] D. Shanks, "Class number, a theory of factorization, and genera," in *Proc. Symp. Math. Soc., 1971*, vol. 20, 1971, pp. 415–440.
[3] J. M. Pollard, "Monte carlo methods for index computation," *Mathematics of computation*, vol. 32, no. 143, pp. 918–924, 1978.
[4] D. Hankerson, S. Vanstone, and A. Menezes, *Guide to elliptic curve cryptography*. Springer, 2004.
[5] A. Faz-Hernandez, S. Scott, N. Sullivan, R. Wahby, and C. Wood, *RFC 9380 Hashing to Elliptic Curves*. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9380.pdf
[6] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Available at SSRN 3440802*, 2008.
[7] R. Broker and P. Stevenhagen, "Constructing elliptic curves of prime order," *arXiv preprint arXiv:0712.2022*, 2007.
[8] J. Choe, T. Moreshet, R. I. Bahar, and M. Herlihy, "Attacking memory-hard scrypt with near-data-processing," in *Proceedings of the International Symposium on Memory Systems*, 2019, pp. 33–37.
[9] L. Foundation, "Litecoin developer guide: Proof of work," https://litecoin.info/docs/key-concepts/proof-of-work, 2024.
[10] A. Biryukov and D. Khovratovich, "Equihash: Asymmetric proof-of-work based on the generalized birthday problem," *Ledger*, vol. 2, pp. 1–30, 2017.
[11] D. Hopwood, S. Bowe, T. Hornby, N. Wilcox *et al.*, "Zcash protocol specification," *GitHub: San Francisco, CA, USA*, vol. 4, no. 220, p. 32, 2016.
[12] S. King, "Primecoin: Cryptocurrency with prime number proof-of-work," *July 7th*, vol. 1, no. 6, 2013.
[13] T. R. Developers, "The stella pow," https://riecoin.xyz/Guides/Stella/, 2025.
[14] F. development team, "Factorn docs," https://docs.fact0rn.io/, 2024.
[15] E. Liones, "FACT0RN: Integer Factorization as Proof of Work," Tech. Rep., 2022.
[16] A. Meneghetti, M. Sala, and D. Taufer, "A new ecdlp-based pow model," *Mathematics*, vol. 8, no. 8, p. 1344, 2020.
[17] C. N. Inc, "Chia documentation," https://docs.chia.net/chia-blockchain/consensus/proof-of-time/, 2025.
[18] J. Tromp, "Cuckoo cycle: A memory bound graph-theoretic proof-of-work," in *International Conference on Financial Cryptography and Data Security*. Springer, 2015, pp. 49–62.
[19] G. development team, "Grin documentation," https://docs.grin.mw/about-grin/proof-of-work/, 2024.
[20] D. Chatterjee, A. Majumder, and S. Mazumdar, "Scalowork: Useful proof-of-work with distributed pool mining," *arXiv preprint arXiv:2504.14328*, 2025.
[21] M. Abadi, M. Burrows, M. Manasse, and T. Wobber, "Moderately hard, memory-bound functions," *ACM Transactions on Internet Technology (TOIT)*, vol. 5, no. 2, pp. 299–327, 2005.
[22] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM conference on Computer and communications security*, 2007, pp. 598–609.
[23] B. Chrobot, D. Lax, and D. O'Donovan, "Torcoin," MIT 6.858: Computer Systems Security, Fall 2014, 2014. [Online]. Available: https://css.csail.mit.edu/6.858/2014/projects/bchrobot-ynnad19-dereklax.pdf
[24] M. Hastings, N. Heninger, and E. Wustrow, "Short paper: The proof is in the pudding: Proofs of work for solving discrete logarithms," in *International Conference on Financial Cryptography and Data Security*. Springer, 2019, pp. 396–404.

M3Suite: Plataforma de Aprendizaje Federado Privado, Modular y Usable

Idoia Gamiz, Julen Bernabé-Rodríguez, Kristina Stemikovskaya, Miguel Esteras
TECNALIA, BRTA
Parque Científico y Tecnológico de Bizkaia, 700
48160 Derio, Bizkaia (España)
{idoia.gamiz, julen.bernabe, kristina.stemikovskaya, miguel.esteras}@tecnalia.com

***Resumen*—El Aprendizaje Federado permite entrenar modelos de inteligencia artificial de manera distribuida, evitando la centralización. Sin embargo, su implementación enfrenta desafíos significativos, tanto a nivel arquitectónico como en la protección de la privacidad. Para abordar estos problemas, este trabajo presenta M3Suite, una solución que combina modularidad, privacidad y usabilidad para facilitar la implementación de FL en diversos escenarios. M3Suite es adaptable a diferentes librerías de inteligencia artificial y ofrece distintos niveles de protección de la privacidad, desde la ausencia de mecanismos hasta la integración de técnicas avanzadas como la Privacidad Diferencial y el Cifrado Homomórfico, permitiendo adaptarse a las necesidades de cada caso de uso. Además, dispone de una interfaz de usuario que permite seleccionar entre diferentes modelos y niveles de privacidad, facilitando su uso por parte de usuarios sin conocimiento en inteligencia artificial o criptografía.**

***Index Terms*—Aprendizaje Federado, Privacidad, Cifrado Homomórfico, Privacidad Diferencial**

I. Introducción

El Aprendizaje Federado (FL, por sus siglas en inglés) [3] es una técnica que permite entrenar modelos de inteligencia artificial (IA) de manera distribuida, sin necesidad de centralizar los datos. Este enfoque es especialmente útil en escenarios donde la privacidad o las regulaciones impiden la compartición de datos personales. Sin embargo, su implementación conlleva una serie de desafíos, tanto a nivel arquitectónico como a nivel de vulnerabilidades de privacidad que dicha arquitectura puede presentar.

Por un lado, los desafíos arquitectónicos están relacionados con la necesidad de diseñar e implementar una infraestructura adecuada para cada caso de uso, que promueva el uso de FL sin necesidad de conocimiento sobre la tecnología subyacente [1]. Dependiendo del tipo de modelo que se desee entrenar, es necesario integrar FL con diferentes librerías, lo que impide la existencia de una solución única o estándar. Es necesario implementar herramientas accesibles que permitan a usuarios no expertos en IA implementar FL de manera sencilla. Además, la configuración de los componentes de FL, como los clientes, los servidores y los protocolos de comunicación, requiere conocimientos técnicos avanzados y una infraestructura bien diseñada.

Por otro lado, los desafíos relacionados con la protección de la privacidad son considerados prioritarios dentro de la comunidad científica [2]. Aunque en FL los datos permanecen en los dispositivos locales de los participantes, se ha demostrado que es posible extraer información sensible mediante ataques de inferencia o reconstrucción, utilizando los modelos intercambiados durante el proceso de entrenamiento. Aunque existen tecnologías como la Privacidad Diferencial (DP, por sus siglas en inglés) [6], el Cifrado Homomórfico (HE) [5] y la Computación Segura Multiparte (SMPC) [4], su implementación requiere conocimientos en criptografía y una adaptación específica para cada modelo. Por lo tanto, en la mayoría de los casos, su integración con FL es manual y dependiente del escenario, lo que limita su accesibilidad para usuarios sin conocimiento en criptografía.

Teniendo en cuenta los desafíos mencionados, los objetivos principales de este trabajo son:

- Abordar los desafíos arquitectónicos de FL mediante la implementación de un sistema modular y adaptable, capaz de integrarse con diferentes librerías de IA y ser aplicado a diferentes casos de uso.
- Mitigar los desafíos de privacidad, especialmente en escenarios donde es necesario garantizar una protección efectiva contra ataques que buscan inferir información sensible a partir de los modelos intercambiados.
- Mejorar la usabilidad de FL, facilitando su adopción por parte de usuarios no expertos en IA y criptografía.

Para lograr estos objetivos, este trabajo presenta M3Suite, una solución que aporta las siguientes contribuciones:

- Simplifica la implementación de FL mediante una arquitectura modular y configurable, que permite adaptarse a múltiples escenarios y requisitos.
- Integra técnicas de protección de la privacidad con diferentes niveles de seguridad, desde la ejecución sin mecanismos de protección hasta la aplicación de DP o HE. Esta capacidad de configuración permite adaptar el sistema a escenarios heterogéneos, en función de las necesidades de privacidad y rendimiento de cada caso.
- Facilita el uso de FL para usuarios sin experiencia en IA y criptografía, mediante una interfaz de usuario que permite seleccionar de forma sencilla las opciones de privacidad y modelos disponibles. Además, incluye documentación clara y ejemplos prácticos que promueven una adopción más accesible.

El resto de este artículo se organiza de la siguiente manera. En la Sección II, se compara M3Suite con trabajos del estado del arte en términos de modularidad, protección de la privacidad y usabilidad. En la Sección III se describe la herramienta M3Suite y se explica su arquitectura. La Sección IV presenta dos casos de uso que muestran la aplicación de M3Suite en escenarios reales. Finalmente, en la Sección V se exponen las conclusiones y las líneas de investigación futuras.

II. Trabajo relacionado

El FL ha evolucionado significativamente en los últimos años, dando lugar a una amplia gama de herramientas. A pesar de ello, la implementación de FL sigue enfrentando desafíos importantes en cuanto a modularidad y adapatabilidad a herramientas de IA, protección de la privacidad y usabilidad, especialmente en contextos donde se requiere acceso para usuarios no expertos. En la Tabla I se presentan las aportaciones de cada herramienta en estos términos.

En cuanto a modularidad, algunas herramientas ofrecen flexibilidad para integrarse con diferentes librerías de FL. Flower [9], desarrollada en la Universidad de Oxford y actualmente gestionada por ADAP-Technik GmbH, destaca por su versatilidad, ya que se puede utilizar junto a cualquier librería de IA, como por ejemplo, PyTorch, TensorFlow, scikit-learn e incluso NumPy. Asimismo, NVFlare [11], desarrollada por NVIDIA, también es compatible tanto con librerías de aprendizaje profundo como con algoritmos de IA tradicional, integrándose con diferentes herramientas.

Sin embargo, otras herramientas no son compatibles con cualquier herramienta de IA. PySyft [8], desarrollada por el proyecto OpenMined, y Federated AI Technology Enabler (FATE) [10], desarrollada por WeBank, son compatibles exclusivamente con las bibliotecas PyTorch y TensorFlow. En cambio, TensorFlow Federated (TFF) [7], desarrollada por Google, restringe su uso a la librería de TensorFlow. Esta limitación reduce la flexibilidad de estas herramientas para adaptarse a diferentes casos de uso y modelos, especialmente para trabajar con algoritmos de IA tradicionales.

En el ámbito de la privacidad, se han desarrollado diversas técnicas, como el uso de DP, HE y SMPC en la agregación. Flower [9] ofrece una implementación propia de DP y SecureSum, así como adaptabilidad para integrar otras librerías de DP como Opacus [12] o Diffprivlib [13]. NVFlare [11] ofrece DP, HE e intersección de conjuntos privados (PSI, por sus siglas en inglés), y PySyft [8] integra DP, HE y SMPC. Por otro lado, FATE [10] proporciona HE y SMPC, mientras que TFF [7] se limita al uso de DP.

Aunque algunas herramientas presentan soluciones efectivas de privacidad, son difíciles de integrar y requieren adaptaciones para cada caso de uso. Es decir, no siempre son generalizables a distintos tipos de modelos o conjuntos de datos, o hace falta conocimiento experto para saber ajustar los parámetros adecuadamente, lo que está directamente relacionado con la usabilidad de la herramienta.

En términos de usabilidad, Flower [9] ofrece documentación detallada y ejemplos prácticos, pero requiere conocimiento técnico en IA para añadir nuevos modelos y estrategias, así como en criptografía para aplicar DP y ajustar los parámetros adecuadamente. Asimismo, NVFlare [11] es altamente flexible, pero entender completamente su arquitectura puede requerir un esfuerzo y conocimiento inicial. PySyft [8] combina múltiples componentes, que requieren una comprensión profunda para configurarlos correctamente, dificultando la instalación y el despliegue para ciertos usuarios. FATE [10] destaca por ser una de las primeras soluciones orientadas al ámbito empresarial en este campo, pero su configuración puede resultar compleja debido a la gran cantidad de módulos y opciones que incluye. Por último, TFF [7] ofrece FL API, una interfaz de alto nivel que facilita a los desarrolladores de IA la integración de modelos existentes de TensorFlow en un entorno federado, sin necesidad de profundizar en los detalles técnicos de los algoritmos de FL.

En resumen, la mayoría de las herramientas están orientadas a usuarios con experiencia en IA y criptografía, lo que limita su adopción por parte de organizaciones sin formación técnica especializada. Esto conlleva una curva de aprendizaje elevada y una baja accesibilidad a la tecnología. A partir de esta revisión del estado del arte, este trabajo propone M3Suite, una herramienta que puede utilizarse como *FL-as-a-service* y que integra modularidad, protección de la privacidad generalizable y una interfaz de usuario accesible.

Tabla I
Trabajo relacionado. Herramientas existentes para Aprendizaje Federado.

Herramienta	Modularidad	Privacidad	Usabilidad
Flower [9]	✓	DP, SecureSum	×
NVFlare [11]	✓	DP, HE, PSI	×
PySyft [8]	Pytorch/TensorFlow	HE, DP, SMPC	×
FATE [10]	PyTorch/TensorFlow	HE, SMPC	×
TFF [7]	TensorFlow	DP	∼

III. M3Suite: Presentación de la herramienta

M3Suite es una solución diseñada para abordar los desafíos arquitectónicos y de privacidad que surgen en los escenarios de FL, con un enfoque particular en la mejora de la usabilidad. Permite a organizaciones y desarrolladores implementar FL de manera sencilla y segura, sin requerir un conocimiento previo en IA o criptografía.

III-A. Descripción

El sistema se ha construido siguiendo una arquitectura modular, lo que facilita su adaptación a diversos casos de uso. Además, le otorga flexibilidad para ofrecer distintos niveles de protección de la privacidad, que pueden activarse o no según los requisitos específicos de cada escenario. Estos niveles de privacidad se materializan en tres modos de uso o *tiers*.

III-B. TIER 0 [FL]

Dentro del modo de uso TIER 0 no se aplica ninguna medida de protección para mantener la privacidad de los datos. No obstante, sí es posible configurar las componentes para que establezcan canales TLS que otorguen autenticación y seguridad en sus comunicaciones. El uso de AES dentro de estos canales TLS permite un cifrado rápido y eficiente, especialmente adecuado para modelos de gran tamaño.

Nótese que, en este modo, el servidor recibe todos los modelos parciales antes de realizar la agregación. Si se compromete el servidor, se puede comprometer toda la seguridad. Es decir, este enfoque no protege frente a un servidor malicioso.

III-C. TIER 1 [FL + DP]

El modo de uso TIER 1 extiende la funcionalidad de TIER 0 utilizando librerías de DP para proteger los modelos antes de intercambiarlos. Dependiendo del modelo a entrenar, se utiliza una de estas dos librerías:

- Opacus [12]. Redes neuronales de Pytorch.
- Diffprivlib [13]. Modelos de scikit-learn, como K-means.

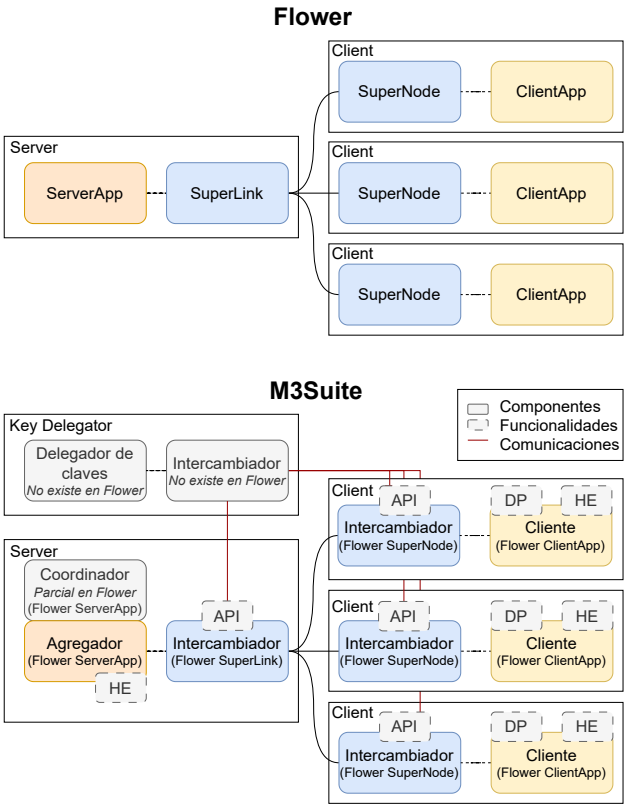


Figura 1. Comparativa entre las arquitecturas de Flower y M3suite

El equilibrio entre privacidad y precisión se ajusta mediante un valor ϵ. Es fundamental encontrar el valor óptimo que garantice un nivel adecuado de privacidad sin comprometer excesivamente la utilidad del modelo.

III-D. TIER 2 [FL + HE]

El modo de uso TIER 2 también extiende la funcionalidad de TIER 0, en este caso mediante la librería de HE OpenFHE [14] para proteger los modelos y realizar la agregación sobre modelos cifrados.

Para que el servidor no disponga de la clave privada y se mantenga la seguridad, se utiliza la técnica *Proxy Re-Encryption*, que requiere de un nuevo componente, el *Key Delegator* (KD), responsable de generar las claves del contexto común y las claves de recifrado.

III-E. Arquitectura

Como base para la propuesta, se ha seleccionado la herramienta Flower [9], que se posiciona como la mejor opción en términos de interoperabilidad y facilidad de uso [15].

En la Figura 1 se muestran los componentes, funcionalidades y flujos de comunicación adicionales de M3Suite, que diferencian ambas soluciones. Las cajas en gris representan los componentes (líneas continuas) y las funcionalidades (líneas discontinuas) nuevas respecto a Flower, mientras que las comunicaciones adicionales se muestran en rojo.

A continuación se explican los componentes, comenzando por el nivel más básico y detallando las funcionalidades adicionales que se incorporan en los niveles superiores.

Server. Se hereda del *Server* de Flower, que se divide en las subcomponentes *ServerApp* y *SuperLink*. Se añade una nueva subcomponente, como extensión de *ServerApp*, que asume el rol de Coordinador, diferenciándose así del Agregador.

- Agregador. Corresponde al componente *Flower ServerApp*, donde se implementan estrategias de agregación.
  Funcionalidades:
  - Agregación. Agrega los modelos locales recibidos de los Clientes para obtener un modelo global agregado. Sigue las especificaciones proporcionadas por el Coordinador.
  - (TIER 2) Agregación cifrada. Agrega los modelos locales cifrados utilizando cifrado homomórfico.
- Coordinador. Está integrado parcialmente en el componente *Flower ServerApp*. Sin embargo, para M3Suite, se extiende añadiendo una interfaz de usuario que permite interactuar con la herramienta de forma sencilla, y se implementa toda la lógica de los *tiers*.
  Funcionalidades:
  - Configuración del proceso. A través de la interfaz, el usuario puede configurar parámetros clave, como el modelo a entrenar, la tipología de datos, el algoritmo de agregación o el tipo de clientes, entre otros.
  - Gestión y coordinación del resto de componentes. Gestiona internamente los componentes de cada ronda según los parámetros especificados por el usuario, incluyendo la asignación de clientes, el uso o no del *Key Delegator*, la ejecución de rondas y la especificación al Agregador del algoritmo a utilizar.
- Intercambiador. En el caso del *Server*, se hace a través del componente *Flower Superlink*.
  Funcionalidades:
  - Comunicación *Server - Client*. Transmite las especificaciones elegidas por el usuario para el entrenamiento, enviándole las instrucciones de las tareas. El Agregador transmite el modelo global y recibe los modelos locales mediante el Intercambiador.
  - (TIER 2) Comunicación *Server - Key Delegator*. Envía peticiones de creación de contexto criptográfico, y recibe el material criptográfico generado.

Client. Se hereda del *Client* de Flower, que se divide en las subcomponentes *ClientApp* y *SuperNode*.

- Cliente. Corresponde al componente *Flower ClientApp*, donde se implementa el entrenamiento local.
  Funcionalidades:
  - Aportación de nuevos datos. A través de una interfaz de usuario, el Cliente recibe y gestiona nuevos datasets que puedan usarse en futuros entrenamientos.
  - Preprocesamiento. Selecciona y homogeneiza los datos necesarios para el entrenamiento, siguiendo las especificaciones proporcionadas por el Coordinador.
  - Entrenamiento. Ejecuta el proceso de entrenamiento local utilizando los datos preprocesados, generando el modelo que posteriormente será agregado.
  - (TIER 1) Protección del modelo. Añade ruido a los datos de entrenamiento o a las actualizaciones de los modelos antes de enviarlos para la agregación.
  - (TIER 2) Generación de claves. Genera un par de claves (pública, privada) propio.

- (TIER 2) Cifrado, recifrado y descifrado. Cifra el modelo local antes de enviarlo. Cuando recibe el modelo global, lo recifra a su propio contexto y después lo descifra.

- Intercambiador. En el caso del *Client*, se hace a través del componente *Flower SuperNode*.
  Funcionalidades:
  - Comunicación *Client - Server*. Recibe las instrucciones para el entrenamiento. También recibe el modelo global y transmite los modelos locales.
  - (TIER 2) Comunicación *Client - Key Delegator*. Recibe todo el material criptográfico necesario. Envía su clave pública para obtener la clave de recifrado asociada.

Key Delegator. No se hereda de Flower, ya que se implementa específicamente para M3Suite.

- Delegador de claves. Se implementa desde cero.
  Funcionalidades:
  - (TIER 2) Generación de contexto criptográfico en base a la selección del usuario.
  - (TIER 2) Generación de claves. Genera un par de claves (pública, privada) del dominio común, así como una clave de recifrado para cada Cliente. Para las claves de recifrado, utiliza la clave pública de cada Cliente y la clave privada común.
- Intercambiador. Se implementa desde cero, requiere modificar las comunicaciones de Flower.
  Funcionalidades:
  - (TIER 2) Comunicación *Key Delegator - Server*. Recibe peticiones de generación de contexto, y envía el material criptográfico asociado.
  - (TIER 2) Comunicación *Key Delegator - Client*. Envía el material criptográfico necesario. Recibe la clave pública de cada Cliente, y envía su clave de recifrado asociada.

IV. Operación

El proceso de operación se estructura en dos fases. La fase de configuración, donde se establecen los parámetros del entrenamiento y se generan las claves criptográficas, y la fase iterativa, donde se ejecutan las rondas de entrenamiento.

IV-A. Fase de configuración

El usuario comienza un entrenamiento a través de la interfaz, especificando el modelo a entrenar y las direcciones de los *Clients* que van a participar. En TIER 2, es necesario aportar datos adicionales, tal como se detalla en la Figura 2. El Coordinador solicita al *Key Delegator* que genere un contexto criptográfico, cc, según las especificaciones del usuario. Al recibir la solicitud, el *Key Delegator* genera el cc y un par de claves (pk, sk) del dominio común. El *Key Delegator* comparte el cc con el *Server* y los Clientes. Asimismo, envía la clave pública común únicamente a los Clientes. Después, cada Cliente genera su propio par de claves ($client\text{-}pk$, $client\text{-}sk$) y envía la clave pública al *Key Delegator*. Finalmente, el *Key Delegator* utiliza la clave pública de cada Cliente junto con la clave privada del dominio común para generar una clave de cifrado específica para cada Cliente, $client\text{-}re\text{-}key$, la cual se envía a su respectivo destinatario.

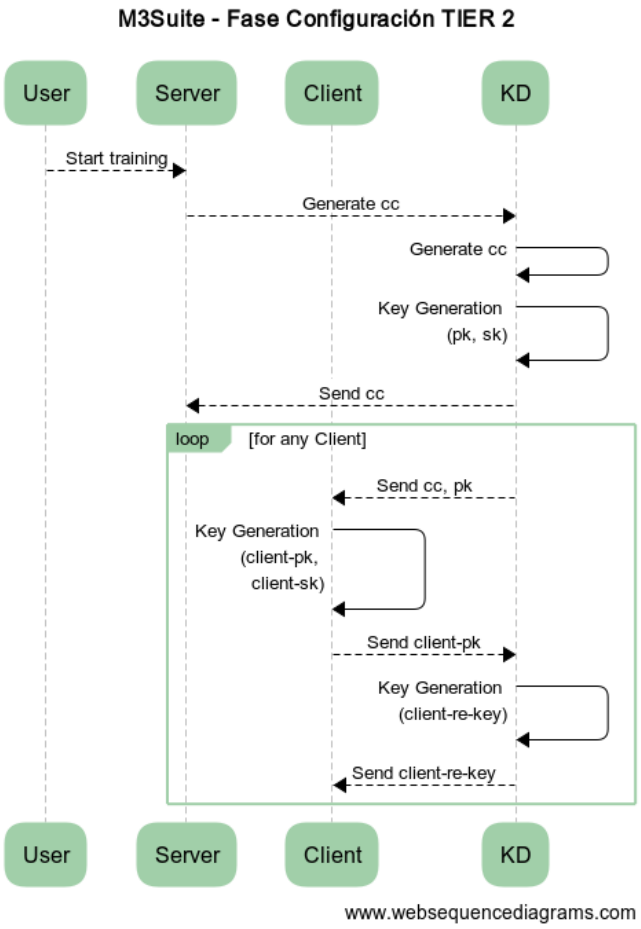


Figura 2. Fase configuración para el TIER 2

IV-B. Fase iterativa

Este proceso se repite iterativamente hasta llegar al número máximo de iteraciones especificadas por el usuario en la fase de configuración. Para algunos niveles de privacidad, es necesario realizar una serie de pasos adicionales, tal como se detalla en la Figura 3 y se explica a continuación.

1. El *Server* envía el modelo global, ω, a cada cliente. En TIER 2, se utiliza la clave pública común, pk, para enviar el modelo global cifrado, $enc(\omega)$.
2. El Cliente entrena el modelo local utilizando el modelo global recibido y sus datos. En TIER 2, antes de entrenar el modelo, el Cliente recifra el modelo global a su propio dominio utilizando la clave de recifrado, $client\text{-}re\text{-}key$, y después lo descifra utilizando su propia clave privada, $client\text{-}sk$.
3. El *Client* envía el modelo local al *Server*. En TIER 1, antes de enviar el modelo, el Cliente le añade ruido utilizando DP, ω'. Sin embargo, en TIER 2, el Cliente utiliza la clave pública común, pk, para cifrar el modelo local antes de enviarlo, $enc(client - \omega)$.
4. El Agregador agrega el resultado. En TIER 2, la agregación se realiza sobre los modelos cifrados utilizando HE.

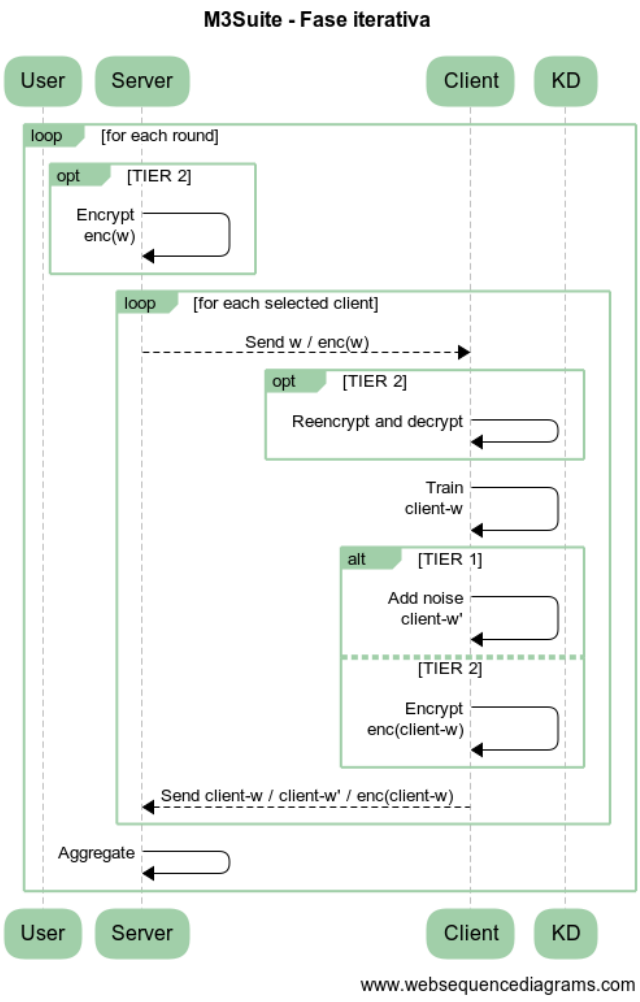


Figura 3. Fase iterativa desglosada por niveles de privacidad

V. Casos de uso y aplicaciones prácticas

Dependiendo de los requisitos del caso de uso, el usuario elegirá un nivel de privacidad u otro. En rasgos generales, los casos de uso de cada uno de los niveles de privacidad son los siguientes:

- TIER 0. Se confía plenamente en el *Server* y no se consideran problemas de privacidad relacionados con él. Sin embargo, al habilitar las comunicaciones cifradas mediante TLS, sí que se mantiene la privacidad respecto al resto de clientes o frente a terceros que puedan interceptar las comunicaciones. Este nivel busca aprovechar FL para distribuir el cómputo, lo que resulta eficiente, aunque puede ser vulnerable a fugas de privacidad por parte del *Server*.
- TIER 1. No se confía en el *Server* y se busca mantener cierto nivel de seguridad frente a ataques de inferencia/reconstrucción. Sin embargo, la prioridad principal es la eficiencia y la flexibilidad para realizar diferentes agregaciones.
- TIER 2. No se confía en el *Server*. La privacidad de los datos es la prioridad principal, incluso si esto implica sacrificar algo de eficiencia computacional. Además, se quiere mantener la precisión del modelo.

A continuación, se presentan dos ejemplos de aplicación práctica para TIER 1 y TIER 2, respectivamente

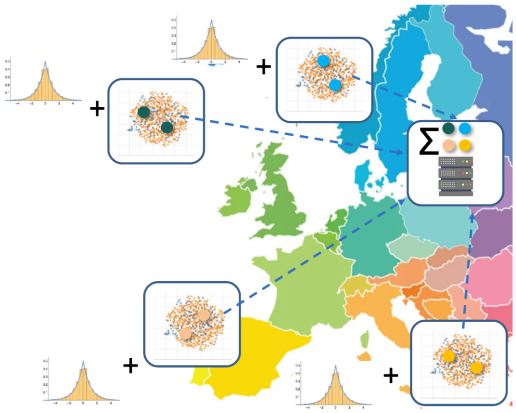


Figura 4. Proyecto COMPASS-NMD

V-A. *TIER 1: Proyecto CoMPaSS-NMD*

El proyecto CoMPaSS-NMD (Computational Models for new Patients Stratification Strategies of Neuromuscular Disorders) tiene como objetivo desarrollar una aplicación de FL para el análisis de enfermedades neuromusculares hereditarias.

Los hospitales que poseen datos, que tomarán el rol de *Client*, están ubicados en Italia, Alemania, Francia y Reino Unido, mientras que el centro de coordinación y procesamiento, que tomará el rol de *Server,* está ubicado en Polonia.

Cada uno de estos países posee datos médicos sensibles de sus pacientes, incluyendo cuatro tipos de información: datos genéticos, resultados de biopsia muscular, imágenes de resonancia magnética y datos clínicos. Estos datos son críticos para la estratificación de pacientes, pero son datos sensibles y están regulados por normativas de privacidad. Por lo tanto, no es posible centralizar toda la información en un solo lugar.

Para implementar la solución, se utiliza M3Suite, y el usuario selecciona la siguiente configuración:

- mode = 'tier1'
- model = 'kmeans'
- strategy = 'kmeans_avg'
- clients = $[address_1, address_2, address_3, address_4]$

donde "clients" es una lista con la dirección de cada cliente, necesaria para configurar el entrenamiento distribuido. En este caso, se selecciona el nivel de privacidad que aplica DP, puesto que realizar la agregación 'kmeans_avg' utilizando HE no es lo suficientemente eficiente para los requisitos del caso de uso, y la aplicación de DP cumple con la normativa necesaria.

En la Figura 4 se muestra cómo cada uno de los Clientes añade ruido antes de compartirlo con el Agregador, siguiendo la operativa de M3Suite para TIER 1.

V-B. *TIER 2: Proyecto REFRESH*

El proyecto REFRESH (Smart dismantling, sorting and REcycling of glass Fibre REinforced composite from wind power Sector through Holistic approach) tiene como objetivo optimizar la toma de decisiones en un caso de uso de reciclaje de turbinas eólicas. Para ello, se está estudiando la posibilidad de utilizar una "Herramienta de toma de decisiones" basada en un modelo de IA. El objetivo es que, dados los datos de una

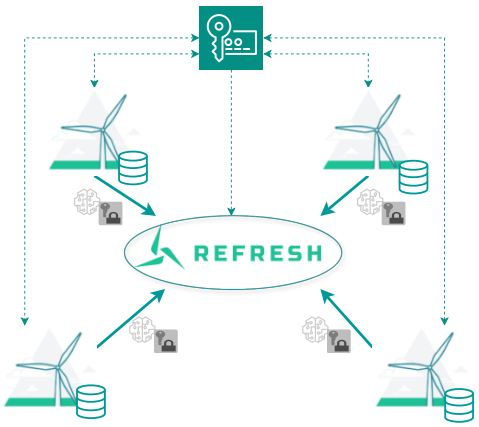


Figura 5. Proyecto REFRESH

turbina, el modelo sea capaz de predecir el mejor camino de reciclaje mediante un clasificador que utiliza rutas de reciclaje predefinidas como etiquetas.

Sin embargo, para entrenar este modelo, se utilizan datos históricos sobre el reciclaje de turbinas eólicas proporcionados por diferentes entidades. Dado que los datos de estas entidades son privados, se propone que el usuario responsable de la herramienta utilice M3Suite para entrenar el modelo, seleccionando la siguiente configuración:

- mode = 'tier2'
- model = 'multiclass'
- strategy = 'fedavg'
- clients = [$address_1$, ..., $address_n$]

En este caso, se selecciona el nivel de privacidad que aplica HE, puesto que la privacidad es la máxima prioridad y la agregación de 'fedavg' es eficiente para el caso de uso.

En la Figura 5 se muestra cómo cada uno de los Clientes cifra el modelo local antes de compartirlo con el Agregador, siguiendo la operativa de M3Suite para TIER 2.

VI. Conclusiones

Este trabajo presenta M3Suite, una solución integral para FL que aborda los principales desafíos de modularidad, privacidad y usabilidad. A diferencia de otras herramientas del estado del arte, que suelen centrarse en un subconjunto limitado de librerías o requieren un alto nivel de especialización técnica, M3Suite está diseñada para ser adaptable a distintas librerías de IA. Además, ofrece diferentes niveles de protección de la privacidad, desde la ausencia de mecanismos hasta la aplicación de técnicas como DP o HE. Esta flexibilidad permite adaptar el sistema a las necesidades específicas de cada escenario, garantizando un equilibrio entre seguridad y rendimiento. Todo ello facilitando además su uso por parte de usuarios no expertos en IA y criptografía, ofreciendo una herramienta de *FL-as-a-service*. Para demostrar su aplicabilidad, se presentan dos ejemplos prácticos de uso real.

A modo de trabajo futuro, se proponen las siguientes líneas:

- Automatizar la aplicación del ruido adecuado en TIER 1 para nuevos modelos no incluidos previamente en la herramienta. El objetivo es que, al añadir un modelo nuevo en TIER 0, el sistema implemente automáticamente un mecanismo para extenderlo a TIER 1, sin necesidad de intervención manual.
- Evaluar las repercusiones en términos de seguridad de la incorporación de un *Key Delegator* en TIER2, considerando que maneja material criptográfico sensible y añade un nuevo punto de confianza y vectores de amenaza como el compromiso del componente o uso indebido de claves.
- Realizar una discusión sistemática para diferentes modelos de amenaza, evidenciando el análisis coste/beneficio de cada modo en diferentes escenarios.
- Implementar agregaciones robustas que sean efectivas frente a ataques relacionados con el modelo, como el envenenamiento de datos. Esto afecta a las agregaciones y a sus equivalentes en HE.
- Incorporar mecanismos para abordar otros pilares como la equidad, la explicabilidad, la responsabilidad y la federación, con el objetivo de avanzar hacia un Aprendizaje Federado Confiable.

Agradecimientos

Este trabajo ha contado con financiación de la Unión Europea a través de los proyectos REFRESH (acuerdo de subvención nº 101096858), y CoMPaSS-NMD (acuerdo de subvención nº 101080874).

Referencias

[1] Carlo Mazzocca, Nicolò Romandini, Matteo Mendula, et al. : "TruFLaaS: Trustworthy Federated Learning as a Service", en *IEEE Internet of Things Journal*, vol. 10, n. 24, pp. 21266-21281, 2023.

[2] Viraaji Mothukuri, Reza M Parizi, Seyedamin Pouriyeh, et al.: "A survey on security and privacy of federated learning", en *Future Generation Computer Systems*, pp. 619–640, 2021.

[3] Brendan McMahan, Eider Moore, Daniel Ramage, et al.: "Communication-efficient learning of deep networks from decentralized data", en *Artificial intelligence and statistics*, pp. 1273-1282, 2017.

[4] Ran Canetti, Uri Feige, Oded Goldreich, y Moni Naor: "Adaptively secure multi-party computation", en *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pp. 639–648, 1996.

[5] Zvika Brakerski: "1 homomorphic encryption : Good , bad or ugly ?", 2018.

[6] Cynthia Dwork: "Differential Privacy", en *Encyclopedia of Cryptography and Security*, pp. 338–340, 2011.

[7] Google: "TensorFlow Federated", https://github.com/google-parfait/tensorflow-federated.

[8] Alexander Ziller, Andrew Trask, Antonio Lopardo, Benjamin Szymkow, Bobby Wagner, et al.: "Pysyft: A library for easy federated learning", en *Federated Learning Systems: Towards Next-Generation AI*, pp. 111-139, 2021.

[9] Daniel J Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Javier Fernandez-Marques, et al.: " Flower: A friendly federated learning research framework", en *arXiv*, 2022.

[10] Yang Liu, Tao Fan, Tianjian Chen, Qian Xu, y Qiang Yang: "Fate: An industrial grade platform for collaborative learning with data protection", en *Journal of Machine Learning Research*, pp. 1-6, 2021.

[11] Holger R Roth, Yan Cheng, Yuhong Wen, Isaac Yang, Ziyue Xu, et al.: "Nvidia flare: Federated learning from simulation to real-world", en *IEEE Data Eng. Bull.*, vol. 46, n. 1, 2023.

[12] Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, et al.: "Opacus: User-Friendly Differential Privacy Library in PyTorch", en *arXiv preprint arXiv:2109.12298*, 2021.

[13] Naoise Holohan, Stefano Braghin, Pól Mac Aonghusa, y Killian Levacher: "Diffprivlib: the ibm differential privacy library", en *ArXiv e-prints, 1907.02444*, 2019.

[14] Ahmad Al Badawi, Andreea Alexandru, Jack Bates, Flavio Bergamaschi, et al.: "Openfhe: Open-source fully homomorphic encryption library", en *Cryptology ePrint Archive Papers* , 2022.

[15] Pascal Riedel, Lukas Schick, Reinhold von Schwerin, et al.: "Comparative analysis of open-source federated learning frameworks - a literature-based survey and review", en *International Journal of Machine Learning and Cybernetics*, vol. 15, n. 11, pp. 5257-5278, 2024.

Quantum Sieving to solve the LWE Problem

Diego Rojas Rodríguez
I. Tec. Físicas y de la Información
ITEFI-CSIC, Madrid
diego.rojas@csic.es

Miguel Ángel González de la Torre
I. Tec. Físicas y de la Información
ITEFI-CSIC, Madrid
ma.gonzalez@csic.es

Luis Hernández Encinas
I. Tec. Físicas y de la Información
ITEFI-CSIC, Madrid
luis.h.encinas@csic.es

***Abstract*—This publication explores a quantum algorithm designed to solve the Shortest Vector Problem (SVP) as a reduction of the Learning With Errors (LWE) problem via quantum embedding. Combining LWE instances in quantum superposition with techniques such as conditional shifts and quadrant measurements, the algorithm generates structured states amenable to quantum sieving. The approach is validated via quantum simulations in low-dimensional lattices, demonstrating the feasibility of preparing the required superpositions and recovering shortest vector candidates. While current quantum hardware noise prevents practical implementation, the results open future lines for research on quantum sieving and error correction techniques in cryptographic lattices.**

***Index Terms*—Quantum Computing, Lattice Cryptography, Post-Quantum Cryptography, Sieving**

I. Introduction

In 1997, Peter Shor presented two variants of a quantum algorithm [1] that fundamentally threaten the security of modern public key cryptosystems. The security of such systems relies on the difficulty of certain computational problems that should render unfeasible the recovery of the private key from the public key, despite their mathematical relation. Shor's algorithm provides efficient solutions for both the Integer Factorization Problem (IFP) and the Discrete Logarithm Problem (DLP) used by most public key cryptosystems. The imminent obsolescence of established cryptographic standards posed a critical threat to the security and confidentiality of communications on the Internet. This motivated international efforts to identify secure alternatives, culminating in the post-quantum cryptography standardization process led by the National Institute of Standards and Technology (NIST). Most of the proposals base their security on hard lattice problems, which have become the foundation for modern asymmetric cryptography. While no known classical or quantum algorithms can efficiently solve these problems in high dimensional lattices, concerns remain within the cryptographic community about the potential for future quantum breakthroughs, which we will explore. Success on the implementation of this algorithm would offer an alternative to most of the other quantum speedups that rely on Grover's algorithm.

The rest of this work is organized as follows. In Section II the mathematical problems related to lattice-based cryptography are presented. Section III recalls some aspects about quantum computing. In Section IV the main algorithm is introduced whereas our results are presented in Section V. Finally, the conclusions are commented in Section VI.

II. Learning With Errors Problem and Cryptosystem

In the final round of the post-quantum cryptography standardization process, the Learning With Errors (LWE) problem emerged as the foundational hard problem underpinning most lattice-based cryptosystems. LWE provides a strong security due to its presumed hardness against both classical and quantum attacks, and its connection to well-studied lattice problems, such as the Shortest Vector Problem (SVP) and Bounded Distance Decoding (BDD). The following formal definition describes the LWE distribution.

Definition 1 (Learning With Errors distribution [2]): Let q be a positive integer modulus and let $s \in \mathbb{Z}_q^n$ be a secret column vector. Let ϕ be a probability distribution over $\mathbb{Z}_q$ that approximates a discrete Gaussian with mean $\mu = 0$ and small variance $\sigma \ll q$. The Learning With Errors (LWE) distribution generates samples of the form $(a_i, b_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, where each $a_i \in \mathbb{Z}_q^n$ is chosen uniformly at random and $b_i = \langle a_i, s\rangle + e_i \pmod q$, with e_i following the probability distribution of ϕ. Here, $\langle \cdot, \cdot \rangle$ denotes the standard inner product over $\mathbb{Z}_q^n$. The LWE distribution with parameters ϕ and s is denoted by $\text{LWE}_{\phi,s}$.

The vectors set $B = \{a_i\}$ define a lattice $\mathcal{L}(B)$, and $b = \{b_i\}$ represents a vector of $\mathbb{Z}_q^n$ that is close to the lattice. Generating random samples with uniformly chosen coefficients in high dimensions produces LWE instances that are, with overwhelming probability, hard to solve. The error terms e_i typically do not lie in the lattice, which significantly increases the problem's hardness by perturbing the lattice structure in a way that masks its linear properties. Note that if we define the matrix $A := (a_1 a_2 \cdots a_n)^T \in \mathbb{Z}_q^{n\times n}$, then

$$A \cdot s = (\langle a_1, s\rangle \cdots \langle a_n, s\rangle) = (b_1 - e_1 \cdots b_n - e_n) := b - e,$$

holds, where $e = \{e_i\}$. Thus, the pair $(A, b := A \cdot s + e) \in \mathbb{Z}_q^{n\times n} \times \mathbb{Z}_q^n$ defines the above $\text{LWE}_{\phi,s}$ sample. As a result, recovering the secret vector s from the samples (A, b) is computationally challenging because the errors introduce uncertainty that obscures the underlying linear relationships.

Definition 2 (LWE Problem): Recover the values of the secret vector s given access to the matrix $A := (a_1 a_2 \cdots a_m)^T \in \mathbb{Z}_q^{m\times n}$ with $m \geq n$, samples of the LWE distribution $\text{LWE}_{\phi,s}$ where $(A, b = A \cdot s + e \pmod q)$.

The security of LWE-based cryptosystems relies on keeping the secret vector s hidden (the secret key). When the error terms e_i are small, the deviation they introduce is limited, making it easier for an attacker to recover s and, consequently, the original message. Increasing the magnitude of the error strengthens the cryptosystem's security, but at the cost of reducing the efficiency and reliability of the decryption process. Brute-force attacks, as well as more sophisticated lattice reduction techniques, often aim to reduce the matrix A in order to find lattice points that are closest to the challenge vector b. This gives rise to the following related problem.

Definition 3 (Bounded Distance Decoding problem (BDD)): Let $\mathcal{L} \subset \mathbb{R}^n$ be a full-rank lattice (where $n = m$), and let $t \in \mathbb{R}^n$ be a target point. The task is to find a lattice point $x \in \mathcal{L}$ such that $\|t - x\| \leq \lambda_1(\mathcal{L}) \pmod q$, where $\lambda_1(\mathcal{L})$ denotes the length of the shortest nonzero vector in $\mathcal{L}$.

The BDD problem shifts the focus to finding lattice points that are closest to a given target vector b. Although solving BDD is not fundamentally easier than solving LWE —since there is no known reduction that decreases the underlying hardness— existing algorithms tend to perform better on BDD instances. When the lattice basis is well-conditioned or nearly orthonormal, the relation between the challenge vector and the basis vectors approaches linearity, making the problem significantly easier to solve in practice.

Definition 4 (Shortest Vector Problem (SVP)): Let $\mathcal{L} \in \mathbb{R}^n$ be a lattice. The *Shortest Vector Problem* (SVP) asks for a nonzero lattice vector $x \in \mathcal{L}$ such that $\|x\| = \lambda_1(\mathcal{L})$. If the lattice contains exactly two nonzero vectors of shortest length (namely x and $-x$), the *unique Shortest Vector Problem* (uSVP) consists in finding one of these two vectors.

Finding the closest lattice points often requires knowledge of the shortest vector in the lattice, as this eases the construction of a reduced basis that simplifies decoding tasks. Since the SVP is known to be computationally hard, the related problems such as BDD and LWE inherit this difficulty. In practical applications, however, these problems do not typically arise directly, because one can often choose the lattice basis to be well-conditioned or already reduced, thereby avoiding the inherent hardness seen in cryptographic contexts.

III. Quantum Computing

Quantum computing plays a significant role in the study of problem hardness and computational complexity, particularly in the context of cryptographic applications. The main difference with classical computing is the use of quantum phenomena to speed up certain operations. Both computing methods are often combined and most algorithms just consider a quantum speedup using a Quantum Processing Unit (QPU) for certain steps.

One of the most important problems in quantum computational complexity is the Hidden Subgroup Problem (HSP). Efficiently solving instances of this problem is crucial to many quantum algorithms, including Shor's algorithm, which provides a solution for the Abelian case of the HSP.

Definition 5 (Hidden Subgroup Problem [3]): Let G be a finite group, $H \subset G$ a subgroup, and $f\colon G \to Y$ an oracle function. The function f is said to *hide* the subgroup H if

$$\forall c_1, c_2 \in G : f(c_1) = f(c_2) \iff c_1 H = c_2 H.$$

The problem consists in determining the hidden subgroup H given access to the oracle f.

Here, we will consider oracles f that return 0 for any $h \in H$ and a nonzero constant for all other elements of G.

For Abelian groups, instances of the HSP can be efficiently solved using quantum algorithms such as Shor's algorithm. However, for non-Abelian groups, no efficient quantum algorithm is currently known. In fact, the hardness of certain non-Abelian instances of HSP, such as those over dihedral groups, underpin the presumed security of various post-quantum cryptographic schemes. In the context of lattice problems, the dihedral group frequently arises in formulations of the HSP.

Definition 6 (Dihedral Group [3]): The dihedral group D_N is the group of symmetries of a regular polygon with N sides (a regular N-gon). The group has order $2N$ and can be generated by two elements, r and s, satisfying the relations $r^2 = s^N = 1$ and $rsr = s^{-1}$. Here, s represents a rotation by $2\pi/N$, and r represents a reflection symmetry.

The geometric interpretation of the dihedral group, illustrated in Figure 1, shows that the generator s corresponds to a rotation of the polygon by an angle $2\pi/N$, while r represents a reflection across an axis of symmetry. Every element of D_N can thus be expressed as a composition of these operations:

$$D_N = \{ s^a r^b : a \in \mathbb{Z}_N,\ b \in \mathbb{Z}_2 \}.$$

Here, s^a denotes a rotation by $2\pi a/N$, and r^b indicates whether a reflection is applied ($b = 1$) or not ($b = 0$). Reductions of the Hidden Subgroup Problem (HSP) for dihedral groups give rise to the Dihedral Coset Problem (DCP), which is believed to be intractable even for quantum computers using known techniques.

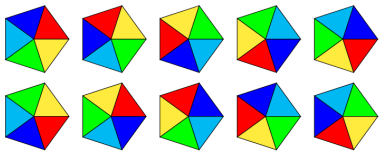


Fig. 1. Elements of the dihedral group D_5 [4]

Definition 7 (Dihedral Coset Problem (DCP)): Given a collection of quantum states $|\psi\rangle$ of the form

$$|\psi\rangle = \frac{1}{\sqrt{2}} \big(|x\rangle\,|0\rangle + |x + d\rangle\,|1\rangle \big),$$

where x is chosen uniformly at random and d is a hidden shift, the task is to determine the value of d.

The DCP is closely related to the HSP for the dihedral group, and its presumed hardness underlies the security of several lattice-based and post-quantum cryptographic schemes. In particular, solving the DCP efficiently would enable the recovery of hidden shifts in problems reducible to dihedral HSP instances.

One of the most notable quantum algorithms for approaching the DCP is Kuperberg's sieve algorithm [5], which solves the problem in subexponential time. Kuperberg's method combines multiple coset states through entanglement and interference to iteratively reduce the size of the hidden shift d. Despite this improvement over brute force, the algorithm still requires a significant amount of qubits and time, making it impractical for large instances.

Reduction 1 (HSP$\leq_p$DCP [6]): Quantum states of the form $|\psi\rangle$ can be transformed into coset states of the dihedral group D_N, where arithmetic is performed modulo N. By applying coset sampling, one can recover the hidden shift d, which corresponds to the difference between the hidden subgroup H and the coset $H + d$. The coset sampling procedure is outlined in Algorithm 1.

The reduction from the HSP to the DCP illustrates how, under a polinomial-time bound, instances of DCP can be framed as hidden subgroup instances over the dihedral group D_N. In this setting, the hidden shift d defines a coset $H + d$ of the hidden subgroup H, and the task is to recover d given quantum access to coset states.

Algorithm 1 Coset Sampling

1) Prepare a uniform superposition over all elements of G:
$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle .$$
2) Query the oracle f in an auxiliary register:
$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle |f(g)\rangle .$$
3) Measure the second register, obtaining some outcome $f(k) \neq 0$. The state collapses to:
$$\frac{1}{\sqrt{|H|}} \sum_{h \in H} |kh\rangle .$$

Since no efficient quantum algorithm is known for the dihedral HSP, the DCP inherits this presumed hardness, reinforcing its role as a foundation for cryptographic security in certain lattice-based and hidden shift cryptosystems.

Given that lattices are periodic (or isotropic) structures that repeat uniformly in space, it is natural to generalize the Dihedral Coset Problem (DCP) to a broader setting where one seeks to recover the relative shift between any two cosets. This leads to the formulation of the Two-Point Problem (TPP).

Definition 8 (Two-Point Problem (TPP)): Given a collection of quantum states $|\psi\rangle$ of the form

$$|\psi\rangle = \frac{1}{\sqrt{2}} \big(|x\rangle |0\rangle + |y\rangle |1\rangle \big), \quad y - x = d,$$

where arithmetic is performed modulo N, the task is to determine the distance d between the states $|x\rangle$ and $|y\rangle$.

Reduction 2 (DCP $\leq_p$ TPP [4]): The TPP is a generalization of the DCP, where instead of distinguishing a subgroup H from a coset $H+d$, one must determine the shift d between two different cosets.

Reduction 3 (uSVP $\leq_p$ TPP [4]): Instances of the uSVP reduce to the TPP by defining a function g that partitions $\mathbb{R}^n$ into cubes of side length r:

$$g(x) = (\lfloor x_1/r - s_1 \rfloor, \lfloor x_2/r - s_2 \rfloor, \ldots, \lfloor x_n/r - s_n \rfloor),$$

where s represents an offset that shifts the cubes away from the center $\mathcal{O}$ of the lattice $\mathcal{L}$. If r is within a factor of $O(\sqrt{n})$ of $\lambda_1(\mathcal{L})$, and if $|\lambda_2(\mathcal{L}) - \lambda_1(\mathcal{L})| \in O(\sqrt{n})$, then only vectors aligned with the shortest vector will fit inside each cube.

To ensure that only two points fall inside each cube, we can use a quantum superposition over lattice points generated by scaled and shifted basis vectors. Let $m, p \in \mathbb{Z}_N$ be scale factors, and select $b_0 \in B$ as a basis vector of $\mathcal{L}(B)$. We define lattice points $x \in \mathcal{L}$ as

$$x = x_0(pb_0) + t(mb_0) + \sum_{i=1}^{k} x_i b_i,$$

where $t \in \mathbb{Z}_2$ determines whether the shift is applied. The points with $t = 1$ are separated by a distance d from those with $t = 0$, and this distance d corresponds to $\lambda_1(\mathcal{L})$, the solution to the uSVP instance.

At this point we have stated that solving an instance of the TPP, which is as hard as solving the HSP, gives a solution to the uSVP. Thus the only missing piece for an algorithm that solves the uSVP is a somewhat efficient solution to the TPP, which in our case will be the aforementioned Kuperberg Sieve [5].

Definition 9 (Kuperberg Sieve [5]): The Kuperberg sieve is a quantum algorithm designed to recover a hidden value y given $2^{O(\sqrt{n})}$ samples of quantum states $|\psi_k\rangle$ of the form

$$|\psi_k\rangle = \frac{1}{\sqrt{2}} \left(|0\rangle + \omega_N^{yk} |1\rangle \right),$$

where $\omega_N = e^{2\pi i/N}$ is a primitive N-th root of unity. The algorithm combines these states in a structured way to extract information about y.

The sieving process in Kuperberg's algorithm systematically combines coset states to amplify information about the hidden value y. The goal is to reduce the problem to simpler instances from which y can be efficiently recovered. The procedure is shown as Algorithm 2.

Algorithm 2 Kuperberg Sieve [5]

1) Prepare a sufficient number of coset states of the form $|\psi_k\rangle$, where each k is chosen independently and uniformly at random.
2) For each $j \in \{0, 1, \ldots, m-1\}$, where $m = \lceil \sqrt{n} \rceil$, group the states into pairs $(|\psi_p\rangle, |\psi_q\rangle)$ such that the j least significant bits of p and q are equal.
3) Combine each pair to form new states $|\psi_{p \pm q}\rangle$ through appropriate quantum operations (e.g., entanglement and measurement). Measure the second qubit of each combined state and discard outcomes of the form $|\psi_{p+q}, 0\rangle$.
4) Repeat the pairing and combining steps until states with progressively fewer significant bits of y remain. The final states will be of the form $|\psi_0\rangle, \ldots, |\psi_{2^{n-1}}\rangle$.
5) Measure the remaining states in the $\{|+\rangle, |-\rangle\}$ basis to determine the least significant bit of y.

IV. Algorithm

The results of this work include the definition of the required steps to adapt an instance of the uSVP to a quantum computer where one could run Kuperberg's sieve. This reduces, theoretically, the hardness of solving uSVP to the cost of running the quantum sieving algorithm, which is on the order $\mathcal{O}(2^{\sqrt{n}})$ for a dihedral group of order $N = 2^n$.

A. Embedding LWE into Quantum Superpositions

Consider a full-rank lattice $\mathcal{L}$ defined by a basis $B \in \mathbb{Z}_q^n$ and with shortest vector length $\lambda_1(\mathcal{L})$. To construct an homogeneous quantum superposition over all points in $\mathcal{L}$, we map the values of $\mathbb{Z}_q^n$ to their corresponding combinations of basis vectors. Assuming we require approximately $N \approx n\lceil \log_2(q) \rceil$

qubits to represent these combinations, we initialize the system in the uniform superposition

$$|+\rangle^{\otimes N} = \frac{1}{\sqrt{2^N}} \sum_{k=0}^{2^N-1} |k\rangle .$$

We define a unitary operation f that maps coefficient states $|x\rangle$ to lattice points modulo q:

$$f(|x\rangle) = U_1 |x\rangle = \frac{1}{\sqrt{|\mathcal{L}(B)|}} \sum_{i=0}^{N-1} |x_i \cdot b_i \bmod q\rangle ,$$

where $|\mathcal{L}(B)|$ is the number of lattice points generated by B. Applying f to the $|+\rangle^{\otimes N}$ state yields

$$|\mathcal{L}\rangle = \frac{1}{\sqrt{|\mathcal{L}(B)|}} \sum_{i=0}^{2^N-1} \sum_{j=0}^{N-1} |x_i \cdot b_j \bmod q\rangle ,$$

representing a uniform superposition over all lattice points modulo q.

Next, we partition $\mathbb{R}^n$ into cubes of side length r with offset $|s\rangle$ using a unitary function g:

$$g(|x\rangle) = U_2 |x\rangle = \sum_{i=0}^{N-1} \left\lfloor \frac{x_i}{r} - s_i \right\rfloor .$$

Choosing r appropriately ensures that at most two points fit at any given cube.

To ensure this holds, we introduce scale factors p and m. The value p is chosen large enough to separate lattice points by at least one cube, while m, $1 \leq m < p$, defines the shift applied to the lattice. The resulting quantum superposition combines scaled and shifted versions of the lattice:

$$\begin{aligned} |\Psi\rangle = & \frac{1}{\sqrt{|\mathcal{L}(pB)|}} \left(\sum_{k=0}^{N-1} p \cdot |v_k\rangle \right) |0\rangle \\ & + \frac{1}{\sqrt{|\mathcal{L}(pB)|}} \left(\sum_{k=0}^{N-1} p \cdot |v_k\rangle + m \cdot |b\rangle \right) |1\rangle , \end{aligned}$$

where $v_k \in \mathcal{L}$. For suitable p and m, the distance between points from the two lattices corresponds to a factor of $\lambda_1(\mathcal{L})$.

B. Definition of the function f

Classically one would run all possible linear combinations of vectors from the basis that generate all different points. Depending on the basis chosen this may be hard with classical computers. Using quantum computing it suffices to multiply the state of homogeneous superposition $|+\rangle$ by the basis of the lattice to have all the points in superposition with the same probability. For our purposes, we decided to use an oracle similar to the one used in Shor's algorithm [1] for exponentiation. In the classical setting, the scale factor p is multiplied to generate the basis pM, which is then applied to the state $|+\rangle^{\otimes}$. The required number of qubits is $n \cdot \log_2(q)$, where n denotes the lattice dimension and q the modulo of the operation. Each number is represented with $\log_2(q)$ qubits and there is one number for each dimension.We used two registers, one that will start with the homogeneous superposition of $n \cdot q$ states and another one to keep the results of the vector multiplication.

C. Shifting the lattice and Quadrant measurement

This step requires to add all points from a shifted lattice to the superposition. Again, the process is easily done with a quantum operation. First one should choose a vector from basis pM and multiply it by a shift factor m. This vector will be added conditionally controlled by a qubit on the state $|+\rangle$. This both adds and does not add the vector keeping on superposition both the points in pM and $pM + s$, where s is the vector chosen for the shift.

The superposition now contains a set of points which are separated by a factor of the distance of the shortest vector in the lattice. However, one still needs to choose the appropriate points as not all are separated by this exact distance with each other. One needs to choose a pair of points that are close and belong to one lattice each. This becomes a problem as the dimension increases, as one may need to perform many measurements to find these two points needed. A first approach is to force the collapse of the first bits in the superposition on each dimension to fix the quadrant on which the points will be measured. If the quadrants are small enough one can guarantee that two points in the same quadrant will be separated by the shortest vector. The question now is whether it is possible to find a balance between the size of the quadrants and the amount of quadrants that have points inside. It would also be relevant to find quadrants shifted from the center as one could have cubes of appropriate size that do not contain two points. We believe that shifting all points in the superposition from the center may allow to fix this issue.

D. Pairing points and Quantum sieving

Assuming that a set of points within the same quadrant has been obtained, it remains necessary to determine the distance between them. This is trivial if both points can be measured, but in lattices with high dimension this may not happen in many shots. We proposed to use the Quantum Fourier Transform (QFT) to force this to happen, as encoding the distance in the frequency domain enables retrieval of its exact value, which should remain consistent across all quadrants.

The last step consists in recovering the distance $\lambda_1(\mathcal{L})$ using a sieving process. The original concept underlying sieving algorithms involves performing reductions of vectors within a large sampled list. Classically this consists of adding or subtracting these vectors looking for the shortest ones. In the context of a quantum sieve, quantum states in superposition are employed, with the objective of recovering the phase difference between them. After applying the QFT in the previous step we are left with exactly this superposition and will apply Kuperberg sieve 2 to get the distance between states which now is a difference in the phase of the states.

V. Results on simulation and quantum hardware

To prove whether the hypothesis holds we decided to make an implementation in Qiskit following these steps to retrieve the shortest vector of small lattices. The chosen parameters for the simulation were the following:

1) $q = 16$ modulo for the operations.
2) $n = 2$ dimension of the lattice.
3) $p = 4$ scale factor.
4) $m = 1$ shift factor.

5) $r = 1$ bits for the quadrants.
6) $M = \begin{pmatrix} 2 & 0 \\ 1 & 2 \end{pmatrix}$.

The circuit for this example requires 17 qubits in total: $n \cdot \log_2(q) = 8$ qubits for the register x controlling the matrix multiplication, another $n \cdot \log_2(q) = 8$ qubits to store the result, and 1 ancillary qubit to control the shift operation. After running a simulation with 32,000 shots of the circuit, we expect to obtain points belonging to the lattices $4 \cdot M$ and $4 \cdot M + (02)^T$, separated by the exact distance $\lambda_1(\mathcal{L}(M))$. To determine this distance, we first measure the most significant bits of the superposition, thereby collapsing the state onto a specific quadrant. Once the quadrant is selected, only two points remain: one from each of the two lattices.

Figure 2 illustrates how the results correspond to a set of points in the lattice, and how these points are indeed separated by the shortest distance. As demonstrated in Figure 3, recovering these points is feasible as long as the parameters n and q remain small. However, both the size of the circuit and the number of points in the superposition grow significantly with these parameters, eventually making such representations impractical.

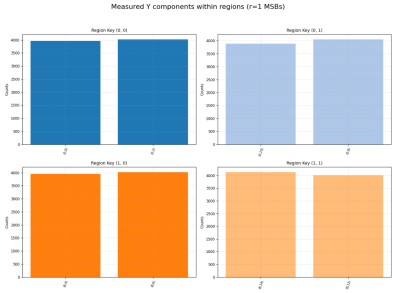


Fig. 2. Counts of the appearances of each point in the lattice grouped by quadrants

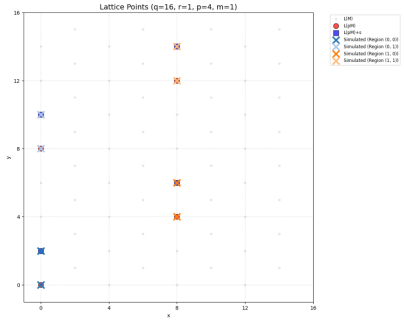


Fig. 3. Results of the simulation of the circuit using the example parameters. The simulator used was AER Simulator. Colours blue and orange represent the four quadrants. Red points belong to $4 \cdot M$ and blue points belong to $4 \cdot M + (02)^T$.

To illustrate the increasing complexity of higher-dimensional lattices, we prepared a three-dimensional example with the following parameters:

1) $q = 16$ modulo for the operations.
2) $n = 3$ dimension of the lattice.
3) $p = 4$ scale factor.
4) $m = 3$ shift factor.
5) $r = 2$ bits for the quadrants.
6) $M = \begin{pmatrix} 0 & 13 & 3 \\ 0 & 5 & 13 \\ 14 & 0 & 2 \end{pmatrix}$.

The circuit for this example required 25 qubits, calculated in the same way as in the two-dimensional case. In this instance, two bits were used for the quadrant selection, since larger cubes would contain more than two points. The choice of m also differs: only for the value $m = 3$ it is possible to identify the shortest distance in this specific lattice. Nevertheless, selecting appropriate parameters does not necessitate multiple executions of the quantum circuit, as the correctness of these values can be verified classically within polynomially bounded time.

As shown in Figure 4, the points in each lattice are again separated by the shortest distance in the original lattice, consistent with the premise of the algorithm. However, in this case, not all sectors contain points, and it is evident that not every quadrant yields two points. In Section IV, we incorporated a shift factor into the algorithm precisely to prevent the quadrants from being centered and to enable the solution of cases such as the one depicted in Figure 4.

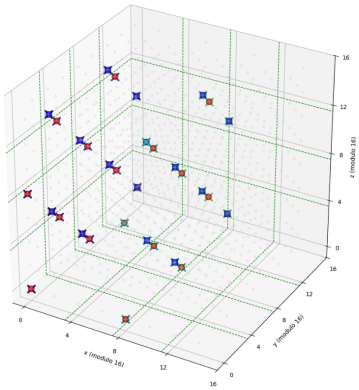


Fig. 4. Simulation of a the algorithm on a 3D lattice. Coloured crosses represent the occurrences in each quadrant during simulation. Red points belong to the lattice generated by M and blue points to the shifted lattice.

The results of this simulation also indicate that the subsequent step —the quantum sieving— would not succeed under these conditions. Another important consideration is the choice of parameters. We included an additional example to illustrate how an inappropriate selection of the scale factor p and the shift factor m can result in an unsolvable instance of the problem (see Figure 5). The purpose of the scale factor is to reduce the number of points in superposition and to align the lattice points along a specific direction. The shift factor introduces a slight displacement of the entire lattice, allowing access to points in all lattices between $\mathcal{L}(M)$ and $\mathcal{L}(pM)$. This mechanism explains why it is possible to recover the shortest distance in the original lattice and not merely in the scaled version, and why values of m closer to 1 or p tend to produce better superpositions (see Figure 6).

When the distance between the shortest vectors in the lattice is sufficiently large, the problem becomes easier to solve. For this reason, adopting an embedding technique such as that proposed by Kannan [7], despite the cost of increasing the lattice dimension, has proven beneficial in classical algorithms. In any case, performing an exhaustive

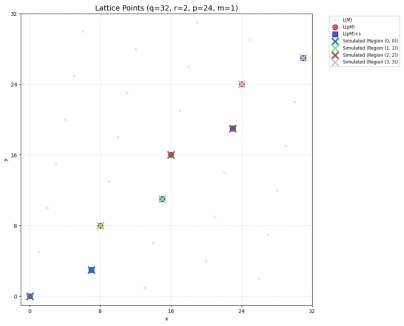


Fig. 5. Results of a simulation of the algorithm on a different lattice with a choice of m that does not lead to the exact $\lambda_1(\mathcal{L})$ but rather just a short vector in the lattice

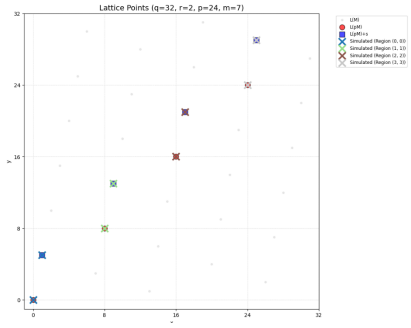


Fig. 6. Results of the simulation of the algorithm on a different lattice with a choice of m that leads to the exact $\lambda_1(\mathcal{L})$. One can see how choosing a different m allows this algorithm to solve the problem.

search over possible values of the scale factor p and the shift factor m should be a feasible strategy prior to executing the quantum simulation.

After proving that the algorithm behaves as expected, we proceeded to run the circuit on a real quantum computer. We transpiled the circuit for the backend IBM Brisbane, of 127 noisy qubits, and got the following results. The results obtained are extremely noisy, which makes the analysis difficult (see Figure 7). We expected to find, despite the noise, 8 clear states which are the states of the points obtained in the simulation. Instead we see 16 peaks with a width of 4 states with more occurrences than the rest. The width of these peaks is a consequence of the noise in the quantum computers. However, there should not be 16 of these peaks. After a more in depth analysis, we concluded this might be a consequence of the resulting transpiled circuit as they do not represent any states belonging to either of the lattices. This might be fixed by implementing an error correction code, but Qiskit does not include any tool to easily implement them. We leave as another line of future work the possibility of implementing quantum error correction to see how it affects to this results and to see if it is possible to obtain results similar to the simulations.

VI. Conclusions

The algorithm presented performed as expected across various lattices with different parameters. Solving the SVP using quantum superposition which requires significantly fewer than the current classical approaches. Nevertheless, we cannot claim that this method is viable as of today. The noise levels in

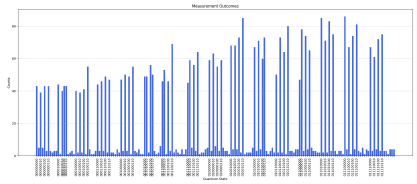


Fig. 7. Histogram result of running the algorithm with the 2D example parameters in the IBM quantum computer Brisbane

state-of-the-art quantum computers make the implementation of this algorithm unpractical on real hardware. The validation of the algorithm was therefore carried out through quantum simulations, which provide a robust foundation for future research into quantum sieving techniques.

At present, the primary limitation for the implementation of large-scale quantum algorithms is not scalability, but rather the quantum noise inherent in current hardware. The results obtained from real quantum devices were so noisy that it was impossible to confirm whether the algorithm functioned as intended. Future work will depend on advances in quantum error correction techniques, which we await before attempting further implementation on noisy quantum hardware.

The final stage of the algorithm involved quantum sieving, but given the level of noise encountered during execution, its implementation would have been impractical. The overall qubit cost for the full algorithm would not have been excessive, as the QFT would have been performed within the same register Y used to store the lattice superposition.

Acknowledgements

D.R.R. thanks the project “Estudio de la Seguridad de los Algoritmos Postcuánticos basados en Retículos (ESAPQR)”, Ref.: 202550E228, and M.A.G.T. thanks the project “Seguridad Algebraica y Aplicaciones de la Inteligencia Artificial en la Criptología Actual y Postcuántica (SAIACAP)” (EFiDiP), Ref.: 202250E141, both of the CSIC for its support.

References

[1] P. W. Shor, “Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer,” *SIAM Journal on Computing*, vol. 26, no. 5, p. 1484â€“1509, Oct. 1997. [Online]. Available: http://dx.doi.org/10.1137/S0097539795293172

[2] Z. Zheng, K. Tian, and F. Liu, *Modern Cryptography Volume 2: A Classical Introduction to Informational and Mathematical Principle*, ser. Financial Mathematics and Fintech. Springer Nature Singapore, 2022. [Online]. Available: https://books.google.es/books?id=pUJmzwEACAAJ

[3] A. M. Childs and W. van Dam, “Quantum algorithms for algebraic problems,” *Reviews of Modern Physics*, vol. 82, no. 1, p. 1â€“52, Jan. 2010. [Online]. Available: http://dx.doi.org/10.1103/RevModPhys.82.1

[4] C. Bright, “From the shortest vector problem to the dihedral hidden subgroup problem,” Technical report, University of Waterloo, 2013. [Online]. Available: https://cs.uwaterloo.ca/research/tr/2013/CS-2013-10.pdf

[5] G. Kuperberg, “A subexponential-time quantum algorithm for the dihedral hidden subgroup problem,” *SIAM Journal on Computing*, vol. 35, no. 1, pp. 170–188, 2005. [Online]. Available: https://doi.org/10.1137/S0097539703436345

[6] Y. Chen, Z. Hu, Q. Liu, H. Luo, and Y. Tu, “Learning with errors with quantum amplitudes: Algorithm, hardness, and oblivious sampling,” 2024. [Online]. Available: https://arxiv.org/abs/2310.00644

[7] R. Kannan, “Minkowski’s convex body theorem and integer programming,” *Mathematics of Operations Research*, vol. 12, no. 3, pp. 415–440, 1987. [Online]. Available: http://www.jstor.org/stable/3689974

Análisis de FAEST como Protocolo de Firma Digital Postcuántica

Fernando Contreras Alcalá
C. U. de Tecnología y Arte Digital, UTAD
I. Tec. Físicas y de la Información
ITEFI-CSIC, Madrid
fernando.contreras@csic.es

Miguel Ángel González de la Torre
I. Tec. Físicas y de la Información
ITEFI-CSIC, Madrid
ma.gonzalez@csic.es

Luis Hernández Álvarez
I. Tec. Físicas y de la Información
ITEFI-CSIC, Madrid
luis.hernandez@csic.es

***Resumen*—Ante la necesidad de nuevos protocolos de firma digital generada por los avances de la computación cuántica y de los algoritmos de criptoanálisis cuánticos, la propuesta FAEST presenta una solución basada en funciones de alta fiabilidad, ya estandarizadas y en cuya seguridad se confía abiertamente. A través del uso de computación multipartita simulada con VOLE-*ith*, y de su prueba interna de conocimiento nulo QuickSilver, FAEST define un protocolo de firma digital seguro y eficiente, siendo una propuesta sólida en la segunda ronda de la convocatoria adicional de firmas digitales postcuánticas realizada por el National Institute of Standards and Technology (NIST). En este trabajo se estudia y descompone cada una de las partes de FAEST con el objetivo de comprender su funcionamiento y las principales diferencias entre sus variantes.**

***Index Terms*—Criptología asimétrica, criptografía postcuántica, FAEST, firma digital.**

I. Introducción

Una vez conocida la vulnerabilidad de la criptografía precuántica gracias a los algoritmos cuánticos de Shor [1], que permiten resolver los problemas de la factorización de enteros y el logaritmo discreto, en tiempo polinómico, y el de Grover [2], que lleva a cabo una búsqueda exhaustiva de claves en los algoritmos simétricos, reduciendo su seguridad a la mitad del tamaño de las claves actuales, el NIST (*National Institute of Standards and Technology*) hizo un llamamiento en busca de algoritmos criptográficos postcuánticos [3] para resolver la falta de seguridad que supondría la implementación de los citados algoritmos cuánticos.

Como consecuencia de este llamamiento, el NIST ha seleccionado dos algoritmos como mecanismos de encapsulado de claves o KEM (*Key Encapsulation Mechanism*), ML-KEM [4] y HQC [5] y como esquemas de firma digital segura ha seleccionado como estándares tres algoritmos: ML-DSA [6], FALCON [7] y SLH-DSA [8]. El hecho de que la seguridad de las dos primeras firmas mencionadas dependa de problemas definidos en retículos (que no han sido lo suficientemente analizados) y que la última base su fortaleza en las funciones hash, con un rendimiento algo escaso, ha llevado al NIST a realizar una convocatoria extraordinaria [9] que busca algoritmos de firma digital alternativos con bases matemáticas diferentes. Las 14 propuestas que han llegado a la segunda ronda de esta convocatoria adicional fundamentan su seguridad en las siguientes áreas matemáticas:

- Teoría de Códigos/Hashes
- Isogenias entre curvas elípticas
- Retículos
- Computación Multipartita Simulada
- Sistemas de ecuaciones multivariantes cuadráticas
- Cifradores Simétricos

De las seis áreas mencionadas, nos detendremos especialmente en la última, que se aprovecha de algoritmos de cifrado clásicos rediseñados de tal forma que permiten generar firmas digitales. Al inicio de la convocatoria adicional de firmas, se seleccionaron cuatro posibles candidatos: AIMer, Ascon-Sign, FAEST y SPHINCS-alpha, pero el único que ha sobrevivido hasta la segunda ronda ha sido FAEST [10]. En este estudio se analizarán los algoritmos internos del esquema con el fin de aportar un análisis claro y sencillo de la propuesta, presentando las principales características de este algoritmo y estudiando las diferencias entre sus variantes.

El documento seguirá la siguiente estructura: en la sección II se explicará el funcionamiento interno de los algoritmos principales sobre los que se apoya FAEST, para en la sección III analizar el funcionamiento interno del protocolo de firma. La sección IV señalará las principales diferencias a nivel de implementación con los algoritmos postcuánticos estandarizados, y en la sección V se expondrán las conclusiones extraídas.

II. Algoritmos

II-A. AES

Una parte fundamental en diseño de FAEST es el uso de AES [11], criptosistema de clave simétrica estandarizado por el NIST en 2001, para el que la amenaza postcuántica solo reduce su seguridad a la mitad la longitud de sus claves, como ya se ha comentado. Como es sabido, AES utiliza cuatro transformaciones principales: SubBytes, que aplica una tabla de sustitución (S-box) a cada byte; ShiftRows, que permuta los bytes del bloque; MixColumns, que mezcla la información del bloque; y AddRoundKey, en la que se calcula la función XOR del estado interno y la subclave correspondiente.

La parte más relevante para el funcionamiento interno de FAEST es la sustitución a través de la S-box. Este proceso (realizado internamente mediante una tabla) calcula el inverso de un byte:

$$\tilde{a} = a^{-1} \bmod (x^8 + x^4 + x^3 + x + 1), \tag{1}$$

y luego aplica la siguiente transformación afín:

$$a_i = \tilde{a}_i \oplus \tilde{a}_{(i+4) \bmod 8} \oplus \tilde{a}_{(i+5) \bmod 8} \oplus \tilde{a}_{(i+6) \bmod 8} \oplus \tilde{a}_{(i+7) \bmod 8} \oplus 0x63. \tag{2}$$

Calcular el inverso en un grupo multiplicativo requiere de operaciones no lineales, característica crucial para hacer de AES un algoritmo seguro, y de la que además se sirve el algoritmo QuickSilver [12] para conseguir generar una prueba de conocimiento nulo o ZKP (*Zero-Knowledge Proof*) que funcionará como verificador de la firma.

II-B. VOLE

VOLE-*ith* surge como consecuencia de la evolución de interacciones de intercambio información mediante OT (*Oblivious Transfer*). Rabin propuso en 1981 una interacción asimétrica, basada en el RSA, en la que un emisor (Alice) envía un secreto a un receptor (Bob), pero sin que Alice sepa si Bob obtiene el secreto, ni Bob gane ninguna información útil si no lo adquiere [13]. Si se aplica este proceso para un intercambio de información secreta, en el que solo uno de los mensajes es útil, Bob tiene $1/2$ de probabilidad de fracaso, pudiendo mejorarse a $1/2^k$ si se repite la interacción k veces.

En 1985 Even et al. propusieron en [14] una generalización del algoritmo que, en vez de enviar todo o nada, divide el secreto en dos mitades y manda una de ellas en función de un bit escogido por el receptor. Esta propuesta se aplica a firmas digitales para lograr un intercambio de información simétrico y se basa en el principio conocido como 1-out-of-2 OT.

Este mecanismo funciona como posible 1-out-of-N y k-out-of-N si se realizan múltiples 1-out-of-2 OT y se concatenan los resultados [15], pero fueron Naor y Pinkas los que propusieron implementaciones más eficientes [16], [17], llegando a publicar el protocolo conocido como OPE (*Oblivious Polynomial Evaluation*) [18].

Tras la propuesta de OPE, Wolf y Wullschleger propusieron una simplificación, llamada *Oblivious Linear-Function Evaluation* [19] (OLE), considerado:

$$y = ax + b, \tag{3}$$

donde Alice conoce a y b, y Bob manda x para recibir la y correspondiente. Con esta definición, se propone el siguiente esquema de compromiso:

Alice tiene como entrada a y elige un valor aleatorio $b \in \mathbb{F}_q$ e introduce (a, b) en un sistema OLE (*Oblivious Linear Evaluation*) sobre el cuerpo finito $\mathbb{F}_q$ ($\mathbb{F}_q$-OLE). Bob elige entonces $x \in \mathbb{F}_q$ y lo introduce en $\mathbb{F}_q$-OLE para obtener y. Finalmente, Alice revela (a, b). Si $a \cdot x + b = y$, Bob acepta el compromiso.

Para generalizar OLE a Vector-OLE o VOLE, solo hace falta repetirlo múltiples veces, permitiendo que se puedan realizar las operaciones en forma matricial. Boyle et al. propusieron compresiones eficientes para VOLE [20], como reutilizar la x en todas la instancias o permitir relaciones entres los distintos a_i y/o b_i. Se define entonces el esquema VOLE mediante la siguiente expresión:

$$\vec{q} = \vec{u} \cdot \Delta + \vec{v}, \tag{4}$$

donde $\vec{u}$ y $\vec{v}$ son dos vectores de la misma dimensión conocidos por uno solo de los participantes, Δ es la entrada del participante opuesto y $\vec{q}$ es el vector de valores retornados del sistema VOLE ante la entrada Δ.

Tras los avances propuestos de OT basados en funciones polinómicas [21], Döttling et al. propusieron TinyOLE [22], un 2PC (*2-Party Computation*) activo (resistente a intentos de engaño por cualquiera de los dos participantes) que implementa la idea de utilizar triples multiplicativos propuesta en el protocolo MASCOT [23]. Tanto MASCOT como TinyOLE introducen autenticación de variables via MAC (*Message Authentication Code*). Estas propuestas emplean el esquema de compromiso basado en OLE para MAC, de modo que para un $a \in \mathbb{F}_q$ Alice genera el valor autenticado $[a]_A = (a, K_a, M_a)$, donde

$$a \cdot \Delta_A + K_a = M_a, \quad \text{donde } M_a, K_a \in \mathbb{F}_q \tag{5}$$

y Δ_A actúa como clave global (simétrica) de Alice para este MAC, usando el mismo valor que está en el OLE para autenticar multiples mensajes. Así, sobre $[a]$ se tiene que Alice tiene acceso a a y K_a y Bob puede verificar con K_a y el valor Δ_A de Alice, una vez ella revele sus valores ocultos.

Con estas bases, se puede comprender el protocolo 2PC de TinyOLE, que consiste en generar un valor autenticado z igual a la suma o multiplicación de dos valores autenticados x e y. Se proponen entonces las siguientes dos funcionalidades:

Se denota como $[a_A]_B$ a la variable a con la que Alice trabaja localmente, autenticada por Bob. De esta forma, el uso de [ ] indica que una variable está autenticada sosteniéndose sobre el sistema de MAC previamente explicado.

- Alice conoce Δ_B y Bob Δ_A.
- Alice pide el valor autenticado $[x_A]$, oculta su valor secreto $\mathbf{x}$ en $x_B = \mathbf{x} - x_A$ y envía a Bob $[x_B]$ con $[x] = [x_A|x_B]$. Bob hace lo mismo para su secreto $\mathbf{y}$.
- **Caso SUMA:** Alice calcula su $[z_A]_A = [x_A]_A + [y_A]_A$, y Bob hace lo mismo para $[z_B]_B$ tal que $[z] = [x] + [y] = [z_A|z_B] = [x_A|x_B] + [y_A|y_B]$.
- **Caso MULTIPLICACIÓN:** Se piden al sistema OLE tripletes autenticados $[a] = [a_A|a_B]$, $[b] = [b_A|b_B]$ y $[c] = [c_A|c_B]$ que cumplen que $c = a \cdot b$. Con esto, se busca calcular $z = x \cdot y$:
  1. Alice calcula $\delta_{aA} = a_A - x_A$ y $\delta_{bA} = b_A - y_A$, y Bob hace lo mismo para sus versiones de dichas variables.
  2. Alice y Bob se intercambian sus δ's y ambos calculan $\delta_x = \delta_{aA} + \delta_{aB}$ y $\delta_y = \delta_{bA} + \delta_{bB}$.
  3. Alice guarda $\delta = \delta_x \cdot \delta_y$ y lo autentica ($[\delta_A]_A$).
  4. Alice calcula $[z_A]_A = [c_A]_A - [x_A]_A \cdot \delta_y - [y_A]_A \cdot \delta_x - [\delta_A]_A$.
  5. Alice calcula $[z_B]_B = [c_B]_B - [x_B]_B \cdot \delta_y - [y_B]_B \cdot \delta_x$.

Si Bob quiere averiguar $\mathbf{x}$, Alice revela x_A y si Alice quiere averiguar $\mathbf{y}$, Bob revela y_B. Se logra de esta forma un sistema mediante compromiso de valor compartido con secreto seguro y resistente a intentos de engaño.

Este proceso es correcto dado que se tiene:

$$\begin{aligned}
&(a_A + a_B) \cdot (b_A + b_B) \\
=&(x_A + \delta_{aA} + x_B + \delta_{aB}) \cdot (y_A + \delta_{bA} + y_B + \delta_{bB}) \\
=&(x_A + x_B + \delta_x) \cdot (y_A + y_B + \delta_y) \\
=&x \cdot y + y_A\delta_x + y_B\delta_x + x_A\delta_y + x_B\delta_y + \delta_x\delta_y,
\end{aligned} \tag{6}$$

donde cada participante utiliza:

$$z_A = c_A - y_A\delta_x - x_A\delta_y - \delta_x\delta_y, \tag{7}$$

$$z_B = c_B - y_B\delta_x - x_B\delta_y, \tag{8}$$

logrando que $[z] = [z_A|z_B]$. Así se obtiene un sistema VOLE eficiente, resistente a engaños, verificable y seguro.

II-C. QuickSilver

Sobre un sistema VOLE funcional, se ha buscado aprovechar su funcionalidad como sistema eficiente de ZK (*Zero Knowledge*, dado que permite el conocimiento de un valor de un polinomio a través de sus valores) en la implementación de circuitos electrónicos, gracias a que funcionan como sistemas no lineales y son compatibles. Las operaciones no lineales de AES previamente mencionadas pueden simularse en un sistema de circuitos [24] con el objetivo de conseguir una ZKP (*Zero Knowledge Proof*) [25] de que la ejecución se ha hecho de forma correcta, sin que ni el emisor ni cualquier atacante haya modificado la información con el objetivo de engañar al receptor. Aunque ha habido propuestas como Wolverine [26] y Mac'n'Cheese [27], QuickSilver [12] es una implementación híbrida de ambas que propone funcionalidad de demostración de testigo tanto sobre circuitos booleanos como sobre polinomios. El caso de los circuitos se comenta a continuación.

Inputs: Alice y Bob tienen un circuito C sobre $\mathbb{F}_q$ con t puertas multiplicativas. Alice además tiene un testigo w de $|w| = n$ que cumple que $C(w) = 1$. En QuickSilver se aplica autenticación y enmascaramiento de variables, de modo que si $[w_i]$ es una variable enmascarada que comparten ambos participantes, se tiene que $k_i = m_i + w_i \cdot \Delta$, donde Alice tiene acceso a w_i y m_i, $\forall i$, mientras que Bob solo tiene acceso a las k_i y a Δ.

Preprocesamiento:

1. El sistema VOLE genera una clave $\Delta \in \mathbb{F}_{p^r}$ para Bob.
2. El sistema VOLE envía también los conjuntos de valores autenticados $\{[\mu_i]\}_{i\in\{1,...,n\}}$ y $\{[v_i]\}_{i\in\{1,...,t\}}$.
3. El sistema VOLE envía a Alice dos valores escalares de enmascaramiento (A_0^*, A_1^*) y su correspondiente $B^* = A_0^* + A_1^* \cdot \Delta$ a Bob.

Fase en Línea: Se revela el circuito C y el vector testigo w.

4. Alice envía a Bob $\delta_i := w_i - \mu_i \in \mathbb{F}_q$, y los dos calculan $[w_i] := [\mu_i] + \delta_i$.
5. Para cada puerta $(\alpha, \beta, \gamma, T) \in C$ (entrada 1, entrada 2, salida y tipo de operación) se realiza en orden topológico:
   - Si la operación T es suma, ambos calculan $[w_\gamma] := [w_\alpha] + [w_\beta]$.
   - Si la operación T es multiplicación, siendo la puerta número i, Alice manda $d_i = w_\alpha \cdot w_\beta - v_i \in \mathbb{F}_p$ a Bob, y calculan entonces $[w_\gamma] = [v_i] + d_i$.
6. Para cada puerta i, los dos comparten el triple autenticado $([w_\alpha], [w_\beta], [w_\gamma])$.
7. Alice guarda los valores $A_{0,i} := m_\alpha \cdot m_\beta \in \mathbb{F}_{p^r}$ y $A_{1,i} := w_\alpha \cdot m_\beta + w_\beta \cdot m_\alpha - m_\gamma \in \mathbb{F}_{p^r}$.
8. Bob calcula $B_i := k_a \cdot k_\beta - k_\gamma \cdot \Delta \in \mathbb{F}_{p^r}$.
9. Se tiene entonces el conjunto de polinomios $A_{0,i} + A_{1,i} \cdot X$ (siendo X una variable), que permite comprobar que $B_i = A_{0,i} + A_{1,i} \cdot \Delta, \forall i \in 1, ..., t$:
   - Bob manda $\chi \leftarrow \mathbb{F}_{p^r}$ a Alice.
   - Alice calcula $U := \sum_{i\in[t]} A_{0,i} \cdot \chi^i + A_0^*$ y $V := \sum_{i\in[t]} A_{1,i} \cdot \chi^i + A_1^*$ y se los envía a Bob.
   - Bob calcula $W := \sum_{i\in[t]} B_i \cdot \chi^i + B^*$ y comprueba que $W = U + V \cdot \Delta$. Si es cierto, se ha realizado la prueba con éxito.
10. Finalmente, para el cable de salida w_h del circuito C, ambos participantes comparten $[w_h]$ que cumple $k_h = m_h + w_h \cdot \Delta$. Bob comprueba entonces que $w_h = 1$ recibiendo m_h de Alice, despejando la igualdad y comprobando que se cumple para su valor k_h.

En FAEST, se define el testigo que será utilizado en la prueba ZKP como salidas de la S-box tras aplicar la función SubBytes en la extensión de claves de AES.

II-D. VOLE-ith

Hasta ahora, todas las propuestas, además de requerir de un tercer entorno de ejecución $\mathbb{F}_q$-OLE que funcione como caja negra para evitar que Alice descubra x e y, presentan un problema grave ante la implementación en firmas digitales, puesto que el esquema requiere ser no interactivo. Para solventar este problema, Baum et al. proponen una ejecución *in the Head* [28], es decir, que simula las elecciones del verificador para que se pueda ejecutar el proceso sin interacciones intermedias, y consiguiendo así una implementación válida y lograr un esquema no interactivo.

Para ello, parte de la funcionalidad se aprovecha del uso de un árbol GGM (Goldreich, Goldwasser, Micali) [29] que, mediante un PRG (*Pseudo Random Generator*) genera dos hojas hijas del mismo árbol. Tras repetirse este proceso tantas veces como niveles de profundidad se busquen, se usan los valores de cada hoja final (es decir, las que estén en el último nivel del árbol) como semillas sd_i, $i \in [0, N-1]$ con N el número total de hojas finales del árbol. Cada una de estas hojas tendrá un mensaje m_i y su hash com_i. Se calcula entonces el hash h del conjunto de todos los com_i combinados y se lo envía al verificador.

La hoja correspondiente a un índice $j \in [0, N-1]$ permanecerá oculta para el verificador. Una vez definido j, se reconstruye el árbol GGM sin repetir el cálculo recursivo de aquellas hojas donde se cumpla que todas sus hijas que sean conocidas. Los nodos generados en esta segunda construcción son los enviados al verificador, a través de $decom_I$ (siendo I la nomenclatura asociada a la posición j seleccionada, que en versiones posteriores se selecciona aplicando una cierta metodología), que solo tiene acceso a las hojas adyacentes a la que desconoce y las afectadas verticalmente por este hecho. Se le da también al verificador el valor del hash de hoja oculta com_j, pero no el mensaje m_j.

El verificador puede reconstruir el resto del árbol a partir de los valores almacenados en $decom_I$ e intentar calcular h con com_j. Si el probador no ha engañado, el h calculado por el verificador será igual al que envió el probador, por lo que puede confirmar que la ejecución de este proceso fue correcta sin conocer el valor oculto asociado.

Para aplicar este concepto a VOLE-*ith*, se implementa un árbol GGM en el que cada hoja del árbol generará un compromiso de tipo VOLE. Para ganar eficiencia, se genera un mini-VOLE independiente por cada hoja del árbol, y luego se alinean en un subespacio VOLE (para que esto sea posible, es necesario que la clave Δ sea relativamente pequeña), en vez de generar un VOLE grande desde el principio [30]. Baum et

al. [31] propusieron también aplicar *batch all-but-one vector commitments* (BAVC), que permite aumentar la eficiencia asociada al intercambio OT debido a la estructura optimizada del árbol. El árbol GGM permite entonces generar un sistema VOLE completo de una forma computacionalmente eficiente, que se usa para realizar una transferencia All-But-One en la que se consiguen compartir $N-1$ elementos de los N originales, y que es poco costosa en tamaño de firma a la hora de enviarse al verificador.

Del VOLE final alineado se extrae su vector $\vec{u}$, con el que se genera un valor enmascarado del testigo (a través de la operación $d_i = w_i - u_i$), obteniendo el vector d sobre el que se podrá aplicar posteriormente QuickSilver como ZKP [32].

Para lograr que el esquema sea no interactivo, se aprovecha el resultado de una función hash para elegir el índice j, haciendo que esta elección sea aleatoria y evitando que pueda ser falsificada por el probador [33]. Con este sistema no interactivo para generar ZKP, queda entonces ver cómo funciona el protocolo de firma digital.

III. FAEST

FAEST es un criptosistema de firma digital postcuántico basado en la implementación de computación multipartita simulada y primitivas de cifrado simétricas. Surge como alternativa a PICNIC [34], que utilizaba el algoritmo de cifrado por bloques LowMC propuesto para casos de computación multipartita simulada [35]. En cambio, FAEST implementa AES como OWF (*One Way Function*) y también utiliza funciones de la familia SHA-3 [36] para lograr su resistencia ante los avances de la computación cuántica.

En la *Tabla I* se muestran los parámetros internos utilizados por FAEST, cuyos valores dependen del nivel de seguridad o variante a utilizar.

Tabla I
PRINCIPALES PARÁMETROS PARA FAEST Y FAEST-EM.

Parámetro	Descripción
λ	Parámetro de seguridad $(128, 192, 256)$
τ	Número de instancias VOLE
ω_{grind}	w bits superiores de Δ que deben ser cero
T_{open}	Umbral máximo de apertura del árbol GGM
n_{leafcom}	Número de bloques de λ bits en cada compromiso hoja
B	Longitud adicional de relleno para la comprobación VOLE
$E_k(x)$	Función unidireccional usada para generar la clave pública
ℓ	Longitud del testigo para la ZKP (en bits)

Se proponen doce variantes del algoritmo base, diferenciados por los siguientes cambios:

- **En función del tamaño en bits de la clave privada** λ: con valores 128, 192 o 256 bits, asociadas directamente al nivel de seguridad de cada variante.
- **Selección de la OWF:** FAEST considera dos posibles OWF, AES o Rijndael con Even-Mansour como OWF [37]. Se etiquetan con el sufijo -EM las que aplican $OWF_x(k) \oplus k$, y presentan tamaños de firma más pequeños, a cambio de que su seguridad no sea tan irrefutable como la que proporciona AES. En los casos básicos, se utiliza la concatenación de OWF, $\text{AES192}_k(x)||\text{AES192}_k(x \oplus 1)$ y $\text{AES256}_k(x)||\text{AES192}_k(x \oplus 1)$ para lograr resultados de 256 bits.
- **Los parámetros "s" o "f":** que nominan como *slow and short* y *fast but long*. Los procesos de tipo "s" generan firmas más pequeñas, gracias a trabajar en VOLEs sobre $\mathbb{F}_{11}$, mientras que los de tipo "f" se ejecutan en menos tiempo (al a utilizar VOLEs sobre $\mathbb{F}_8$) pero generan firmas más grandes asociadas a una mayor cantidad de VOLEs necesarias (τ).

Los tres algoritmos del esquema de firma digital en FAEST son los siguientes:

III-A. Generación de claves

FAEST requiere de una clave privada, que utiliza Alice para firmar, y una clave pública, que utiliza Bob para verificar la firma. Para obtener la clave privada, Alice genera una clave de λ bits aleatorios, y se asegura de que los dos bits menos significativos sean ambos 1, $sk = k$. Esta clave privada se emplea como clave simétrica de la OWF seleccionada, a continuación, Alice genera un mensaje x de forma aleatoria y computa $\text{OWF}(sk, x) = y$. La clave pública de Alice será entonces la entrada y la salida de dicha ejecución, $pk = (x, y)$, haciendo que la seguridad del algoritmo recaiga en la resistencia de la OWF que se haya utilizado en la generación de claves.

III-B. Elaboración de la firma

El proceso de firma se divide en varios pasos:

1. Como entrada, el algoritmo de generación de firma recibe pk, sk y el mensaje a firmar m.
2. Se generan λ-bits aleatorios y se almacenan en la variable ρ. Se guardan en el valor interno μ el valor del hash del mensaje con la clave pública. Posteriormente, se computa el hash de ρ, μ y la clave privada para obtener los parámetros r e iv^{pre}. Por seguridad, se lleva a cabo un hash adicional de iv^{pre}, generando iv, que se usa, junto a r, para inicializar el árbol GGM.
3. El siguiente proceso consiste en ejecutar el VOLECommit, con entradas r e iv, para generar el árbol GGM que encapsula las llamadas pequeñas a VOLE. Esta función devuelve:
   - **decom:** un decompromiso que contiene información suficiente para reconstruir gran parte del árbol GGM.
   - **com:** compromiso asociado al valor oculto del árbol.
   - **u:** valor fijado en el VOLE.
   - $(\mathbf{c_i})_{\mathbf{i} \in \mathbf{1},\ldots,\tau-\mathbf{1}}$: Valores correctores para la u_i de cada VOLE pequeño en las hojas del árbol GGM.
   - **V:** verificadores del MAC para cada u_i.

   Con esta información, se genera el primer desafío $chall_1 = hash\left(\mu||com||c_1||...||c_{\tau-1}||iv^{pre}\right)$.
4. Se guardan los valores hasheados, con la función VOLEHash, del $chall_1$ con **u** en $\tilde{\mathbf{u}}$ y con **V** en $\tilde{\mathbf{V}}$. Además, tiene lugar una parte crucial en la generación de la ZKP: ejecutando la expansión de clave de AES se guardan en el testigo w ciertos valores intermedios. En las rondas impares el resultado de la S-box completa, y en las pares solo la norma inversa de cada byte. Este testigo **w** se guarda enmascarado con **u** en **d**, y se genera a su vez el segundo desafío $chall_2 = hash\left(chall_1||\tilde{\mathbf{u}}||\tilde{\mathbf{V}}||\mathbf{d}\right)$.

5. A partir del testigo w, los valores **u**, **V**, pk y $chall_2$, se aplica QuickSilver para generar (a_0, a_1, a_2) (en lugar de solo dos coeficientes, modificación realizada en la v2 para lograr un menor tamaño de firmas) generando la ZKP con **w** y pk. Por la definición del protocolo QuickSilver, existe un polinomio $\left(P(x) = a_0 + a_1 x + a_2 x^2 + z x^3\right)$, que cumple que z=0 si el probador no ha engañado a la hora de generar dicho polinomio. Tras esto, los a_i's son enmascarados de la siguiente forma:

$$\tilde{a}_0 := ZKHash\left(chall_2, (a_0 || u_0^*)\right), \tag{9}$$

$$\tilde{a}_1 := ZKHash\left(chall_2, (a_1 || v_0^* + u_1^*)\right), \tag{10}$$

$$\tilde{a}_2 := ZKHash\left(chall_2, (a_2 || v_1^*)\right), \tag{11}$$

donde los u_i^* y v_i^* denotan los valores adaptados de **u** y **V** y ZKHash es una función resumen que conserva las correlaciones entre los elementos al enmascarar. Por las propiedades del VOLE-*ith*, para toda Δ se genera un q^* que verifica $u_0^* + (u_1^* + v_0^*)\Delta + v_1^* \Delta^2 = q^*$.

6. Con estos datos se genera el tercer desafío $chall_3 = H\left(chall_2 || \tilde{a}_0 || \tilde{a}_1 || \tilde{a}_2 || \mathbf{ctr}\right)$, donde **ctr** es un contador que parte de cero y aumenta hasta que el resultado de la asignación anterior devuelva que $chall_3$ tenga a cero los ω bits menos significativos. Este valor representa la implementación de la transformación Fiat-Shamir, que permite que se cumpla el esquema de firma digital no interactivo. De $chall_3$ se asigna un valor posición en I denotando el nodo oculto en el árbol GGM, y se crea el decompromiso $decom_I$ de VOLE-*ith*.
7. Finalmente, se envía la firma al verificador con la siguiente estructura:

$$\sigma = \left((c_i)_{i \in [1 \ldots \tau)}, \tilde{\mathbf{u}}, \mathbf{d}, \tilde{a_1}, \tilde{a_2}, decom_I, chall_3, iv^{pre}, \mathbf{ctr}\right). \tag{12}$$

El tamaño final, σ, de la firma viene dado por:

$$\sigma = \tau \cdot (\ell + 3\lambda + B) + T_{open} \cdot \lambda + n_{leafcom} \lambda \tau + \lambda + 128 + 32, \tag{13}$$

donde la primera parte contiene los compromisos del testigo y las respuestas con los verificadores del VOLE, la segunda y tercera son la apertura del árbol GGM, la cuarta son λ bits para el Δ, la quinta son 128 bits de salt o aleatoriedad y los últimos 32 bits son para el contador asociado al pulido de Δ.

III-C. Verificación de la firma

Para comprobar la firma, el verificador debe de realizar los siguientes pasos:

1. Recibe σ (ver *Ec.* (12)).
2. Calcula μ e iv de la misma forma a como lo hizo el firmante. Comprueba que $chall_3$ tenga los últimos ω_{grind} bits a cero (este número se deduce de la variante del esquema de firma utilizado), devolviendo error de firma si no se cumple esta condición.
3. Reconstruye el árbol GGM de VOLEs con $decom_I$, $chall_3$, las correcciones $\{c_i\}_{i \in \{1,\ldots,\tau-1\}}$ y el parámetro de aleatoriedad iv. Calcula el valor del compromiso *com* por las propiedades de correlación del VOLE. Si el proceso falla, devuelve error en la verificación. Guarda también el valor **Q** del VOLE.
4. Calcula el hash $chall_1$ con $\mu, com, \mathbf{c}_i, iv^{pre}$. Con este valor y **Q**, calcula $\tilde{\mathbf{Q}}$ (es crucial recordar que VOLEHash conserva la correlación del sistema VOLE). Luego, halla $\mathbf{D} := \tilde{\mathbf{Q}} \oplus \tilde{\mathbf{u}} \cdot \delta_i$, donde cada delta es el desafío al mini-VOLE extraído de $chall_3$. Como $\tilde{\mathbf{Q}} = \tilde{\mathbf{u}} \cdot \delta + \tilde{\mathbf{V}}$, se cumple que $\tilde{\mathbf{V}} = \tilde{\mathbf{Q}} + \tilde{\mathbf{u}} \cdot \delta$ (con $\oplus$ como suma), por lo que $\tilde{\mathbf{V}}$ debe ser igual a **D**. Con este desarrollo, se puede calcular $chall_2$ con un hash de $chall_1$, $\tilde{\mathbf{u}}$, **D** y **d**.
5. Se aplica la verificación de QuickSilver usando los valores **d**, **Q**, $chall_2, chall_3, \tilde{a}_1, \tilde{a}_2$, y pk. Para ello, calcula $\mathbf{w}_i = \mathbf{Q}_i + \mathbf{d}_i \cdot \Delta$, y genera $P(x)$ con pk igual que el firmante. Adapta **Q** a q^*, calcula $P(\Delta) = \mathbf{b}$, y halla $\tilde{q} := ZKHash\left(chall_2, (\mathbf{b} || q^*)\right)$, que genera $\tilde{a_0}$ al cumplirse la siguiente igualdad:

$$\tilde{q} = \tilde{a}_0 + \tilde{a}_1 \Delta + \tilde{a}_2 \Delta^2, \tag{14}$$

$$\tilde{a}_0 := \tilde{q} - \tilde{a}_1 \Delta - \tilde{a}_2 \Delta^2. \tag{15}$$

6. Finalmente, el verificador calcula el hash $chall_3'$ con $\tilde{a}_0$, $\tilde{a}_1$, $\tilde{a}_2$, $chall_2$ y el contador **ctr**, y verifica si $chall_3' = chall_3$. En caso afirmativo, la verificación es correcta, y en caso contrario devuelve error de verificación.

IV. Comparaciones

La *Tabla II* muestra el nivel de seguridad asociado a cada algoritmo, el tamaño de las claves públicas (pk) y el de las firmas en bytes, así como el número de ciclos de procesador (en miles) tanto para las firma como para las verificaciones, para distintos algoritmos post-cuánticos. Aunque este esquema no es competitivo comparado con las firmas basadas en retículos, ofrece una alternativa fiable a SLH-DSA, cuyas firmas ocupan el doble de tamaño y, como se puede comprobar, requiere de una cantidad considerablemente menor de ciclos de procesador para generar firmas del mismo nivel de seguridad.

Tabla II
Comparativa de diferentes parámetros para distintos algoritmos de firma post-cuánticos.

Algoritmo	Nivel	Tamaño		Ciclos	
		pk	Firma	Firma	Verif.
FAEST-128s	Cat. 1	32	4 506	12 787	9 783
FAEST-128f	Cat. 1	32	5 924	1 722	1 413
FAEST-192s	Cat. 3	48	11 260	54 687	42 290
FAEST-192f	Cat. 3	48	14 948	7 045	6 079
FAEST-256s	Cat. 5	64	20 696	76 330	74 546
FAEST-256f	Cat. 5	64	26 548	11 071	10 241
FAEST-EM-128s	Cat. 1	32	3 906	9 403	7 398
FAEST-EM-128f	Cat. 1	32	5 060	1 404	1 113
FAEST-EM-192s	Cat. 3	48	9 340	39 282	36 239
FAEST-EM-192f	Cat. 3	48	12 380	5 117	4 665
FAEST-EM-256s	Cat. 5	64	17 984	62 465	59 738
FAEST-EM-256f	Cat. 5	64	23 476	9 436	8 725
ML-DSA-44	Cat. 1	1 312	2 420	333	118
ML-DSA-65	Cat. 3	1 952	3 309	529	179
ML-DSA-87	Cat. 5	2 592	4 627	642	280
Falcon-512	Cat. 1	897	666	1 010	81
Falcon-1024	Cat. 5	1 793	1 280	2 053	161
SLH-DSA-SHAKE-128s	Cat. 1	32	7 856	4 683 000	4 764 000
SLH-DSA-SHAKE-128f	Cat. 1	32	17 088	239 794	12 910
SLH-DSA-SHAKE-192s	Cat. 3	48	16 224	8 091 420	6 465 506
SLH-DSA-SHAKE-192f	Cat. 3	48	35 664	386 862	19 876 926
SLH-DSA-SHAKE-256s	Cat. 5	64	29 792	7 085 272	10 216 560
SLH-DSA-SHAKE-256f	Cat. 5	64	49 856	763 942	19 886 032

V. Conclusiones

En este trabajo se han analizado los algoritmos fundamentales que componen el protocolo de firma digital FAEST, estudiando su funcionamiento y ofreciendo una visión estructurada del esquema básico de la propuesta. En futuras

investigaciones, se analizará la seguridad matemática de las diversas implementaciones y se estudiarán las implementaciones de las variantes y su resistencia ante ataques de canal lateral, inducción de fallos e inteligencia artificial, a través de la información extraída en sus trazas de ejecución.

AGRADECIMIENTOS

F.C.A. agradece al programa JAE Intro del ITEFI-CSIC su apoyo. M.A.G.T. y L.H.A. agradecen al proyecto Seguridad Algebraica y Aplicaciones de la Inteligencia Artificial en la Criptología Actual y Postcuántica (SAIACAP), Ref.: 202450E017, del CSIC su apoyo.

REFERENCIAS

[1] P.W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, 1997. [Online]. Available: http://dx.doi.org/10.1137/S0097539795293172

[2] L.K. Grover, "A fast quantum mechanical algorithm for database search," in *Proc. 28th Annual ACM Symposium on Theory of Computing*, STOC '96. ACM, 1996, pp. 212–219. [Online]. Available: https://doi.org/10.1145/237814.237866

[3] NIST, "Call for additional digital signature schemes for the post-quantum cryptography standardization process," 2022. [Online]. Available: https://csrc.nist.gov/Projects/pqc-dig-sig/standardization/call-for-proposals

[4] ——, "Module-lattice-based key-encapsulation mechanism (ML-KEM)," FIPS 203, 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.203.pdf

[5] HQC Consortium, "Hamming quasi-cyclic (HQC) specification," Submission to NIST Post-Quantum Cryptography Project, 2025. [Online]. Available: https://pqc-hqc.org/doc/hqc_specifications_2025_08_22.pdf

[6] NIST, "Module-lattice-based digital signature standard (ML-DSA)," FIPS 204, 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.204.pdf

[7] ——, "Falcon update (2024)," NIST CSRC Presentation, 2024. [Online]. Available: https://csrc.nist.gov/Presentations/2024/falcon

[8] ——, "Stateless hash-based digital signature standard (SLH-DSA)," FIPS 205, 2024. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.205.pdf

[9] ——, "Post-quantum cryptography: Additional digital signature schemes," 2022. [Online]. Available: https://csrc.nist.gov/projects/pqc-dig-sig

[10] C. Baum, W. Beullens, L. Braun, C.D. de Saint Guilhem, M. Klooß, C. Majenz, S. Mukherjee, E. Orsini, S. Ramacher, C. Rechberger *et al.*, "FAEST v2: Algorithm specifications," 2025. [Online]. Available: https://faest.info/

[11] NIST, "Advanced encryption standard (AES)," FIPS 197, 2001. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197-upd1.pdf

[12] K. Yang, P. Sarkar, C. Weng, and X. Wang, "QuickSilver: efficient and affordable zero-knowledge proofs for circuits and polynomials over any field," in *Proc. 2021 ACM SIGSAC Conference on Computer and Communications Security*, CCS '21. ACM, 2021, pp. 2986–3001. [Online]. Available: https://doi.org/10.1145/3460120.3484556

[13] M.O. Rabin, "How to exchange secrets by oblivious transfer," *Technical Report TR-81*, Aiken Computation Lab, Harvard University, 1981. [Online]. Available: https://eprint.iacr.org/2005/187.pdf

[14] S. Even, O. Goldreich, and A. Lempel, "A randomized protocol for signing contracts," *Commun. ACM*, vol. 28, no. 6, pp. 637–647, 1985. [Online]. Available: https://doi.org/10.1145/3812.3818

[15] Y. Ishai, J. Kilian, K. Nissim, and E. Petrank, "Extending oblivious transfers efficiently," in *Proc. CRYPTO 2003*. Springer, Berlin, 2003, pp. 145–161. [Online]. Available: https://doi.org/10.1007/978-3-540-45146-4_9

[16] M. Naor and B. Pinkas, "Oblivious transfer and polynomial evaluation," in *Proc. Thirty-First Annual ACM Symposium on Theory of Computing*, STOC '99. ACM, 1999, pp. 245–254. [Online]. Available: https://doi.org/10.1145/301250.301312

[17] ——, "Computationally secure oblivious transfer," *J. Cryptology*, vol. 18, no. 1, pp. 1–35, 2005. [Online]. Available: https://doi.org/10.1007/s00145-004-0102-6

[18] ——, "Oblivious polynomial evaluation," *SIAM J. Comput.*, vol. 35, no. 5, pp. 1254–1281, 2006. [Online]. Available: https://doi.org/10.1137/S0097539704383633

[19] S. Wolf and J. Wullschleger, "Oblivious transfer is symmetric," in *Proc. EUROCRYPT 2006*, Springer, Heidelberg, 2006, pp. 222–232. [Online]. Available: https://doi.org/10.1007/11761679_14

[20] E. Boyle, G. Couteau, N. Gilboa, and Y. Ishai, "Compressing vector OLE," ser. CCS '18. ACM, 2018, pp. 896–912. [Online]. Available: https://doi.org/10.1145/3243734.3243868

[21] N. Döttling, D. Kraschewski, and J. Müller-Quade, "Statistically secure linear-rate dimension extension for oblivious affine function evaluation," in *Proc. 6th International Conference on Information Theoretic Security*, ICITS'12. Springer, Heidelberg, 2012, pp. 111–128. [Online]. Available: https://doi.org/10.1007/978-3-642-32284-6_7

[22] N. Döttling, S. Ghosh, J.B. Nielsen, T. Nilges, and R. Trifiletti, "TinyOLE: Efficient actively secure two-party computation from oblivious linear function evaluation," CCS '17. ACM, 2017, pp. 2263–2276. [Online]. Available: https://doi.org/10.1145/3133956.3134024

[23] M. Keller, E. Orsini, and P. Scholl, "MASCOT: Faster malicious arithmetic secure computation with oblivious transfer," CCS '16. ACM, 2016, pp. 830–842. [Online]. Available: https://doi.org/10.1145/2976749.2978357

[24] J. Boyar and R. Peralta, "A small depth-16 circuit for the AES S-Box," in *Proc. Information Security and Privacy Research*, vol. 376. Springer, Heidelberg, 2012, pp. 287–298. [Online]. Available: https://doi.org/10.1007/978-3-642-30436-1_24

[25] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," *SIAM J. Comput.*, vol. 18, no. 1, pp. 186–208, 1989. [Online]. Available: https://doi.org/10.1137/0218012

[26] J. Li, H. Ma, J. Wang, Z. Song, W. Xu, and R. Zhang, "Wolverine: A scalable and transaction-consistent redactable permissionless blockchain," *Trans. Info. For. Sec.*, vol. 18, pp. 1653–1666, 2023. [Online]. Available: https://doi.org/10.1109/TIFS.2023.3245406

[27] C. Baum, A. J. Malozemoff, M. B. Rosen, and P. Scholl, "Mac'n'Cheese: Zero-knowledge proofs for Boolean and arithmetic circuits with nested disjunctions," in *Proc. CRYPTO 2021*. Springer, Heidelberg, 2021, pp. 92–122. [Online]. Available: https://doi.org/10.1007/978-3-030-84259-8_4

[28] C. Baum, L. Braun, C.D. de Saint Guilhem, M. Klooβ, E. Orsini, L. Roy, and P. Scholl, "Publicly verifiable zero-knowledge and post-quantum signatures from VOLE-in-the-Head," in *Proc. CRYPTO 2023*, vol. 14085. Springer, Cham, 2023, pp. 553–584. [Online]. Available: https://doi.org/10.1007/978-3-031-38554-4_19

[29] O. Goldreich, S. Goldwasser, and S. Micali, "How to construct random functions," *J. ACM*, vol. 33, no. 4, pp. 792–807, 1986. [Online]. Available: https://doi.org/10.1145/6490.6503

[30] L. Roy, "SoftSpokenOT: Quieter OT extension from small-field silent VOLE in the minicrypt model," in *Proc. CRYPTO 2022*. Springer, Heidelberg, 2022, pp. 657–687. [Online]. Available: https://doi.org/10.1007/978-3-031-15802-5_23

[31] C. Baum, W. Beullens, S. Mukherjee, E. Orsini, S. Ramacher, C. Rechberger, L. Roy, and P. Scholl, "One tree to rule them all: Optimizing GGM trees and OWFs for post-quantum signatures," in *Proc. ASIACRYPT 2024*. Springer, Singapore, 2024, pp. 463–493. [Online]. Available: https://doi.org/10.1007/978-981-96-0875-1_15

[32] C. Baum, S. Dittmer, P. Scholl, and X. Wang, "SoK: Vector OLE-based zero-knowledge protocols," *Des. Codes Cryptog.*, vol. 91, no. 11, pp. 3547–3585, 2023. [Online]. Available: https://doi.org/10.1007/s10623-023-01292-8

[33] A. Fiat and A. Shamir, "How to prove yourself: Practical solutions to identification and signature problems," in *Proc. CRYPTO' 86*. Springer, Heidelberg, 1987, pp. 186–194. [Online]. Available: https://doi.org/10.1007/3-540-47721-7_12

[34] G. Zaverucha, D. Stebila, C. Paquin, D. Cooper, M. Naehrig, and S. Ranellucci, "The Picnic signature algorithm specification, version 3.0," 2020. [Online]. Available: https://microsoft.github.io/Picnic/

[35] ——, "Founding cryptography on oblivious transfer – efficiently," in *Proc. CRYPTO 2008*, vol. 5157. Springer, Berlin, 2008, pp. 572–591. [Online]. Available: https://doi.org/10.1007/978-3-540-85174-5_32

[36] NIST, "SHA-3 standard: Permutation-based hash and extendable-output functions," FIPS 202, 2015. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[37] S. Even and Y. Mansour, "A construction of a cipher from a single pseudorandom permutation," *J. Cryptology*, vol. 10, no. 3, pp. 151–161, 1997. [Online]. Available: https://link.springer.com/article/10.1007/s001459900025

[38] PQShield research team, *Post-Quantum signatures zoo*, 2024. Disponible en: https://pqshield.github.io/nist-sigs-zoo/wide.html.

DataSecOnt: Ontología para la Seguridad de los Datos de Google Play

Alejandra Gavino-Dias González
Grupo de Inv. en Ingeniería de la Privacidad (IngPriv)
Universidad de Valladolid, España
alejandra.gavinodias@gmail.com

M. Mercedes Martínez-González
Grupo de Inv. en Ingeniería de la Privacidad (IngPriv)
Universidad de Valladolid, España
mercedes@infor.uva.es

Alejandro Pérez-Fuente
Grupo de Inv. en Ingeniería de la Privacidad (IngPriv)
Universidad de Valladolid, España
alejandro.perez.fuente@uva.es

***Resumen*—La preocupación por el riesgo para la privacidad que conlleva el uso de aplicaciones móviles aumenta en paralelo con su expansión como herramienta cotidiana. En este contexto, la falta de estandarización en la descripción de los tipos de datos y sus propósitos dificulta el análisis sistemático de las declaraciones con las que se informa a los usuarios sobre las prácticas de privacidad de las aplicaciones. Este trabajo presenta DataSecOnt, una ontología diseñada para estructurar y representar el tratamiento de los datos personales en el ecosistema de Google Play. Permite mejorar la interoperabilidad, la trazabilidad y el análisis semántico de la información declarada por los desarrolladores. DataSecOnt ofrece un modelo formal capaz de clasificar los tipos de datos, sus finalidades y los riesgos asociados a su tratamiento, proporcionando una base sólida para auditorías automáticas y sistemas de evaluación de riesgos en aplicaciones móviles. La ontología se ha construido siguiendo la metodología *Ontology Development 101*, en combinación con el documento *Ontology Requirements Specification Document (ORSD)*, e implementada íntegramente en RDF/OWL.**

***Index Terms*—Aplicación móvil, OWL, RDF, privacidad, Google Play, Web semántica, datos personales**

I. Introducción

La digitalización del ecosistema móvil ha impulsado la creación de millones de aplicaciones que acceden a información personal sensible[1]. En un esfuerzo de mejorar la comprensión de los usuarios de la información sobre el tratamiento de sus datos personales, Google introdujo las conocidas como *Privacy Labels*, que los usuarios encuentran bajo el epígrafe *Seguridad de los Datos* [1], [2], [3], [4].

En la Figura 1 se muestra un ejemplo real de la sección *Seguridad de los Datos* de una aplicación disponible en Google Play[1]. Este apartado resume la información declarada por los desarrolladores sobre la recopilación, el uso y la protección de los datos personales, incluyendo aspectos como el tipo de información tratada, el cifrado durante el tránsito o la posibilidad de solicitar su eliminación.

Para aportar esta información, los desarrolladores informan sobre la recopilación y el uso de datos personales a través de formularios proporcionados por la propia plataforma, que incluyen campos predefinidos y descripciones complementarias [1]. Sin embargo, estos formularios no se basan en una

[1]Android Usage Statistics (2025) – Users & Market Share. Disponible en https://www.demandsage.com/android-statistics/

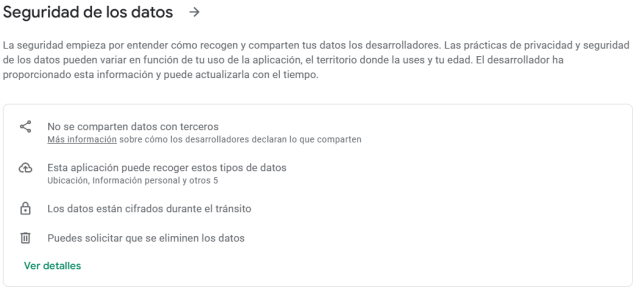


Figura 1: Sección *Seguridad de los Datos* en una aplicación de Google Play.

estructura formal y unificada que defina de manera consistente los conceptos, las relaciones y las categorías utilizadas para describir los tipos de datos y sus tratamientos. Como consecuencia, cada término o categoría puede interpretarse de forma diferente entre aplicaciones, ya que Google Play combina descripciones textuales, listas predefinidas y definiciones dispersas en distintos apartados de su documentación.

Aunque la información disponible en la *Seguridad de los Datos* constituye un avance hacia una mejor transparencia e información a los usuarios, esta ausencia de una base semántica que aporte interoperabilidad semántica dificulta la interpretación automática, la comparación entre aplicaciones y el análisis de riesgos de privacidad. A su vez esto limita la posibilidad de crear herramientas que ayuden tanto a auditores como a usuarios a evaluar con precisión la seguridad de sus datos. Por ejemplo, un auditor no puede determinar automáticamente si dos aplicaciones tratan el mismo tipo de dato cuando una declara "ubicación aproximada" y otra "datos de localización", pese a referirse conceptualmente a la misma categoría. De igual modo, un usuario no puede saber si "interacciones dentro de la aplicación" y "actividad del usuario" representan prácticas equivalentes o implican distintos niveles de riesgo. Sin una semántica común que unifique estos términos, resulta imposible comparar de forma precisa el alcance real de la recopilación de datos, las finalidades declaradas o las medidas de seguridad aplicadas.

Con el objetivo de resolver este problema, se propone DataSecOnt, una ontología que formaliza las relaciones entre

tipos de datos, finalidades, medidas de seguridad y actores implicados en los tratamientos de datos personales por parte de las aplicaciones. La ontología se ha diseñado con la intención de proporcionar una representación explícita, estructurada y extensible del dominio de la seguridad de datos en aplicaciones móviles, capaz de facilitar consultas automáticas y la integración con otros modelos ontológicos.

Se ha utilizado la metodología *Ontology Development 101* para su diseño, en combinación con el documento *Ontology Requirements Specification Document (ORSD)*. En la fase de implementación, se ha optado por el estándar más expandido para la representación de ontologías formales, RDF/OWL.

En comparación con los modelos previos, DataSecOnt introduce dos aportaciones principales:

1. Su integración de las diversas fuentes heterogéneas proporcionadas por Google Play, incluyendo categorías de datos [1], [2], propósitos y prácticas de seguridad [3], lo que permite unificar y mapear de forma consistente las declaraciones realizadas por los desarrolladores en la sección de seguridad de datos.
2. Su diseño modular, que facilita la extensión del modelo hacia nuevos ámbitos, como el análisis de permisos, la estimación de métricas de riesgo o la evaluación de la transparencia algorítmica.

De este modo, la propuesta se sitúa en la intersección entre la ingeniería ontológica y la auditoría de privacidad en ecosistemas móviles, ofreciendo una base semántica sólida para el análisis y la comparación de aplicaciones.

Los resultados obtenidos durante la validación muestran que DataSecOnt es permite identificar incoherencias, patrones de riesgo y relaciones complejas entre tipos de datos, finalidades y riesgos que no serían detectables mediante análisis manual.

En la sección II se revisa la utilización de ontologías en el área de la privacidad y la repercusión de la propuesta de Google, la *Seguridad de los Datos*. La sección III presenta la metodología seguida para el diseño de la ontología DataSecOnt. Las fases de implementación y validación son el objeto de las secciones IV y V respectivamente. Finalmente, las conclusiones y trabajo futuro se discuten en la sección VI.

II. Trabajos Relacionados

II-A. Ontologías para la privacidad

Desde que se inició la andadura en la regulación de los derechos relativos a la privacidad en el mundo digital se han sucedido los esfuerzos por conseguir ontologías que representen el conocimiento relativo al tratamiento de datos personales y su impacto en los derechos de los sujetos titulares de los datos objeto de tratamiento. En particular, la llegada del Reglamento General de Protección de Datos (RGPD) supuso el desencadenante de un conjunto de esfuerzos destacados. Una revisión extendida de las ontologías de privacidad se puede encontrar en [5] y [6]. Entre esos esfuerzos merece la pena resaltar ontologías como como *PPO (Privacy Preference Ontology)* [7] y *Data Privacy Vocabulary (DPV)* [8] . Esta ontología recoge un exhaustivo catálogo de categorías o tipos de datos personales, sujetos relacionados con estos tratamientos (responsables, encargados, otros) y actividades. Su base es el propio Reglamento, RGPD, de modo que su vocabulario y conceptos se corresponden con los utilizados en el propio texto normativo. Un vocabulario y conceptos más ricos que el que se encuentra en las *Privacy labels* de Google Play, y en muchas ocasiones difíciles de alinear [9].

No obstante, la principal fuente de inspiración para el desarrollo de DataSecOnt fue la ontología de permisos de Google Play del proyecto *App-PI* [10], creada por el Grupo de Investigación en Ingeniería de la Privacidad de la Universidad de Valladolid, centrada en la caracterización semántica de los permisos en aplicaciones de Google Play [11].

II-B. Declaraciones sobre el tratamiento de datos personales

El Reglamento General de Protección de Datos (RGPD) [12] y la Ley Orgánica 3/2018, LOPD-GDD [13] exigen que las políticas de privacidad informen adecuadamente sobre los datos personales que cada aplicación recoge, con qué fin y con quién los comparte. Es el conocido como deber de informar, recogido en los artículos 13 y 14 del RGPD, que responde al principio de 'licitud, transparencia y lealtad'. Más recientemente, Google ha incorporado en Google Play una sección denominada *Seguridad de los Datos* en la cual los desarrolladores informan a sus usuarios de los datos a los que acceden. Algunos investigadores la ven como una alternativa más legible a las políticas de privacidad [14]. No obstante, los estudios demuestran que los desarrolladores presentan bastantes desviaciones en sus prácticas respecto a lo que sería deseable para disponer de unas descripciones normalizadas, fácilmente comparables [15].

III. Metodología de Desarrollo Ontológico

DataSecOnt unifica las diversas fuentes de información definidas por Google [1], [2], [3], [4], como categorías de datos, recopilación y propósitos de uso, estructurándolas bajo un modelo semántico común que facilita la consulta, interoperabilidad y el análisis automatizado.

El desarrollo de DataSecOnt se llevó a cabo mediante una combinación metodológica de dos enfoques complementarios ampliamente reconocidos en el ámbito de la ingeniería ontológica:

- **Ontology Development 101** [16], que ofrece una guía práctica y progresiva para el diseño y estructuración conceptual de ontologías.
- **Ontology Requirements Specification Document (ORSD)** [17], que proporciona un marco formal para la especificación de los requisitos funcionales y no funcionales del modelo.

Esta aproximación dual permitió abordar el proyecto tanto desde una perspectiva conceptual, identificación, definición y jerarquización de los elementos del dominio, como desde una perspectiva documental y verificable, basada en la trazabilidad y validación de los requisitos.

Ontology Development 101 proporcionó la base estructural para organizar el proceso de desarrollo, mientras que ORSD se utilizó principalmente para formalizar los requisitos funcionales mediante *Competency Questions* (CQs) y para definir los requisitos no funcionales, como la interoperabilidad o la mantenibilidad del modelo.

III-A. Etapas del proceso (Ontology Development 101) [16]

El proceso metodológico se dividió en seis fases progresivas:

1. **Determinación del dominio y alcance:** se definió el dominio de estudio como los datos gestionados por aplicaciones móviles en Google Play [1], con el objetivo de clasificar los tipos de datos y sus relaciones con los distintos propósitos o tratamientos posibles.
2. **Identificación de fuentes de conocimiento:** se emplearon como base las descripciones oficiales de Google Play [1] y la documentación pública sobre su sistema de categorías y propósitos [3], que sirvieron como punto de partida para construir el vocabulario ontológico.
3. **Definición de términos clave y clases iniciales:** el vocabulario ontológico de DataSecOnt se construyó a partir de un conjunto de conceptos fundamentales que estructuran el dominio de la seguridad y el tratamiento de datos en aplicaciones móviles. Entre los conceptos principales se incluyen Aplicación, Tipo de Dato, Categoría de Dato, Propósito, Medida de Seguridad, Organización, Permiso, Instancia y Usuario.
4. **Formulación de *Competency Questions* (CQs):** se redactaron quince *Competency Questions* derivadas de los requisitos funcionales definidos en ORSD, destinadas a validar la cobertura semántica del modelo. Algunos ejemplos incluyen:
   - CQ1 : *¿Qué tipos de datos pueden recopilar las aplicaciones?*
   - CQ8 : *¿Cuáles son los distintos fines por los que se recopilan datos?*
   - CQ15 : *¿Qué aplicaciones utilizan cifrado en tránsito como medida de seguridad?*
5. **Implementación en OWL/RDF:** se utilizaron las herramientas Protégé [18] y GraphDB [19] para codificar las clases, propiedades y axiomas del modelo, asegurando la compatibilidad con estándares semánticos como OWL [20] y RDF [21], y con vocabularios externos existentes.
6. **Validación mediante SPARQL y revisión iterativa:** se ejecutaron consultas SPARQL [22] sobre instancias representativas, basadas en aplicaciones reales de Google Play con el fin de verificar la coherencia y completitud de las relaciones ontológicas. Este proceso permitió realizar ajustes sucesivos hasta alcanzar un modelo estable, lógico y reutilizable.

III-B. Especificación de requisitos (ORSD)

La metodología ORSD [17] se empleó para documentar formalmente los requisitos ontológicos de DataSecOnt, sirviendo como complemento al proceso de diseño definido en *Ontology Development 101* [16]. Su aplicación permitió estructurar los objetivos del proyecto en dos niveles:

- **Requisitos funcionales**, expresados en forma de *Competency Questions* (CQs), que definen las preguntas en lenguaje natural que la ontología debe poder responder mediante consultas SPARQL.
- **Requisitos no funcionales**, orientados a garantizar el rendimiento, la interoperabilidad, la mantenibilidad y la coherencia semántica del modelo.

A partir de estos principios, basándose en ORSD, se formalizó un conjunto de objetivos concretos:

1. R1. Clasificar los tipos de datos según su naturaleza y nivel de sensibilidad.
2. R2. Mapear cada tipo de dato con su propósito y tratamiento asociado.
3. R3. Relacionar datos y medidas de seguridad implementadas.
4. R4. Facilitar la interoperabilidad con vocabularios externos y sistemas de análisis semántico.

El uso combinado de ambos enfoques no fue casual, sino una decisión metodológica orientada a garantizar tanto la solidez conceptual del modelo como la correcta formalización de sus requisitos. *Ontology Development 101* [16] aporta una guía práctica centrada en la identificación, organización y estructuración de los conceptos del dominio, lo que resulta esencial en un contexto donde las definiciones de Google Play proceden de fuentes dispersas y heterogéneas. Sin embargo, este enfoque no proporciona mecanismos formales para documentar requisitos, validar la cobertura semántica o garantizar la trazabilidad durante el desarrollo. Por ello, se incorporó el marco ORSD [17], que permite especificar de forma exhaustiva los requisitos funcionales mediante *Competency Questions* y describir los requisitos no funcionales asociados a interoperabilidad, mantenibilidad y coherencia del modelo. La combinación de ambos enfoques permitió equilibrar la fase conceptual, centrada en la construcción del vocabulario y la jerarquía semántica, con una fase documental y verificable, asegurando que cada decisión de modelado estuviese justificada y vinculada a un requisito explícito.

IV. Implementación de DataSecOnt

La ontología se implementó íntegramente en OWL 2 (Web Ontology Language) [20] y se representó en formato RDF [21]/XML [23], lo que garantiza su compatibilidad con herramientas semánticas y sistemas de inferencia. El modelo ontológico incluye una amplia jerarquía de clases y relaciones, organizadas en tres niveles principales:

- **Tipos de datos (DataType):** estructura basada en la documentación de Google Play [1], que incluye categorías como *Ubicación*, *Contactos*, o *Archivos*. Es el eje central de clasificación semántica de los datos tratados por una aplicación. Se refiere al tipo de información que recogen o tratan las aplicaciones. Se alinea con las categorías oficiales de datos definidas en Google Play. Por ejemplo, la *Ubicación aproximada* se refiere a la ubicación física del dispositivo en un radio igual o superior a 3 km^2, mientras que la *Ubicación precisa* se refiere a una distancia igual o menor que 3 km^2. A partir de Android 12 los usuarios pueden gestionar la precisión de la ubicación mediante los permisos. La figura 2 recoge un extracto de la visualización en Protégé de la clase *Tipo de Dato* y sus subclases.

- **Finalidades (Purpose):** conjunto de categorías que describen el uso de los datos, como *Marketing*, *Análisis*, *Personalización*, *Seguridad*. De nuevo, su semántica se corresponde con la expresada en Google Play.

- **Actores y relaciones:** entidades como Desarrollador, Usuario o Terceros, conectadas mediante propiedades semánticas como *recopila*, *comparte*, o *requiere*. El gráfico de la figura 3 representa las relaciones entre diez clases de la ontología, cada una representada con un segmento de color diferente, los cuales están conectados por acordes que representan las relaciones. Por ejemplo, existe una relación entre la Ubicación Aproximada (segmento de color verde amarillo) y la PublicidadOMarketing (segmento de color rojo): la ubicación aproximada se utiliza para publicidad y marketing.

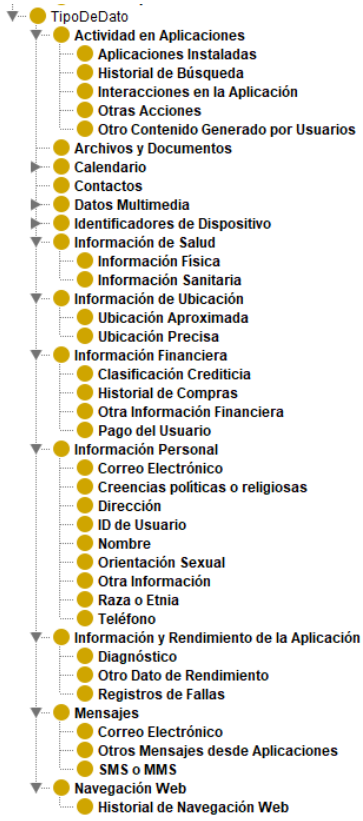


Figura 2: Tipos de Datos

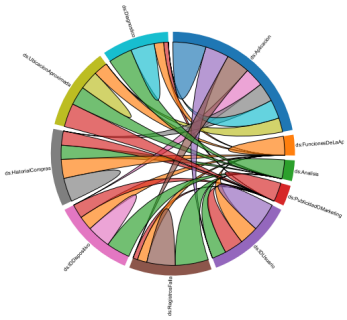


Figura 3: Gráfico de relaciones

La ontología se almacenó en un repositorio GraphDB [19], donde se llevaron a cabo diversas consultas SPARQL para comprobar su coherencia y validar la estructura del modelo. Un ejemplo de consulta se puede ver en la figura 4, que se corresponde con la expresión SPARQL de la pregunta de competencia CQ1.

```
SELECT DISTINCT ?tipo ?label
WHERE {
  ?tipo rdfs:subClassOf+ ds:TipoDeDato .
  OPTIONAL { ?tipo rdfs:label ?label . }
}
ORDER BY ?tipo
```

Figura 4: Pregunta de competencia CQ1 en SPARQL

V. Validación y Resultados

La validación de una ontología permite confirmar su utilidad. En el caso de **DataSecOnt** la validación se centró en comprobar la coherencia del modelo, su capacidad para responder a las *Competency Questions* (CQs) definidas en el documento ORSD y su comportamiento operativo dentro de un entorno funcional. Para ello, se ejecutaron consultas SPARQL y revisiones visuales iterativas sobre la ontología cargada en GraphDB [19] y Protégé [18], permitiendo verificar tanto las relaciones semánticas como la consistencia de los datos representados.

V-A. Validación mediante consultas

Las consultas se organizaron en dos bloques principales:

1. ***Competency Questions* (CQs)**: formuladas en las fases iniciales del proyecto como criterios de validación funcional del modelo.
2. **Consultas complementarias**: diseñadas para explorar en mayor detalle la información representada por la ontología y comprobar su capacidad de inferencia.

Los resultados obtenidos confirmaron que el modelo es consistente, completo y alineado con los objetivos de representación definidos. En el primer bloque, las quince CQs permitieron comprobar exhaustivamente el comportamiento de la ontología. Se validó, por ejemplo, la correcta jerarquía de tipos de datos y subtipos (CQs 1–3), la aplicación de medidas de seguridad y propiedades booleanas como *permiteEliminacionDatos* e *implementaMedidaSeguridad* (CQs 4 y 10), y la distinción entre datos personales y datos sensibles mediante el atributo *datoEsSensitivo* (CQs 5–6, 11–12).

Las consultas también demostraron la capacidad del modelo para representar relaciones complejas, como los casos de datos con múltiples propósitos de uso (CQ13), confirmando que la ontología soporta relaciones n-arias y escenarios en los que un mismo dato puede emplearse para distintas finalidades, por ejemplo, *personalización* y *publicidad*. Asimismo, la CQ7 validó la diferenciación entre recopilación y compartición de datos mediante la propiedad *modoUsoDato*, reflejando correctamente la trazabilidad de la información a lo largo de su ciclo de tratamiento.

El segundo bloque de consultas permitió explorar más a fondo los patrones de uso y medidas de seguridad de un conjunto de aplicaciones. Se identificaron, por ejemplo, las aplicaciones que comparten datos con fines de marketing (segmento de color rojo en la figura 3) y aquellas que recopilan la ubicación (segmento de color verde amarillo en la figura 3). También se puede saber qué aplicaciones permiten la eliminación de datos. Este último caso, relativo

a una aplicación dirigida al público infantil, evidenció la utilidad de la ontología para detectar inconsistencias éticas y de cumplimiento normativo, incluso en apps certificadas por la plataforma, Google Play.

V-B. Validación visual

Además de la validación automatizada mediante consultas, se llevaron a cabo validaciones visuales iterativas en Protégé [18] y GraphDB [19], revisando la estructura jerárquica, las relaciones entre clases y la inferencia de propiedades. La figura 5 muestra una visualización de alto nivel de la ontología, donde se reconocen las clases de primer nivel: *Aplicaciones*, *Entidades*, *Tipos de datos*, *Propósito*, *Medidas de seguridad*. Este proceso permitió ajustar las definiciones y detectar inconsistencias semánticas en etapas tempranas del desarrollo.

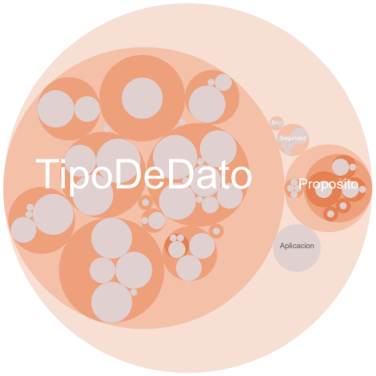


Figura 5: Validación con Gráficos

V-C. Prueba de concepto: Análisis de Aplicaciones Modeladas

Para ejemplificar la aplicación práctica de la ontología, se seleccionaron dos instancias representativas de aplicaciones reales disponibles en Google Play.

YouTube Kids (*com.google.android.apps.youtube.kids*) [24] está diseñada para un público infantil y declara la recopilación de un conjunto limitado de datos, como identificadores de usuario, ubicación aproximada, historial de búsqueda e interacciones, destinados principalmente al funcionamiento de la app, la personalización del contenido y el cumplimiento de las políticas de seguridad infantil [3]. Según los metadatos disponibles en Google Play, se ha podido caracterizar como una aplicación que incluye cifrado en tránsito y permite la eliminación de datos. Esto nos permite a su vez clasificarla dentro el conjunto de aplicaciones que aplican buenas prácticas de transparencia y protección de la privacidad de sus usuarios.

Twitch (*tv.twitch.android.app*) [25] es una plataforma de *streaming* y creación de contenido en directo. Recoge información más amplia, incluyendo identificadores, ubicación, interacciones, mensajes y contenido generado por el usuario. Se pudo comprobar que estos datos se emplean para personalización, análisis, publicidad dirigida y cumplimiento normativo. Esta aplicación sirvió para probar el potencial de **DataSecOnt** para representar escenarios de tratamiento de datos más complejos.

Los resultados confirmaron que DataSecOnt ofrece una representación semántica sólida y expresiva del dominio, capaz de identificar incoherencias entre la información declarada por los desarrolladores y los permisos realmente utilizados, así como de automatizar evaluaciones de riesgo y transparencia en el ecosistema de Google Play.

VI. Conclusiones

Se ha comprobado la utilidd de la ontología **DataSecOnt** para representar y analizar la seguridad y privacidad de los datos personales en el ecosistema de aplicaciones móviles. DataSecOnt constituye una base semántica estructurada que permite describir de forma precisa las prácticas de recopilación, compartición y protección de datos declaradas en Google Play [3]. Integra en un modelo formal las categorías oficiales de datos, sus finalidades y las medidas de seguridad asociadas, lo que permite superar las limitaciones inherentes a las descripciones textuales y heterogéneas proporcionadas por los desarrolladores.

Los resultados obtenidos durante la validación muestran que DataSecOnt no solo es capaz de responder adecuadamente a las *Competency Questions* planteadas, sino que también permite identificar incoherencias, patrones de riesgo y relaciones complejas entre tipos de datos, permisos y finalidades que no serían detectables mediante análisis manual. Esta capacidad de estructurar y razonar sobre la información declarada convierte a la ontología en una herramienta valiosa para auditores, investigadores y analistas interesados en mejorar la transparencia y la trazabilidad del tratamiento de datos en aplicaciones móviles.

Asimismo, las pruebas realizadas apuntan que DataSecOnt puede servir como base para desarrollar comparativas sistemáticas entre aplicaciones, evaluar el grado de transparencia ofrecido por los desarrolladores y detectar prácticas de tratamiento potencialmente problemáticas, especialmente en apps que manejan datos sensibles o dirigidas a menores. En conjunto, estas funcionalidades demuestran el potencial de la ontología para contribuir a ecosistemas más seguros, coherentes y responsables.

VII. Trabajo Futuro

A partir del trabajo desarrollado y de los resultados obtenidos, se identifican diversas oportunidades para ampliar y fortalecer el alcance de DataSecOnt.

- *Ampliación del conjunto de instancias y casos de uso.* Una línea prioritaria consiste en incorporar un conjunto más amplio y diverso de aplicaciones reales, incluyendo distintas categorías (salud, educación, finanzas, entretenimiento, apps dirigidas a menores). Esto permitiría analizar patrones sectoriales, tendencias de recopilación o diferencias en la transparencia entre tipos de aplicaciones.
- *Desarrollo de un agente inteligente capaz de interpretar consultas en lenguaje natural.* Una extensión relevante consiste en implementar una capa de razonamiento que traduzca preguntas formuladas por el usuario a consultas SPARQL sobre la ontología. Este agente permitiría, por ejemplo, preguntar "¿qué aplicaciones recopilan datos sensibles sin permitir su eliminación?" o "¿qué apps comparten datos con terceros con fines de publicidad?".

Este enfoque democratizaría el acceso al conocimiento representado en DataSecOnt, reduciendo la barrera técnica y habilitando su uso por auditores, analistas o incluso usuarios finales.

- *Integración con ontologías y vocabularios complementarios*. La modularidad de DataSecOnt abre la posibilidad de ampliarla mediante su integración con modelos como App-PI (permisos de Google Play), DPV (Data Privacy Vocabulary) o vocabularios específicos de seguridad y protección de datos. Esto permitiría facilitando análisis cruzados y evaluaciones de cumplimiento basadas en evidencia semántica.
- *Desarrollo de métricas y perfiles de riesgo semánticamente informados*. La ontología proporciona una base sólida para definir indicadores de riesgo basados en relaciones entre datos sensibles, permisos de alto impacto, actores externos y medidas de seguridad.
- *Implementación de herramientas de visualización y análisis basadas en la ontología*. Generar dashboards interactivos para mostrar patrones de uso de datos, mapas de sensibilidad, comparativas entre aplicciones, o tendencias evolutivas. Estas visualizaciones facilitarían la interpretación de resultados y apoyarían procesos de toma de decisiones, auditoría y evaluación de cumplimiento.

Estas líneas de evolución abrirían la puerta a la construcción de un ecosistema de conocimiento más completo y dinámico, capaz de contribuir a la auditoría semántica, a la evaluación automatizada del cumplimiento normativo y al análisis comparado de aplicaciones, fortaleciendo la transparencia y la confianza en el ámbito de las aplicaciones móviles.

Datos de investigación

La ontología *DataSecOnt*, las preguntas de competencia, y datos para las dos aplicaciones mencionadas en la sección V-C, *You Tube Kids* y *Twitch*, se encuentran disponibles en https://apppi.inf.uva.es/cloud/s/8W9kGntYC4BMBRF.

Agradecimientos

Este trabajo se incluye en las actividades del Proyecto Estratégico de Ciberseguridad App-PI (App Privacy Impact): Un ecosistema para la evaluación del impacto de apps para dispositivos móviles sobre la privacidad y seguridad de sus usuarios (C120.23_UVa), el cual se realiza al amparo de un convenio de colaboración entre la Universidad de Valladolid y la S.M.E. Instituto Nacional de Ciberseguridad de España M.P., S.A. para la promoción de proyectos estratégicos de Ciberseguridad en España, en el marco de los Fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation).

Referencias

[1] Google, "Ayuda de play console: Proporciona información para la sección de seguridad de los datos de google play," https://support.google.com/googleplay/android-developer/answer/10787469?hl=es-419, 2025, consultado en 2025.

[2] ——, "Ayuda de play console: Datos del usuario," https://support.google.com/googleplay/android-developer/answer/10144311?hl=es-419&ref_topic=9877467, 2025, consultado en 2025.

[3] ——, "Ayuda de google play: Comprende las prácticas de privacidad y seguridad de las apps con la sección de seguridad de los datos de google play," https://support.google.com/googleplay/answer/11416267?hl=es-419&co=GENIE.Platform%3DAndroid, 2025, consultado en 2025.

[4] ——, "Ayuda de google play: Usar google play protect para proteger tus aplicaciones y la privacidad de tus datos," https://support.google.com/googleplay/answer/2812853?hl=es, 2025, consultado en 2025.

[5] P. Ryan, H. J. Pandit, and R. Brennan, "A Common Semantic Model of the GDPR Register of Processing Activities," *CoRR*, vol. abs/2102.00980, 2021. [Online]. Available: https://arxiv.org/abs/2102.00980

[6] B. Esteves and V. Rodríguez-Doncel, "Analysis of ontologies and policy languages to represent information flows in GDPR," *Semantic Web*, vol. 1, 2022.

[7] S. Kirrane, U. Khokhar, A. O'Connor, and D. O'Sullivan, "Ppo: A privacy preference ontology for linked data," in *Proceedings of the 10th International Conference on Semantic Systems (SEMANTiCS)*. ACM, 2013, pp. 1–4. [Online]. Available: https://doi.org/10.1145/2506182.2506200

[8] H. J. Pandit, C. Debruyne, D. O'Sullivan, and D. Lewis, "The data privacy vocabulary (dpv)," *Semantic Web Journal*, vol. 13, no. 1, pp. 41–63, 2022. [Online]. Available: https://doi.org/10.3233/SW-210444

[9] Pablo Abel Criado-Lozano, M. Mercedes Martínez-González, A. Pérez-Fuente, and A. Aparicio, "El espejismo de la transparencia: Discrepancias en las declaraciones de privacidad en apps móviles," in *Actas de las X Jornadas Nacionales de Investigación en Ciberseguridad (JNIC 2025)*, ser. Jornadas Nacionales de Investigación en Ciberseguridad (JNIC), J. Carrillo-Mondéjar, J. G. Pérez, S. Pastrana, and R. J. Rodrígez, Eds. Universidad de Zaragoza, 2025, pp. 18–25.

[10] Grupo de Investigación en Ingeniería de la Privacidad de la Universidad de Valladolid, "App-PI Ontologies & SKOS Vocabularies for Android Applications," 2024, App-PI (App Privacy Impact) Research Project C120.23_UVa.

[11] Android Developers, "Android permissions overview," 2025, documentación oficial de Android. Último acceso: febrero 2025. [Online]. Available: https://developer.android.com/guide/topics/permissions/overview

[12] European Union, "General data protection regulation (gdpr)," *Official Journal of the European Union*, vol. L119, pp. 1–88, 2016.

[13] Jefatura del Estado, "Ley orgánica 3/2018, de 5 de diciembre, de protección de datos personales y garantía de los derechos digitales." *Boletín Oficial del Estado*, vol. 294, 2018, ELI: https://www.boe.es/eli/es/lo/2018/12/05/3/con.

[14] J. Krämer, "The death of privacy policies: How app stores shape GDPR compliance of apps," *Internet Policy Rev.*, vol. 13, no. 2, 2024. [Online]. Available: https://doi.org/10.14763/2024.2.1757

[15] R. Khandelwal, A. Nayak, P. Chung, and K. Fawaz, "Unpacking privacy labels: A measurement and developer perspective on google's data safety section," in *33rd USENIX Security Symposium, USENIX Security 2024, Philadelphia, PA, USA, August 14-16, 2024*, D. Balzarotti and W. Xu, Eds. USENIX Association, 2024. [Online]. Available: https://www.usenix.org/conference/usenixsecurity24/presentation/khandelwal

[16] N. F. Noy and D. L. McGuinness, "Ontology development 101: A guide to creating your first ontology," Stanford University, Tech. Rep., 2001.

[17] M. C. Suárez-Figueroa, A. Gómez-Pérez, and B. Villazón-Terrazas, "How to write and use the ontology requirements specification document," Ontology Engineering Group, Universidad Politécnica de Madrid, Tech. Rep., 2009.

[18] Stanford Center for Biomedical Informatics Research, "Protégé," 2024, Último acceso: febrero 2025. [Online]. Available: https://protegeproject.github.io/protege/

[19] Ontotext, "Graphdb documentation release 10.3.3," https://graphdb.ontotext.com/documentation/10.3/pdf/GraphDB.pdf, 2023, consultado en 2023.

[20] W3C, "Owl web ontology language reference," https://www.w3.org/TR/owl-ref/, W3C Recommendation, Tech. Rep., 2004.

[21] ——, "Best practice recipes for publishing rdf vocabularies," https://www.w3.org/TR/swbp-vocab-pub/, W3C Working Group Note, Tech. Rep., 2012.

[22] Ontotext, "Graphdb documentation: Sparql," https://graphdb.ontotext.com/documentation/10.3/sparql.html, 2025, consultado el 23 de enero de 2025.

[23] W3C, "Rdf 1.1 xml syntax," https://www.w3.org/TR/rdf-syntax-grammar, W3C Recommendation, Tech. Rep., 2014.

[24] Google LLC, "Youtube kids," 2025, aplicación en Google Play. Último acceso: febrero 2025. [Online]. Available: https://play.google.com/store/apps/details?id=com.google.android.apps.youtube.kids

[25] Twitch Interactive, Inc., "Twitch," 2025, aplicación en Google Play. Último acceso: febrero 2025. [Online]. Available: https://play.google.com/store/apps/details?id=tv.twitch.android.app

FHE en Detección de Anomalías: Aplicando Cifrado a la Evaluación de Modelos de IA

Julen Bernabé-Rodríguez
TECNALIA, Basque Research and Technology Alliance (BRTA)
Derio, España
julen.bernabe@tecnalia.com

María Carmen Palacios
TECNALIA, Basque Research and Technology Alliance (BRTA)
Derio, España
maricarmen.palacios@tecnalia.com

Marina Ruiz del Olmo
Universidad del País Vasco (EHU)
Leioa, España
ruizmarina1223@gmail.com

Maite Álvarez Piernavieja
TECNALIA, Basque Research and Technology Alliance (BRTA)
Derio, España
maite.alvarez@tecnalia.com

***Abstract*—Los cifrados homomórficos han supuesto una revolución en el ámbito del procesamiento del dato. Al permitir realizar operaciones sobre datos cifrados, es posible ceder éstos a terceros para su tratamiento, aunque no sean de confianza. Además, con el auge de técnicas de inteligencia artificial, el dato cada vez cobra más importancia. En este trabajo, se estudian las capacidades actuales de los cifrados homomórficos en la evaluación de algoritmos de inteligencia artificial aplicada a la detección de anomalías. Como resultado, se muestra la viabilidad de compartir logs cifrados, y poder aplicar sobre ellos máquinas de vector soporte e incluso redes neuronales que predigan si éstos son anómalos o no. Los resultados obtenidos y su consiguiente implementación son extensibles a otros casos de uso y aplicaciones. Por último, debido a su importancia en estos algoritmos, también se estudian nuevas técnicas que mejoren la eficiencia de la multiplicación de matrices cifradas.**

***Index Terms*—cifrado homomórfico, detección de anomalías, autoencoder, SVM**

Tipo de contribución: *Investigación original*

I. Introducción

Un *Security Operations Center* (SOC) es un centro de operaciones de seguridad formado por un equipo de personas, junto a diversas herramientas y software. Este centro tiene como objetivo monitorizar infraestructuras digitales, detectando, analizando y dando respuesta a los incidentes de seguridad en tiempo real. Dependiendo de la organización, el SOC puede ser creado internamente o subcontratado a un tercero. En la actualidad, se suele tender a lo segundo, debido al alto coste en términos de contratación y de gestión de SOCs.

Esta situación se acrecenta al hablar de SOCs en el ámbito industrial. Debido al ámbito especializado de cada empresa, y a los diferentes grados de digitalización que pueden encontrarse en ellas, es habitual que el personal interno de la misma no tenga conocimientos de seguridad suficientes como para asumir la creación de un SOC interno. Por ello, existen cada vez más empresas que ofrecen SOCs como servicio dentro del ámbito industrial. En este documento, se asume este contexto y, de ahora en adelante, se denotará como *empresa contratante* a aquella que subcontrata los servicios de un SOC externo.

Dentro del SOC, es muy habitual el uso de sistemas *Security Information and Event Management* (SIEM). Los SIEM son herramientas software que permiten centralizar la monitorización, detección y respuesta a incidentes. Estos sistemas están en continua evolución, ya que tienen que mantenerse actualizados con respecto a nuevas amenazas a detectar y responder. De hecho, debido al auge de la inteligencia artificial (IA), muchas soluciones modernas de SIEM incluyen algoritmos de detección automática de amenazas basados en IA. Así, el SIEM es capaz de aprender de los datos conocidos para poder detectar amenazas con mayor rapidez.

No obstante, existe una problemática más que evidente en el paradigma de los SOC industriales subcontratados con respecto a la detección de amenazas: la falta de privacidad. Por ello, suele acordarse contractualmente el objetivo del tratamiento de los datos enviados al SOC, asumiendo así que éste es de confianza plena. Sin embargo, el problema de base no desaparece: todos los posibles datos de vulnerabilidades llegan *en claro* al SIEM, siendo éste un punto crítico de la empresa contratante. Por ello, cada vez se hace más necesario proteger la privacidad de los datos al ser enviados y procesados por el SIEM.

Con este contexto, en este artículo se presenta una solución para la detección privada y automática de anomalías empleando cifrados homomórficos e inteligencia artificial. Partiendo de algoritmos de detección automática previamente entrenados, se propone un nuevo *framework* en el que tanto la empresa contratante como el SOC pueden seguir efectuando la monitorización, detección y gestión de incidentes sin que los datos de la empresa contratante se vean comprometidos. Este framework ha sido implementado y se presenta su viabilidad, así como sus limitaciones y trabajo futuro detectados.

II. El Algoritmo de Detección de Anomalías Presente en el SOC

La empresa AYESA, junto con TECNALIA, han desarrollado un algoritmo de detección automática de anomalías que, tras ser entrenado con datos y logs de los dispositivos de la empresa contratante, es capaz de clasificar si cada uno de ellos presenta una anomalía o no. Para tal fin, el algoritmo consta de dos modelos: un *autoencoder* (AE) y una *support vector machine* (SVM).

Un AE es un tipo de red neuronal especializado en una tarea de codificación y compresión de la información. En este caso, el AE toma como entrada las muestras de la empresa contratante para codificarlas en datos con una mayor explicabilidad y además etiquetarlas como anómalas o no. Posteriormente, el AE es capaz de decodificar dichos datos para obtener los valores originales. El modelo concreto consta

de 7 capas. El entrenamiento de este AE se ha realizado de forma no supervisada.

El segundo modelo es una SVM. Una SVM es un modelo de clasificación en el que el objetivo es encontrar unos vectores de soporte que definan un hiperplano que separe las dos clases a clasificar. Así, cada nueva entrada del algoritmo terminará a un lado u otro de dicho hiperplano, pudiendo así clasificarla. La SVM ha sido entrenada a partir de la salida del AE, es decir, a partir de los datos codificados y etiquetados, para clasificarlos como anómalos o benignos. Por tanto, en este caso, el entrenamiento de la SVM sí ha sido supervisado.

Con estos dos procesos, la empresa AYESA, con la colaboración de TECNALIA, está investigando en la adición de algoritmos basados en inteligencia artificial que puedan complementar sus SIEM para poder ofrecer SOCs más completos a sus clientes. Estos algoritmos pueden ser entrenados en un primer momento con datos de la empresa contratante. Este primer entrenamiento se realiza en claro, ya que es un proceso puntual que además utiliza muestras actuales y antiguas para ello. A partir de este momento, para la detección de anomalías en nuevas muestras, no es necesario reentrenar el modelo, sino únicamente aplicarlo para detectar nuevas anomalías en base a lo aprendido durante el entrenamiento.

III. Detección Privada de Anomalías empleando Cifrados Homomórficos

A. Breve Contexto sobre Cifrados Homomórficos y CKKS

La técnica clave para el desarrollo del framework de detección automática y privada de anomalías presentado en este documento son los cifrados homomórficos. Los cifrados homomórficos tienen su origen en 2009 cuando Craig Gentry construye el primero en [1]. Desde entonces, se fueron desarrollando otros cifrados homomórficos [2], [3], [4], [5]. De especial interés el cifrado CKKS [5], ya que soporta intrínsecamente operaciones sobre valores reales y no enteros, sin necesidad de hacer transformaciones sobre ellos. Es con este tipo de valores con los que trabajan el AE y la SVM de la Sección II. El cifrado CKKS, en su versión moderna, consta de 4 algoritmos:

- $\texttt{KeyGen}(context)$: permite generar las claves CKKS necesarias para el contexto dado. El contexto es un compendio de elementos que permiten establecer:
  - La seguridad del esquema CKKS utilizado.
  - La complejidad de los algoritmos que pueden ejecutarse sobre textos cifrados en dicho contexto.
- $\texttt{Encrypt}(\mathbf{m}, \texttt{pk})$: permite cifrar mensajes $\mathbf{m} = (m_1, \ldots, m_n) \in \mathbb{C}^n$ empleando una clave pública $\texttt{pk}$ generada mediante $\texttt{KeyGen}(context)$.
- $\texttt{Eval}(\texttt{op}, \mathbf{c})$: permite realizar la operación $\texttt{op}$ sobre un mensaje o mensajes cifrados $\mathbf{c}$. Las dos operaciones básicas son:
  - $\texttt{Eval}(\texttt{add}, \mathbf{c}_1, \mathbf{c}_2)$: calcula la suma $\mathbf{c}_1 + \mathbf{c}_2$ componente a componente.
  - $\texttt{Eval}(\texttt{mult}, \mathbf{c}_1, \mathbf{c}_2)$: calcula el producto $\mathbf{c}_1 \cdot \mathbf{c}_2$ componente a componente.

  En su versión moderna, existen otras operaciones más complejas como pueden ser
  - $\texttt{Eval}(\texttt{sum}, \mathbf{c})$: calcula el sumatorio de las componentes del mensaje encapsulado en $\mathbf{c}$.
  - $\texttt{Eval}(\texttt{merge}, \mathbf{c}_1, \ldots, \mathbf{c}_k)$: permite fusionar todas las primeras componentes de los mensajes encapsulados en $\mathbf{c}_1, \ldots, \mathbf{c}_k$ y crear un nuevo texto cifrado con todas ellas.
  - $\texttt{Eval}(\texttt{chebyshev}, f, \mathbf{c})$: permite calcular la aproximación de la función f componente a componente sobre $\mathbf{c}$ empleando aproximaciones polinómicas de Chebyshev [6].
  - $\texttt{Eval}(\texttt{bootstrap}, \mathbf{c})$: el *bootstrapping* [1] es una técnica que permite reducir el ruido de un texto cifrado a su estado inicial.
- $\texttt{Decrypt}(\mathbf{c}, \texttt{sk})$: permite descifrar textos cifrados $\mathbf{c}$ empleando una clave secreta $\texttt{sk}$ generada mediante $\texttt{KeyGen}(context)$, para obtener el mensaje subyacente.

En cuanto a implementaciones existentes, el esquema CKKS se puede encontrar en diferentes librerías de código abierto como LATTIGO [7], HELib [8], SEAL [9] y OpenFHE [10]. Esta última ha sido la elegida para la implementación, por soportar estas operaciones modernas y tener una comunidad activa manteniéndola. Es de especial interés la última, debido a su carácter ampliamente abierto y a la comunidad activa detrás de la misma.

B. Framework de Detección Privada y Automática de Anomalías

En este contexto, se introduce un *framework* de detección privada y automática de anomalías basado en el cifrado La arquitectura de este framework se representa gráficamente en la Figura 1. En esta arquitectura, se presuponen ya entrenados tanto el AE como la SVM, y por tanto únicamente se muestra el flujo de datos durante la inferencia de nuevos logs generados por los equipos de la empresa contratante.

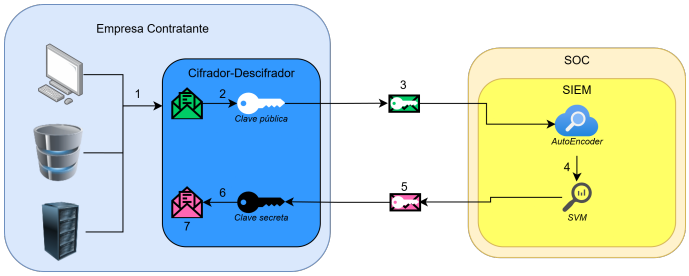


Fig. 1. Framework de Detección Privada y Automática de Anomalías.

Tal y como puede observarse en la Figura 1, este nuevo framework precisa de un nuevo componente Cifrador-Descifrador, así como de la adaptación de la funcionalidad del SIEM para poder ejecutar la detección automática sobre datos cifrados mediante Estas dos componentes son suficientes para poder realizar detecciones privadas y automáticas de anomalías siguiendo los siguientes pasos:

1) La empresa contratante recoge los datos y logs generados por todos sus dispositivos, los unifica y envía al componente Cifrador-Descifrador que ya tiene desplegado en sus instalaciones.
2) El componente Cifrador-Descifrador, habiendo generado previamente un par de claves CKKS,

utiliza la clave pública para cifrar los datos recibidos en el paso anterior.
3) El componente Cifrador-Descifrador envía al SOC los datos cifrados, así como otro material criptográfico necesario para operar sobre los mismos (claves de evaluación, contexto criptográfico...). En el SOC, los datos son enviados al SIEM para ser procesados.
4) Dentro del SIEM, se ejecutan de forma secuencial el AE y la SVM sobre cada uno de los textos cifrados recibidos. El resultado de esta ejecución, es decir, las predicciones, terminan cifradas, todas bajo la misma clave pública.
5) Las predicciones cifradas son enviadas de vuelta al componente Cifrador-Descifrador.
6) El Cifrador-descifrador usa su clave privada para descifrar los textos recibidos, obteniendo las predicciones de cada uno de los datos enviados en el paso 3.
7) El componente Cifrador-Descifrador interpreta los resultados obtenidos.

Nótese que la interpretación del paso 7 es sencilla. La clasificación de la SVM devuelve un -1 en caso de anomalía o un 1 en caso contrario, y la salida del AE contiene una columna con ese mismo valor. Por tanto, esta interpretación puede hacerse dentro del componente Cifrador-Descifrador de la empresa contratante.

IV. Implementación y Resultados

A la hora de implementar el framework descrito en la Sección III-B, se han desarrollado dos kits de desarrollo de software (SDK). El primero contiene toda la funcionalidad necesaria para implementar con facilidad el componente Cifrador-Descifrador que tendría que utilizar la empresa contratante. Con este software, es posible abstraerse de la generación de claves y material criptográfico de CKKS, cifrar muestras usando claves generadas previamente, e incluso descifrar e interpretar los resultados recibidos. Además, en vistas a posibles evoluciones de los algoritmos de detección de anomalías, es posible configurar de forma sencilla el contexto criptográfico a utilizar en función de los algoritmos existentes. Como resultado, el SDK facilita mucho el desarrollo de un componente Cifrador-Descifrador. Este SDK, no obstante, no debe asumirse como novedoso; es un compendio y abstracción de las técnicas ya existentes en la librería OpenFHE [10].

La principal contribución técnica y de desarrollo se encuentra en el SDK del SIEM, donde se han tenido que reescribir los algoritmos de inferencia del AE y la SVM para poder ejecutarlos sobre datos cifrados. A continuación se describen los problemas encontrados durante este desarrollo y las soluciones propuestas para solventarlos. En primer lugar, se presenta el desarrollo realizado para la SVM, ya que su inferencia consiste en un algoritmo más sencillo y además permite introducir algunos problemas encontrados también en el AE. Sea $\mathbf{X} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ la muestra de datos a predecir, la SVM consiste en un par $(\mathbf{W}, b)$, formado por los pesos $\mathbf{W} = (w_1, \ldots, w_n) \in \mathbb{R}^n$ y el *bias* $b \in \mathbb{R}$. Con este par $(\mathbf{W}, b)$ y $\mathbf{X}$, el algoritmo de inferencia de la SVM puede reducirse a:

$$\texttt{Sign}(\langle \mathbf{X}, \mathbf{W} \rangle + b). \tag{1}$$

En la Ecuación (1), `Sign` es la función signo, mientras que la notación $\langle \cdot, \cdot \rangle$ corresponde al producto escalar en $\mathbb{R}^n$.

Empleando las operaciones de CKKS ($cf.$ Sección III-A), el producto escalar puede operarse sobre datos cifrados empleando $\mathbf{P} = \texttt{Eval}(\texttt{mult}, \mathbf{X}, \mathbf{W})$ y $S = \texttt{Eval}(\texttt{sum}, \mathbf{P})$. Sobre este resultado, se suma b mediante $r = \texttt{Eval}(\texttt{add}, S, b)$, y finalmente se aplica la función signo empleando $\texttt{Eval}(\texttt{chebyshev}, \texttt{Sign}, r)$. Este último punto merece una especial mención, ya que los polinomios de Chebyshev se definen dentro de un intervalo $(\alpha, \beta) \in \mathbb{R}$. Al estar los datos cifrados de inicio, es necesario definir correctamente tanto α como β con el fin de que la aproximación se realice para valores dentro del intervalo. Por ello, se ha normalizado los datos en $\mathbf{X}$ al intervalo $[0, 1]$.

El algoritmo de inferencia del AE es considerablemente más complejo. El AE es una tripleta $(\mathcal{W}, \mathcal{B}, \mathcal{F})$ donde

- $\mathcal{W} = (\mathbf{W}_1, \ldots, \mathbf{W}_k)$ con $\mathbf{W}_i \in \mathbb{R}^{n_i \times m_i}$ para $i = 1 \ldots, k$,
- $\mathcal{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_k)$ con $\mathbf{b}_i \in \mathbb{R}^{m_i}$ para $i = 1, \ldots, k$,
- y $\mathcal{F} = (f_1, \ldots, f_k)$, donde cada f_i es una función de activación.

Es decir, cada una de las capas consiste en una tripleta $(\mathbf{W}_i, \mathbf{b}_i, f_i)$, donde $\mathbf{W}_i$ es la matriz de pesos, $\mathbf{b}_i$ es el vector de biases y f_i es la función de activación.

Con esta estructura, dada una muestra de datos $\mathbf{X} \in \mathbb{R}^n$, el algoritmo de inferencia del AE evalúa cada una de las capas sobre $\mathbf{X}$ de forma secuencial. Por tanto, sin pérdida de generalidad y con intención meramente explicativa, el proceso de evaluación de $\mathbf{X}$ sobre una sola capa $(\mathbf{W}, \mathbf{b}, f)$ con $\mathbf{W} \in \mathbb{R}^{n \times m}$ y $\mathbf{b} \in \mathbb{R}^m$, puede expresarse como sigue:

$$f(\mathbf{X}^T \cdot \mathbf{W} + \mathbf{b}) \tag{2}$$

Nótese que (2) es muy similar a (1), sólo que en vez de productos escalares se tienen productos matriciales, y que ahora la función de activación es diferente de la función signo.

En este caso, el producto matricial $\mathbf{X}^T \cdot \mathbf{W}$ puede expresarse como m productos escalares si se expresa $\mathbf{W}$ por columnas $\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_m)$:

$$\mathbf{X}^T \cdot \mathbf{W} = (\langle \mathbf{X}, \mathbf{w}_1 \rangle, \cdots, \langle \mathbf{X}, \mathbf{w}_m \rangle) = (\mathbf{P}_1, \ldots, \mathbf{P}_m).$$

Por tanto, es posible efectuar este producto matricial empleando la combinación de operaciones usada en la SVM, además de la función $\texttt{Eval}(\texttt{merge}, \mathbf{P}_1, \ldots, \mathbf{P}_k) = \mathbf{P}$ para fusionar todos los sumatorios en un solo texto cifrado. Finalmente, se suma $\mathbf{B}$ usando $r = \texttt{Eval}(\texttt{add}, \mathbf{P}, \mathbf{b})$ y se aplica $\mathbf{C} = \texttt{Eval}(\texttt{chebyshev}, f, r)$.

Es importante tener en cuenta que el proceso anterior ha de ser repetido para cada capa. En la práctica, esto supone un problema: tanto la función $\texttt{Eval}(\texttt{merge}, \mathbf{P}_1, \ldots, \mathbf{P}_k)$ como $\texttt{Eval}(\texttt{chebyshev}, f, r)$ son muy costosas. Esto supone que el contexto generado para soportar la inferencia del AE sea demasiado grande como para poder generar sus claves en la práctica. Por ello, se ha decidido emplear $\texttt{Eval}(\texttt{bootstrap}, \mathbf{C})$. Es decir, la evaluación final del AE consta de dos pasos, repetidos de forma secuencial por cada capa a evaluar: la ejecución de $\mathbf{C} = f(\mathbf{X}^T \cdot \mathbf{W} + \mathbf{b})$, y la reducción de ruido mediante $\texttt{Eval}(\texttt{bootstrap}, \mathbf{C})$.

Por tanto, ha sido posible adaptar ambos algoritmos de inferencia para encapsularlos dentro del SDK del SIEM.

Este SDK no sólo permite abstraerse de la ejecución de ambos algoritmos, sino que ha sido desarrollado pensando siempre en la naturaleza cambiante de estos modelos. Por tanto, es posible cambiar el número de capas del AE y sus dimensiones, o los hiperplanos y funciones de la SVM de forma sencilla. Además, se han seguido las estructuras de modelos de Pytorch, siendo más sencillo para un científico de datos aplicar estos cambios. Para el SOC, esta funcionalidad es vital, ya que su SIEM se abstrae del formato en que vengan los datos. Es decir, si la empresa contratante decide cifrar sus datos, su SIEM sólo tiene que usar las funcionalidades de este SDK sin cambiar radicalmente su implementación.

A. Resultados de la implementación

Ambos SDK han sido finalmente instanciados para realizar una prueba de concepto que permita ver las características y limitaciones de la ejecución de ambos algoritmos. La Tabla I resume algunos de los resultados más relevantes, como el tiempo de ejecución, los recursos empleados, los tamaños de textos y claves, etc., para cada uno de los algoritmos. Las métricas han sido tomadas en una máquina con procesador Core i7-12850HX con 8CPU y 32GB de RAM.

TABLE I
RESULTADOS DE LA PRUEBA DE CONCEPTO DE DETECCIÓN PRIVADA Y AUTOMÁTICA DE ANOMALÍAS.

	AE (sin seg.)	AE (128-bit)	SVM (128-bit)
Tiempo `KeyGen` (s)	4.40	161.74	2.51
RAM `KeyGen` (GB)	0.53	27.1	0.41
Tiempo `Encrypt` (s)	0.03	7.62	0.72
RAM `Encrypt` (GB)	0.07	4.3	0.08
Tiempo `Eval` (s)	15.66	180.45	20.60
RAM `Eval` (GB)	0.06	3.7	0.1
Tiempo `Decrypt` (s)	0.003	4.23	0.40
RAM `Decrypt` (GB)	0.0001	0.72	0.0001
Tamaño *secret key* (GB)	0.0003	0.012	0.003
Tamaño *public key* (GB)	0.0007	0.032	0.008
Tamaño *ciphertext* (GB)	0.002	5.7	0.08
Error Cifr. vs. Claro	10^{-3}	> 0.1	10^{-3}

Los resultados obtenidos son prometedores, puesto que muestran que, con la configuración adecuada, es posible mantener la precisión de las ejecuciones en comparación con los resultados en claro. No obstante, la ejecución del AE requiere de unos contextos que, aun habiendo sido reducidos, siguen generando claves muy pesadas cuando se añade seguridad. De hecho, para la ejecución con seguridad, se ha reducido la precisión a prácticamente el mínimo con la intención de poder ejecutar el algoritmo. La SVM no presenta esta problemática.

V. NUEVAS TÉCNICAS DE MULTIPLICACIÓN DE MATRICES

Los dos pasos más limitantes de cara a la evaluación del AE son la evaluación de polinomios de Chebyshev y la multiplicación de matrices. La primera de ellas añade muchos productos, obligando al algoritmo a utilizar *bootstrapping* para lograr un mínimo de eficiencia. El uso de *bootstrapping* es lo que dispara la generación de claves con seguridad (ver filas 1-2 en Tabla I). No obstante, la única forma de mejorar su eficiencia es buscando polinomios de menor grado que aproximen con gran precisión las funciones no lineales originales. En la práctica, esto normalmente reduce la precisión final del algoritmo, arrastrando errores muy grandes en la evaluación del AE.

TABLE II
PARÁMETROS NECESARIOS PARA LOS DISTINTOS MÉTODOS DE MULTIPLICACIÓN DE MATRICES CIFRADAS $A_{n\times m} \cdot B_{m\times p}$. POR COLUMNAS, SE CONSIDERAN EL NÚMERO DE CIPHERTEXTS, PRODUCTOS Y ROTACIONES, Y LA PROFUNDIDAD MULTIPLICATIVA NECESARIA. EL VALOR DE d PUEDE VARIAR EN FUNCIÓN DEL MÉTODO: * $d = \max(n, m)$, † $d = \max(n, m, p)$, § (n, m, p) COPRIMOS 2 A 2 CON $\max(n, p) < m$, ‡ $d = \max(m, p)$, Y ** (n, m, p) COPRIMOS 2 A 2 CON $m = 2^k$.

Método	# CT	# Mult.	# Rot.	Prof.
Naïve	(n, m)	mn	mn	3
[11], [12]	(n, m)	mn	$mn \log p$	2
[21]†	$(1, 1)$	$5d$	$3d + 5d^{\frac{1}{2}}$	3
[13]†	$(1, 1)$	$2d$	$2d + 3\log d - 2$	2
[20]†	$(\frac{d}{s}, \frac{d}{s})$	$2\frac{d}{s}$	$2\frac{d}{s}(1 + \log d)$ $+3\log s - 2$	2
[14]†	$(1, 1)$	d	$d \log d + d$	2
[15]‡	$(1, 1)$	m	$m \log d + m$	2
[16]*	$(1, 1)$	$2d$	$4d$	2
[17] Comp†	$(1, 1)$	$O(1)$	$O(\log d)$	2
[17] Large†	$(1, 1)$	$O\left(d^{2-\frac{2}{3\rho}}\right)$	$O\left(d^{2-\frac{2}{3\rho}}\right)$	2
[18] BMM-I§	$(1, 1)$	m	$2m + 2$	1
[18] BMM-II§,**	$(1, 1)$	1	$3\log d$	1

En consecuencia, sólo queda tratar de ganar eficiencia durante el producto $\mathbf{X}^T \cdot \mathbf{W}$. Este producto afecta principalmente al tiempo de evaluación, no tanto a la generación de claves (ver filas 5-6 en Tabla I). En este campo, sí existen más alternativas que proponen empaquetados eficientes de matrices dentro de los textos cifrados [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], cada uno imponiendo sus requisitos. Se resumen estos requisitos para los métodos con implementación existente en la Tabla II.

En esta Tabla, la primera fila muestra los requisitos impuestos por el método *naïve*, que es el implementado dentro del SDK. Como se puede observar, este método requiere de muchos ciphertexts (n para la primera matriz y m para la segunda), así como un número de multiplicaciones y rotaciones cuadráticos en función del número de filas y columnas. Aunque la profundidad multiplicativa sea similar a las demás, es aditiva con respecto al número de multiplicaciones matriciales a realizar: para ℓ multiplicaciones se requiere una profundidad multiplicativa de 3ℓ. Además, puede deducirse de la Tabla II que es posible reducir considerablemente los demás elementos. Por tanto, sí merece la pena pensar en este tipo de métodos para ganar eficiencia al multiplicar matrices.

Sin embargo, no todos estos métodos pueden aplicarse en la práctica. En efecto, sólo algunos de los métodos han sido implementados para el esquema CKKS [13], [18], [20], [21]. Los demás presentan implementaciones para BGV, y por tanto no son de interés para este trabajo. Por otro lado, sólo dos de los restantes han sido implementados en OpenFHE [20], [21].

Veamos, por tanto, cómo se compara el método *naïve* con estos dos últimos. La máquina utilizada vuelve a ser la misma que la empleada para tomar métricas del AE y la SVM: una máquina con procesador Core i7-12850HX con 8CPU y 32GB de RAM. Los productos de matrices efectuados son para matrices cuadradas $d \times d$, con $d = 2, 4, 8, 16, 32, 64, 128$. Se

han realizado productos de 2 y 3 matrices, tomando tiempos de ejecución y uso de memoria en todos los casos para los tres métodos:

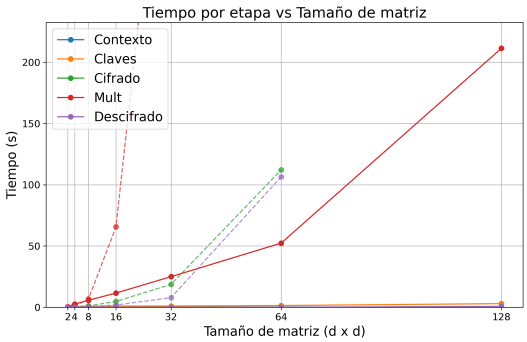


Fig. 2. Método Jiang *et al.*: Tiempos para multiplicar dos matrices.

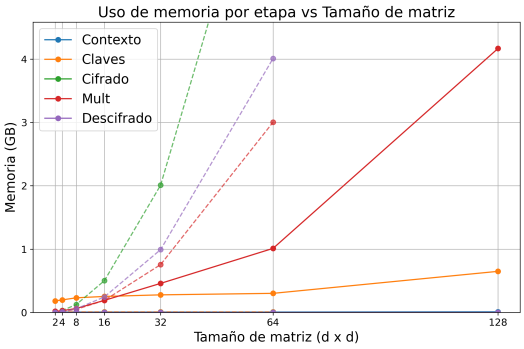


Fig. 3. Método Jiang *et al.*: RAM para multiplicar dos matrices.

- *Naïve*.
- El método propuesto por Jiang *et al.* en [21].
- El método propuesto por Aikata & Roy en [20].

En cada caso, se han comparado dentro de la misma gráfica los resultados para el método *naïve* (líneas discontinuas) frente a los nuevos métodos (líneas continuas).

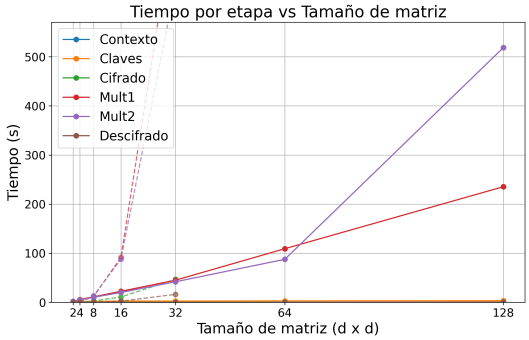


Fig. 4. Método Jiang *et al.*: Tiempos para multiplicar tres matrices.

En el caso del método propuesto por Jiang *et al.* en [21], se observa una gran mejoría frente al método *naïve* con matrices de tamaños 32×32 y 16×16 en adelante (resp. para 2 y 3 matrices). Esta mejoría se va haciendo cada vez más latente en cuanto aumentan las dimensiones de las matrices, demostrando la poca eficiencia del método *naïve* en estas situaciones. Teniendo en cuenta que los algoritmos de IA

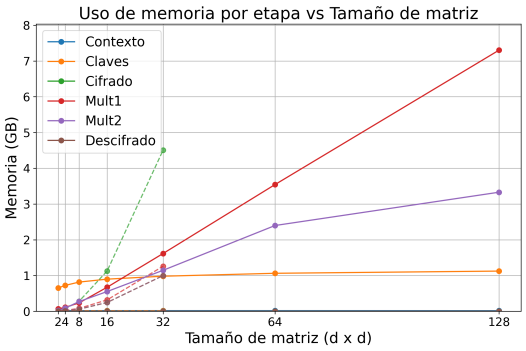


Fig. 5. Método Jiang *et al.*: RAM para multiplicar tres matrices.

suelen implementar matrices de altas dimensionalidades, la ganancia con respecto al método *naïve* es más que evidente.



Fig. 6. Método Aikata & Roy: Tiempos para multiplicar dos matrices.

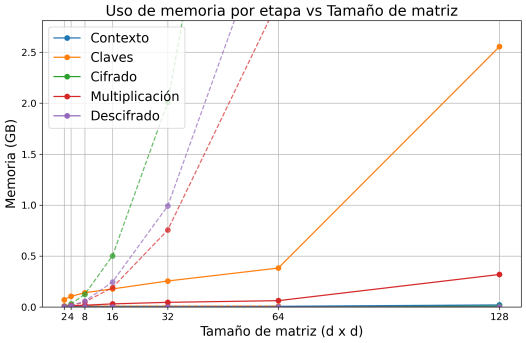


Fig. 7. Método Aikata & Roy: RAM para multiplicar dos matrices.

El método de Aikata & Roy es aún más eficiente que el propuesto por Jiang *et al.*, tanto en términos de tiempo como de memoria. Reduce a la mitad el tiempo de ejecución de la multiplicación de 3 matrices 128×128 con respecto al primero, y en 5 veces la generación del contexto. En este momento, y sin tener otras implementaciones disponibles para OpenFHE, parece el método *de facto* a utilizar para multiplicación de matrices cifradas.

VI. Conclusiones

Los cifrados homomórficos resultan ser una herramienta muy interesante para hacer uso de la IA a la hora de realizar detecciones privadas y automáticas de anomalías. Este documento presenta un framework de detección automática y

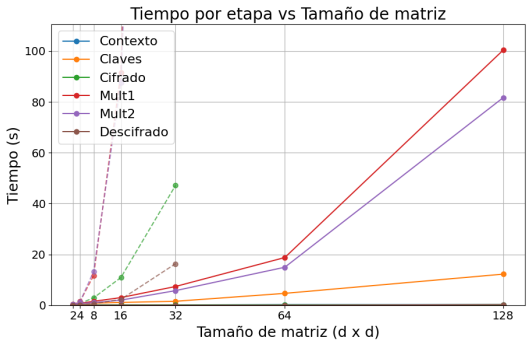


Fig. 8. Método Aikata & Roy: Tiempos para multiplicar tres matrices.

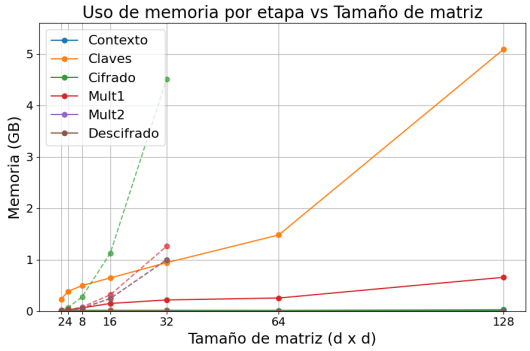


Fig. 9. Método Aikata & Roy: RAM para multiplicar tres matrices.

privada de anomalías basado en IA y el esquema homomórfico CKKS, que permite extender la funcionalidad de los SIEM actuales para soportar este tipo de detecciones. Este framework ha sido implementado en dos SDK diferentes, uno con la funcionalidad del componente Cifrador-Descifrador, y otro con los contenidos del SIEM. En este último, se muestra que ya es posible implementar algoritmos de IA usando cifrados homomórficos. Este SDK ha sido diseñado de forma flexible con el objetivo de poder adaptarse a nuevos modelos.

Ambos SDK han sido probados en una prueba de concepto, validando su viabilidad para realizar la detección privada y automática de anomalías en la práctica. Como resultado, se siguen necesitando muchos requisitos computacionales a la hora de generar y utilizar claves CKKS que permitan ejecutar este tipo de algoritmos. No obstante, se han buscado nuevas alternativas para ganar eficiencia en la multiplicación de matrices que estos algoritmos requieren, con resultados significativos. Además, la sola posibilidad de poder ejecutarlo, aunque los recursos necesarios sean muy altos, indica que la tecnología ya está preparada para afrontar este tipo de problemáticas. Teniendo en cuenta la comunidad ya existente trabajando en aceleración hardware de cifrados homomórficos y los resultados de la misma [22], [23], es previsible que estos resultados sigan mejorando en eficiencia los próximos años.

Esta investigación ha permitido a AYESA añadir una nueva funcionalidad a sus SOC en la que las empresas contratantes envíen sus datos cifrados al SIEM para la detección de anomalías. En este nuevo paradigma, el SIEM no puede aprender nada sobre las muestras de las empresas contratantes, pero puede seguir detectando sus anomalías.

Agradecimientos

Este trabajo se ha realizado dentro de la colaboración con Integrated Technology Systems SL e Ibermática SA en el proyecto NCIS (Nº MIG-20221060 – EXP: 00154859), cofinanciado por el CDTI y apoyado por el Ministerio de Ciencia e Innovación, a través del Programa de ayudas MISIONES y por la Unión Europea (NextGeneration EU).

References

[1] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the forty-first annual ACM symposium on Theory of computing*, 2009, pp. 169–178.

[2] Z. Brakerski and et al., "(Leveled) fully homomorphic encryption without bootstrapping," *ACM Transactions on Computation Theory (TOCT)*, vol. 6, no. 3, pp. 1–36, 2014.

[3] J. Fan and F. Vercauteren, "Somewhat Practical Fully Homomorphic Encryption," Cryptology ePrint Archive, Paper 2012/144, 2012. [Online]. Available: https://eprint.iacr.org/2012/144

[4] I. Chillotti and et al., "TFHE: Fast Fully Homomorphic Encryption Over the Torus," *Journal of Cryptology*, vol. 33, no. 1, pp. 34–91, 1 2020.

[5] J. H. Cheon and et al., "Homomorphic encryption for arithmetic of approximate numbers," in *Advances in Cryptology–ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I 23*, 2017, pp. 409–437.

[6] P. L. Chebyshev, *Théorie des mécanismes connus sous le nom de parallélogrammes*. Imprimerie de l'Académie impériale des sciences, 1853.

[7] C. V. Mouchet and et al., "Lattigo: A multiparty homomorphic encryption library in go," in *Proceedings of the 8th Workshop on Encrypted Computing and Applied Homomorphic Cryptography*, 2020, pp. 64–70.

[8] S. Halevi and V. Shoup, "Algorithms in helib," in *Advances in Cryptology–CRYPTO 2014: 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I 34*. Springer, 2014, pp. 554–571.

[9] "Microsoft SEAL (release 4.1)," https://github.com/Microsoft/SEAL, Jan. 2023, microsoft Research, Redmond, WA.

[10] A. Al Badawi and et al., "Openfhe: Open-source fully homomorphic encryption library," in *Proceedings of the 10th Workshop on Encrypted Computing & Applied Homomorphic Cryptography*, 2022, pp. 53–63.

[11] W.-j. Lu, S. Kawasaki, and J. Sakuma, "Using fully homomorphic encryption for statistical analysis of categorical, ordinal and numerical data," *Cryptology ePrint Archive*, 2016.

[12] S. Wang and H. Huang, "Secure outsourced computation of multiple matrix multiplication based on fully homomorphic encryption." *KSII Transactions on Internet & Information Systems*, vol. 13, no. 11, 2019.

[13] P. Rizomiliotis and A. Triakosia, "On matrix multiplication with homomorphic encryption," in *Proceedings of the 2022 on Cloud Computing Security Workshop*, 2022, pp. 53–61.

[14] D. Rathee, P. K. Mishra, and M. Yasuda, "Faster pca and linear regression through hypercubes in helib," in *Proceedings of the 2018 Workshop on Privacy in the Electronic Society*, 2018, pp. 42–53.

[15] H. Huang and H. Zong, "Secure matrix multiplication based on fully homomorphic encryption," *The Journal of Supercomputing*, vol. 79, no. 5, pp. 5064–5085, 2023.

[16] L. Zhu, Q.-s. Hua, Y. Chen, and H. Jin, "Secure outsourced matrix multiplication with fully homomorphic encryption," in *European Symposium on Research in Computer Security*. Springer, 2023, pp. 249–269.

[17] X. Zheng, H. Li, and D. Wang, "A new framework for fast homomorphic matrix multiplication (2023)."

[18] J. Chen, L. Yang, W. Wu, Y. Liu, and Y. Feng, "Homomorphic matrix operations under bicyclic encoding," *IEEE Transactions on Information Forensics and Security*, 2024.

[19] J. H. Park, "Ciphertext-ciphertext matrix multiplication: Fast for large matrices," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2025, pp. 153–180.

[20] A. Aikata and S. S. Roy, "Secure and efficient outsourced matrix multiplication with homomorphic encryption," in *International Conference on Cryptology in India*. Springer, 2024, pp. 51–74.

[21] X. Jiang, M. Kim, K. Lauter, and Y. Song, "Secure outsourced matrix computation and application to neural networks," in *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, 2018, pp. 1209–1222.

[22] N. Samardzic and et al., "Craterlake: a hardware accelerator for efficient unbounded computation on encrypted data," in *Proceedings of the 49th Annual International Symposium on Computer Architecture*, 2022, pp. 173–187.

[23] Y. Yang and et al., "Poseidon: Practical homomorphic encryption accelerator," in *2023 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE, 2023, pp. 870–881.

Integridad y auditoría para DNS: auditabilidad a largo plazo para IoT y redes distribuidas

Daniel Diaz-Sanchez, Florina Almenares Mendoza , and Carlos García-Rubio
Universidad Carlos III de Madrid
Leganés, 28911, Spain
Email: {dds, florina, cgr}@it.uc3m.es

***Resumen*—Este es un resumen extendido del artículo publicado [1]. DNSSEC tradicional presenta desafíos en entornos dinámicos debido al gran tamaño de las respuestas autenticadas y la falta de auditabilidad a largo plazo. La validación de la firma requiere transmitir todos los Registros de Recursos (RRs), causando ineficiencia y complejidad. Proponemos APEX (*Authenticated Proof of Existence*), un nuevo registro DNS que mejora DNSSEC. APEX utiliza Árboles de Merkle (*VDS*) para generar pruebas criptográficas concisas de RRs individuales. Este método reduce drásticamente el tamaño de las respuestas, optimiza la autenticación y permite auditoría a largo plazo. APEX ofrece una base escalable y segura para IoT y la adopción de la criptografía post-cuántica.**

***Index Terms*—DNSSEC, Merkle Tree, DNS Authentication, Auditability, IoT Security**

I. Introducción

DNSSEC tradicional presenta desafíos en entornos altamente dinámicos y geo-distribuidos. Los problemas principales incluyen el tamaño considerable de las respuestas autenticadas y la falta de auditabilidad a largo plazo. Aunque DNSSEC proporciona autenticación criptográfica, su método de validación requiere transmitir todos los Registros de Recursos (RRs), lo que genera transferencia de datos ineficiente y complejidad operacional. Estas ineficiencias son críticas en despliegues a gran escala como el Internet de las Cosas (IoT), y se exacerban con la transición a la criptografía post-cuántica debido a la mayor dimensión de sus firmas.

Para mitigar estas limitaciones, proponemos un nuevo enfoque que mejora las capacidades de DNSSEC sin alterar su mecanismo de autenticación, centrándose en la eficiencia y la integridad verificable. Nuestra solución introduce un nuevo tipo de registro DNS, Authenticated Proof of Existence (APEX), el cual utiliza Estructuras de Datos Verificables (VDS) —específicamente Árboles de Merkle— para ofrecer pruebas criptográficas cortas de RRs individuales. Este método reduce significativamente el tamaño de las respuestas y mejora la eficiencia de la autenticación. También habilita registros auditables a largo plazo mediante almacenamiento verificable por terceros.

Al extender el rol de DNS en la gestión de claves y la autenticación de servicios, especialmente en escenarios IoT a gran escala, APEX ofrece una base escalable, segura y auditable. Nuestros resultados de implementación demuestran que esta firma alternativa supera a DNSSEC, proporcionando una respuesta DNS más compacta incluso en el peor de los casos.

El resto del artículo se organiza de la siguiente manera: La Sección II revisa el estado del arte; la Sección III detalla el mecanismo de firma alternativo propuesto con Árboles de Merkle; la Sección III-D introduce el registro APEX y sus capacidades de verificación; la Sección IV describe el control de consulta y respuesta diferencial; la Sección V presenta los resultados de la implementación; y la Sección VI concluye el trabajo.

II. Estado del Arte

DNSSEC-bis [2] es el estándar del IETF que proporciona autenticación de origen, negación de existencia e integridad de datos para el DNS. DNSSEC no ofrece confidencialidad a menos que se combine con TLS o DTLS [3][4]. La versión actual (*-bis*) resuelve los problemas de escalabilidad iniciales mediante los registros *Delegation Signer* (DS), que introducen una indirección para mejorar la escalabilidad.

La firma DNSSEC se realiza adjuntando un registro RRSIG a los registros DNS. Este RRSIG contiene una firma generada con la *Zone Signing Key* (ZSK), certificada a su vez por la *Key Signing Key* (KSK). La cadena de confianza se establece a través de registros DS que apuntan a la KSK de la zona hija, creando una cadena de certificación alternativa a PKI.

Un RRSIG especifica el tipo de registro firmado (ej. `A`), el algoritmo de firma (ej. `5` para RSA/SHA1) y el campo labels, vital para validar *wildcards*.

DNS-based Authentication of Named Entities (DANE) [5] utiliza DNSSEC para autenticar asociaciones entre nombres de dominio y credenciales, abordando la vulnerabilidad de PKI ante CAs comprometidas. DANE permite a los servicios publicar credenciales de autenticación en sus zonas DNS a través del registro TLSA, indicando qué certificados o CAs esperar en una conexión TLS. Es relevante para la autenticación de servicios efímeros en entornos masivos de IoT.

Sin embargo, la firma DNSSEC es muy ineficiente en entornos con un alto número de registros a autenticar (cientos de miles o millones, como en IoT), ya que debe combinar registros por tipo y enviar toda la información al verificador para reconstruir el *hash* y validar la firma.

La transición a la criptografía post-cuántica (PQC) amplifica este problema, dado que las firmas PQC son generalmente de mayor tamaño. Trabajos como ARRF [6] y [7] abordan esto fragmentando la respuesta (RRFRAG) para evitar exceder el tamaño recomendado de ≈ 1232 bytes [8], pero no buscan reducir el tamaño de la respuesta, sino hacerla flexible para enviar bloques grandes en trozos pequeños.

Otras propuestas como [9] buscan la negociación de algoritmos criptográficos para DNSSEC, siendo nuestra propuesta agnóstica al mecanismo de firma y permitiendo que solo se firme la raíz. Otra alternativa [10] busca reducir el tamaño

de la respuesta poniendo las claves en la *blockchain*, siendo complementaria a nuestra solución.

Nuestra propuesta se centra en mejorar la eficiencia y seguridad de DNSSEC, reduciendo el tamaño de los mensajes al firmar solo la raíz del árbol en lugar de agrupar y firmar todos los registros. Además, ofrece flexibilidad para usar diversos tipos de firmas, respuestas diferenciales y minimiza los ataques de amplificación utilizando *hashes* resistentes a la computación post-cuántica.

III. Mecanismo de Firma Alternativo

Como se ha mencionado, DNSSEC agrupa registros por tipo y genera una firma para todos ellos utilizando una o más claves. Para la verificación, el cliente debe recibir todos los registros y la clave pública correspondiente. Por ejemplo, si se usan 4 claves para firmar 2000 registros TLSA en grupos de 500, el cliente recibe los 500 registros y la clave pública, calcula su *hash* y verifica la firma. Esto implica la necesidad de enviar los 500 registros completos para la verificación.

El uso masivo y dinámico de registros DNS no fue contemplado inicialmente por DNSSEC. El problema fundamental de enviar 500 registros es el tamaño requerido, el consumo energético y el tiempo de procesamiento. Una alternativa sería aumentar el número de claves para reducir el tamaño de los grupos. Sin embargo, esto introduce el problema de la custodia, transmisión y mantenimiento de un número creciente de claves. Por ejemplo, limitar el grupo a 20 registros requeriría gestionar y proteger 100 pares de claves públicas y privadas.

La transición a la criptografía post-cuántica impone resultados de mayor tamaño, a menudo proporcionales a la entrada. Es crucial abordar el problema del uso masivo de DNS para almacenar datos de autenticación o confianza, y la necesidad de manejar salidas más grandes. Por lo tanto, cualquier mejora destinada a disminuir el tamaño de la entrada para la firma en DNSSEC, y consecuentemente disminuir la transmisión de datos para verificar firmas, aumentará drásticamente el rendimiento tanto en el presente como con la adopción de la criptografía post-cuántica.

III-A. Propuesta de uso de Árboles de Merkle para DNSSEC

Las alternativas para reducir el tamaño de la entrada a la firma incluyen los Árboles de Merkle y los compromisos polinómicos KZG [11]. Los compromisos KZG requieren curvas con *pairing* y no son resistentes a la criptografía post-cuántica (PQC). En contraste, los compromisos basados en Merkle solo dependen de funciones *hash* (ej. SHA-3), heredando su resistencia a colisiones, y se consideran resistentes a ataques cuánticos (duplicando la longitud de salida del *hash* para mitigar el efecto Grover). Los Árboles de Merkle son de fácil adopción y resistentes a PQC.

III-B. Definición del Árbol de Hash

El árbol de Merkle elegido para DNS es binario por simplicidad y escalabilidad. Cada hoja almacena un registro DNS (nombre de dominio, clase, tipo, longitud y contenido).

Cualquier nodo (raíz, intermedio u hoja) almacena metadatos que indican su ubicación: nivel (siendo 0 la raíz), posición en el *hash* del padre [0—1] y un identificador único N-bits (`hid`). Estos metadatos no contribuyen al *hash*. La raíz almacena además la generación del árbol.

Cada RR añadido se coloca en un espacio específico y se recalculan los *hashes* hasta la raíz. Se mantiene un orden consistente al concatenar el *hash* del nodo en posición 0 con el *hash* del nodo en posición 1 para calcular el *hash* del padre.

```
{ meta: { hid: Nbits, level: num ,pos:[0|1] },
hash: hashval, RRData}
```

III-C. Inserción y Eliminación de Registros

El orden de los registros es indiferente en condiciones de igualdad, pero se pueden ordenar por Time-To-Live (TTL) para reducir la necesidad de recalcular *hashes* intermedios. Al eliminar un registro, puede ser reemplazado por uno nuevo o por un registro vacío.

El registro vacío es un registro DNS sin nombre ni tipo, que debe existir al menos uno por árbol, y se utiliza para probar la no existencia de un registro de forma certificada (NXDOMAIN).

El coste de inserción de un elemento i en un árbol binario es $O(\log i)$, requiriendo $O(\log n)$ *hashes* donde n es el número de hojas. El coste total de construcción del árbol con m hojas es $O(m \log m)$, en contraste con DNSSEC, cuyo coste de firma sobre m registros es $O(m)$ más la firma. La búsqueda en el árbol cuesta $O(\log n)$, y el tamaño de la prueba de verificación para m hojas es $O(\log m)$.

III-D. El Registro Authenticated Proof of Existence (APEX)

El registro APEX es una alternativa al RRSIG que proporciona autenticación de mensaje DNS equivalente, además de capacidad de auditoría.

A diferencia del RRSIG, que es único para un grupo de registros del mismo tipo, el registro APEX es único para cada registro DNS individual.

Contiene el identificador del nodo del árbol donde se ubica el registro, la ruta de verificación hasta la raíz y la firma de la raíz.

Para una respuesta DNS con un solo RR, el registro APEX contendría:

```
domain_name IN APEX TTL RRlen RRData
```

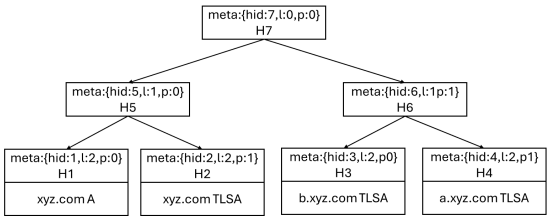


Figura 1. Ejemplo de árbol de Merkle DNS

Donde `RRlen` es la longitud y `RRData` el contenido. `RRData` es la codificación binaria del conjunto de metadatos y *hashes* que constituyen el camino del registro a la raíz. Para el ejemplo del nodo `xyz.com TLSA` en la Figura 1, el campo `RRData` sería:

```
xyz.com IN APEX TTL {TLSA
[{meta:{hid:2,l:2,p:1},hash:[]},
{meta:{hid:1,l:2,p:0},hash:H1},
{meta:{hid:6,l:1,p:1},hash:H6}]}
```

El verificador calcularía `H2` (el *hash* no se incluye) y usaría `H1` y `H6` para calcular la raíz.

Si la respuesta DNS contiene múltiples registros, la información se proporciona de forma diferencial para que cada

elemento identificador (*hash* y metadatos) se envíe una sola vez, minimizando la repetición. Al solicitar todos los registros TLSA del ejemplo, se proporcionarían los siguientes APEX:

```
xyz.com IN APEX TTL {TLSA
[{meta:{hid:2,l:2,p:1},hash:[]},
{meta:{hid:1,l:2,p:0},hash:H1}]}

b.xyz.com IN APEX TTL
{TLSA [{meta:{hid:3,l:2,p:0},hash:[]} ]}

a.xyz.com IN APEX TTL
{TLSA [{meta:{hid:4,l:2,p:1},hash:[]}]}
```

Con esta información, el cliente puede calcular los *hashes* de los nodos (`H2`, `H3`, `H4`) a partir de la información del registro solicitado, y luego los *hashes* intermedios (`H5`, `H6`) hasta llegar a la raíz `H7`. De esta manera, se evita enviar información repetida, maximizando la reducción del tamaño de la respuesta y evitando la amplificación de ataques.

III-E. Verificación y Auditoría de APEX

Un cliente que recibe la respuesta a una consulta (ejemplo TLSA) puede calcular los *hashes* hasta la raíz y verificar la firma del árbol.

El cliente o un observador puede realizar consultas periódicas y acumular evidencia de verificación, sirviendo como información probatoria. Este mecanismo de auditoría voluntaria permite a entidades independientes auditar el comportamiento del DNS y proporcionar evidencia criptográfica ante actos maliciosos o ilegítimos. Esto es especialmente relevante dado que el DNS se utiliza cada vez más para información de confianza (ej. DANE) y *bootstrap* de Encrypted Client Hello (ECHO) mediante registros SVCB [12].

Aunque este método es más eficiente que la firma DNSSEC original, puede optimizarse aún más mediante un mecanismo de solicitud diferencial.

IV. Mecanismo de Control de Consulta y Respuesta Diferencial

Minimizar la información transmitida es clave para la adopción de algoritmos post-cuánticos y para optimizar el consumo energético en entornos masivos.

Para evitar la transmisión de información innecesaria de forma segura, se utilizan cookies EDNS con ciertas modificaciones. Definimos la siguiente información clave en el servidor: gid (versión del árbol); trh (*hash* de la raíz); trt (tiempo de validez); tinf (*metadata* de la generación); RRset (registros solicitados); y RRhtid (conjunto de *metadata* de *hash* y opcionalmente valor: hashid, hashlevel, hashpos, hashvalflag e hashvalue).

El cliente puede enviar el parámetro `Sum_RRhtid`, que corresponde al conjunto de *metadata* de *hash* que tiene almacenado, sin incluir sus valores (hashvalflag en 0).

El servidor utiliza una clave secreta ksr y una función `f()` para generar la *cookie* del servidor (cs). El cliente tiene una función `fc()` para calcular su *cookie* con el prototipo `fc(tgid, cs, h(tinf|cs), Sum_RRhtid)`.

Con estos parámetros, se define un protocolo diferencial que utiliza la *cookie* EDNS para reducir el tamaño del mensaje. El cliente mantiene el estado, mientras que el servidor solo almacena una *cookie*. Los registros APEX se envían a través de EDNS.

Cliente Legítimo con Soporte de *Cookie* Diferencial: Este caso aplica a solicitudes recurrentes dentro del periodo de validez trt, sin que cambie la generación tgid. El Protocolo 1 detalla la primera interacción mientras que las solicitudes subsiguientes se definen en el Protocolo 2.

Solicitudes fuera de tiempo o clientes no compatibles: Si un cliente hace una solicitud fuera del período de validez (`trt`), el *hash* de la raíz puede haber cambiado. El servidor verifica la *cookie* `cs(n)` y la prueba de trabajo. Si la *cookie* es correcta, el servidor genera una respuesta completa `RES=[RRset,[],tinf] =[RRset,RRhtid,tgid', trh', trt]` y `cs(n)`, donde $\texttt{tgid}' \neq \texttt{tgid}$ y $\texttt{trh}' \neq \texttt{trh}$. El cliente puede seguir interactuando. Si el cliente no soporta *cookies* diferenciales o pruebas Merkle, el servidor generará una respuesta DNS estándar.

Protocol 1 Primera solicitud. Cualquier cliente

1. Solicitud 1
   - *a*) El cliente genera una *cookie* cc1 aleatoria y realiza la solicitud.
2. Respuesta 1
   - *a*) El servidor comprueba la *cookie*: cliente desconocido.
   - *b*) El servidor genera una respuesta (o parcial) para minimizar la amplificación.
   - *c*) El servidor genera $\texttt{cs1} = f(\text{ksr}, \text{RES}, [], \text{tinf}, \text{cc1})$ (se omite `RRhtid`).
   - *d*) El servidor entrega `RES=[RRset,[],tinf] =[RRset,[],tgid, trh, trt]` y `cs1` (si el cliente soporta APEX/Merkle por EDNS). De lo contrario, entrega solo RRset.
   - *e*) El cliente recibe la respuesta y almacena `tgid`, `trh`, `trt` y `cs1`.

Protocol 2 Otras solicitudes. Cliente con soporte de *cookies* EDNS y pruebas Merkle

1. Solicitud 2
   - *a*) El cliente genera la *cookie* cc2 $= fc$(tgid,cs1,h(tinf—cs1), `[]`), con `SUM_RRhtid` vacío, y envía la solicitud.
2. Respuesta 2
   - *a*) El servidor verifica la *cookie* `cc2` (con la prueba de trabajo h(tinf—cs1)) e identifica un cliente conocido recurrente.
   - *b*) El servidor genera la *cookie* $\texttt{cs2} = f(\text{ksr}, \text{RES}, \text{RRhtid}, \text{tinf}, \text{cc2})$.
   - *c*) El servidor entrega la respuesta completa `RES=[RRset,[],tinf]= [RRset,RRhtid,tgid, [], trt]` (se omite `trh`) y `cs2`.
   - *d*) El cliente recibe la respuesta y almacena `tgid`, `trh`, `trt`, `RRhtid` y `cs1`.
   - *e*) El cliente verifica la firma y las pruebas Merkle.
3. Solicitudes/Respuestas Subsiguientes
   - *a*) El cliente genera cc(n+1) $= fc$(tgid,cs1,h(tinf—cs(n)), `SUM_RRhtid`).
   - *b*) El cliente entrega la solicitud conteniendo `SUM_RRhtid` y cc(n+1).
   - *c*) El servidor genera una respuesta diferencial basada en `SUM_RRhtid`.

Cliente Malicioso: Si un cliente altera la *cookie* o `SUM_RRhtid`, el servidor verificará la *cookie* `cs(n)` y la prueba de trabajo `h(tinf|cs(n-1))`. Si la *cookie* es incorrecta, pero el cliente indica soporte para *cookies* diferenciales y pruebas Merkle, el servidor generará una respuesta truncada o parcial `RES=[RRset,[],tinf]=[RRset,[],tgid, trh, trt]` y `cs1` (se omite `RRhtid`). De esta manera, la única exposición del servidor es el cálculo de verificación de `cs(n)`.

V. Resultados

El método de firma propuesto usando Árboles de Merkle ha sido implementado y probado. Para las pruebas, se utilizó una sola *Zone Signing Key* (ZSK) con diferentes algoritmos. Las pruebas consideran el peor caso (sin solicitud o respuesta diferencial). Los resultados experimentales hasta 10000 registros considerando firmas RSA de 3072 bits, ECDSA-P256 y ED25519 muestran un crecimiento lineal llegando a aproximadamente 1MB para poblaciones de 10000 registros (Figura 2).

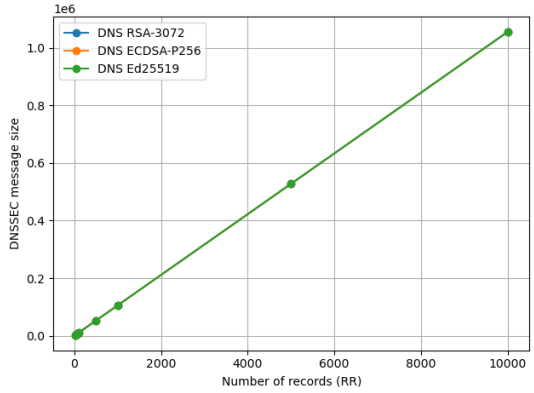


Figura 2. Tamaño del mensaje DNSSEC con clave única

Respecto al tamaño del mensaje DNSSEC al firmar usando Árboles de Merkle (Figura 3), se observa un tamaño mayor para la firma RSA 2048 (contribución de 256 bytes) comparado con las firmas ECDSA-P256 y ED25519 (64 bytes). Sin embargo, el tamaño total del mensaje es significativamente menor y no crece linealmente con el número de mensajes, a diferencia del DNSSEC tradicional.

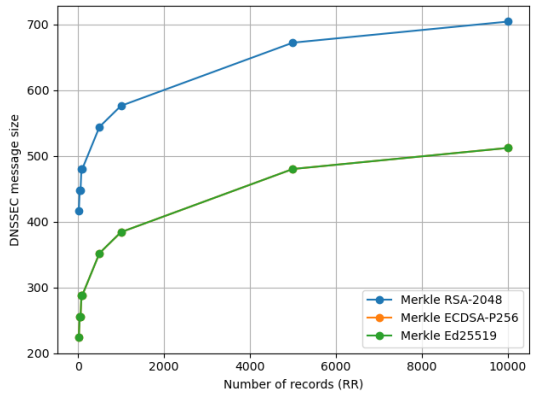


Figura 3. Tamaño del mensaje de la firma Merkle

VI. Conclusiones

Este artículo presenta un enfoque novedoso para mejorar DNSSEC mediante la introducción del registro Authenticated Proof of Existence (APEX), que utiliza Árboles de Merkle para proporcionar pruebas criptográficas compactas de registros DNS individuales. Nuestra solución reduce significativamente el tamaño de las respuestas y mejora la eficiencia de la autenticación, algo crucial para entornos dinámicos y geodistribuidos, especialmente en escenarios de IoT a gran escala. Además, este método habilita registros auditables a largo plazo de la información DNS a través de almacenamiento verificable por terceros, abordando la necesidad de hechos verificables. El mecanismo de control de consulta y respuesta diferencial propuesto optimiza aún más la transmisión de información, minimizando ataques de amplificación y soportando *hashes* resistentes a la criptografía post-cuántica. La implementación y las pruebas de nuestro método de firma basado en Árboles de Merkle demuestran una reducción sustancial en el tamaño del mensaje DNSSEC en comparación con los métodos tradicionales, particularmente a medida que aumenta el número de registros. Nuestro trabajo subraya el potencial para operaciones DNS más eficientes y seguras, especialmente en el contexto de tecnologías emergentes y la transición a la criptografía post-cuántica.

Agradecimientos

Esta investigación ha sido financiada por el proyecto I-SHAPER (INCIBE c114.23), por el programa de actividades de I+D TEC-2024/COM-504 (RAMONES-CM) de la Comunidad de Madrid, y por el Gobierno de España bajo el proyecto DISCOVERY (PID2023-148716OB-C33), financiado por MICIU/AEI/10.13039/501100011033.

Referencias

[1] D. Díaz-Sánchez, F. Almenares-Mendoza, C. García-Rubio, C. Campo-Vazquez, and J. Blanco-Romero, "Next-generation dns data integrity: Compact proofs and long-term auditability for iot and distributed cloud networks," in *Proceedings of the UCAmI 2025 Conference*, ser. Lecture Notes in Networks and Systems. Firenze, Italy: Springer, november 2025.

[2] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, "Resource Records for the DNS Security Extensions," RFC 4034 (Proposed Standard), RFC Editor, Fremont, CA, USA, Mar. 2005. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4034.txt

[3] Z. Hu, L. Zhu, J. Heidemann, A. Mankin, D. Wessels, and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)," RFC 7858 (Proposed Standard), RFC Editor, Fremont, CA, USA, May 2016, updated by RFC 8310. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7858.txt

[4] T. Reddy, D. Wing, and P. Patil, "DNS over Datagram Transport Layer Security (DTLS)," RFC 8094 (Experimental), RFC Editor, Fremont, CA, USA, Feb. 2017. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8094.txt

[5] P. Hoffman and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA," RFC 6698 (Proposed Standard), RFC Editor, Fremont, CA, USA, Aug. 2012. [Online]. Available: https://www.rfc-editor.org/rfc/rfc6698.txt

[6] J. R. Goertzen and D. Stebila, "Post-quantum signatures in dnssec via request-based fragmentation," pp. 535–564, 2022.

[7] A. S. Rawat and M. P. Jhanwar, "Post-quantum dnssec over udp via qname-based fragmentation," *IACR Cryptol. ePrint Arch.*, vol. 2024, p. 1320, 2023.

[8] L. Eggert, G. Fairhurst, and G. Shepherd, "UDP Usage Guidelines," RFC 8085 (Best Current Practice), RFC Editor, Fremont, CA, USA, Mar. 2017, updated by RFC 8899. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8085.txt

[9] A. Herzberg, H. Shulman, B. Crispo, "Less is more: cipher-suite negotiation for dnssec," *Proceedings of the 30th Annual Computer Security Applications Conference*, 2014.

[10] M. Raavi, S. Wuthier, and S.-Y. Chang, "Securing post-quantum dnssec against fragmentation mis-association threat," *ICC 2024 - IEEE International Conference on Communications*, pp. 97–102, 2024.

[11] A. Kate, G. M. Zaverucha, and I. Goldberg, "Constant-size commitments to polynomials and their applications," in *Advances in Cryptology – ASIACRYPT 2010*, ser. Lecture Notes in Computer Science, vol. 6477. Springer, 2010, pp. 177–194.

[12] B. M. Schwartz, M. Bishop, and E. Nygren, "Service binding and parameter specification via the dns (dns svcb and https rrs)," Internet Engineering Task Force, Internet-Draft draft-ietf-dnsop-svcb-https-12, Mar. 2023, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-dnsop-svcb-https-12

Un protocolo basado en blockchain para la entrega equitativa de mensajes con acuse de recibo

Gerardo-Francisco Chévez-Alvarado
Dpto. de Matemáticas e Informática
Universitat de les Illes Balears
Ctra. Valldemossa km 7.5
gf.chevez@uib.es

M. Francisca Hinarejos
Dpto. de Matemáticas e Informática
Universitat de les Illes Balears
Ctra. Valldemossa km 7.5
xisca.hinarejos@uib.es

Josep-Lluis Ferrer-Gomila
Dpto. de Matemáticas e Informática
Universitat de les Illes Balears
Ctra. Valldemossa km 7.5
jlferrer@uib.es

Andreu-Pere Isern-Deyà
Dpto. de Matemáticas e Informática
Universitat de les Illes Balears
Ctra. Valldemossa km 7.5
andreupere.isern@uib.es

***Resumen*—Este es un resumen extendido de la publicación [1], donde se presenta un protocolo basado en blockchain para la entrega segura de un documento con acuse de recibo. La entrega de mensajes o documentos con acuse de recibo es un servicio habitualmente utilizado por las administraciones públicas para notificaciones de sanciones o citaciones, y que tradicionalmente ha dependido de protocolos basados en terceras partes de confianza (TTPs). Actualmente, existen soluciones basadas en la tecnología blockchain que sustituyen parcial o totalmente a las TTPs. No obstante, estas soluciones presentan limitaciones para satisfacer plenamente los requisitos de seguridad exigibles, como la equidad o la confidencialidad. El protocolo presentado emplea un *smart contract* para automatizar la entrega y la generación de acuses de recibo verificables, sin depender del uso de TTPs. El análisis de seguridad demuestra que el protocolo garantiza la equidad y la confidencialidad del intercambio, ofreciendo así una solución segura y efectiva.**

***Index Terms*—notificaciones electrónicas, acuse de recibo, equidad, blockchain, smart contract**

I. Introducción

La entrega de mensajes con acuse de recibo es un servicio utilizado en múltiples ámbitos. En el caso de la administración pública, por ejemplo, se aplica en procesos como las citaciones o las notificaciones de sanciones. Sin embargo, dado que los avances tecnológicos están promoviendo la sustitución del papel por documentos electrónicos, resulta necesario adaptar este servicio al entorno digital. En este contexto, es importante garantizar condiciones equitativas para todos los participantes durante el proceso de entrega. En particular, en el caso del acuse de recibo que se aborda en este artículo, el requisito principal es la equidad: el emisor desea que el destinatario no pueda acceder al contenido del documento si no firma un acuse de recibo, mientras que el destinatario no quiere firmar dicho acuse de recibo sin tener la garantía de poder acceder al documento.

Tradicionalmente, se han utilizado terceras partes de confianza (TTPs) para resolver este problema [2], [3]. Sin embargo, la utilización de TTPs presenta diversas limitaciones, como la posibilidad de que sean deshonestas o de constituir un punto único de fallo. Para abordar este problema, la tecnología blockchain ha surgido como una posible alternativa a las TTPs. La blockchain es un registro descentralizado, asegurado mediante técnicas criptográficas, que almacena información en bloques. Cada bloque contiene un conjunto de transacciones protegidas criptográficamente y se distribuye a través de una red *peer-to-peer*. Estos bloques están encadenados entre sí, lo que garantiza la inmutabilidad de los datos: modificar un bloque requeriría alterar todos los bloques posteriores, impidiendo así la manipulación de la información almacenada en la cadena.

En el contexto de la tecnología blockchain, los *smart contracts*, o contratos inteligentes, permiten automatizar operaciones y acuerdos directamente sobre la blockchain. Un *smart contract* es un código ejecutable que se despliega y ejecuta en un entorno virtual (en el caso de Ethereum, la *Ethereum Virtual Machine* (EVM)), lo que posibilita la automatización de acuerdos y la creación de aplicaciones descentralizadas (DApps) [4]. Un *smart contract* utiliza la memoria de almacenamiento de la EVM para guardar la información necesaria para la ejecución de sus funciones. En este escenario, las transacciones generan recibos que incluyen detalles sobre el resultado de la transacción y los eventos registrados durante su ejecución [5].

El resto del artículo está estructurado de la siguiente manera. En la sección II se presenta el estado del arte. En la sección III se resume la solución propuesta. Finalmente, se exponen las conclusiones en la sección IV.

II. Estado del arte

El servicio de entrega de documentos con acuse de recibo ha recibido varias denominaciones, como correo electrónico certificado, notificaciones certificadas o entrega en línea. Con el objetivo de encontrar las soluciones existentes basadas en blockchain bajo estas denominaciones, se realizó una búsqueda en bases de datos como Scopus, Web of Science y Google Scholar. Tras analizar los resultados, se identificaron diversas propuestas relevantes, que se resumen a continuación.

En [6], los autores proponen dos soluciones para notificaciones certificadas en el escenario de un destinatario. Una de ellas no cumple con el requisito de confidencialidad y la otra requiere la intervención de una TTP. En [7], los mismos autores presentan dos esquemas para escenarios con varios

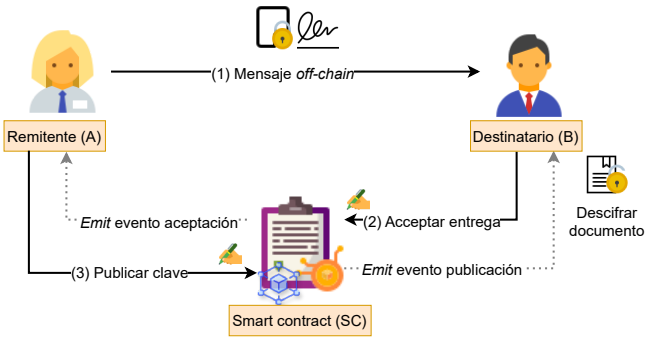


Figura 1. Escenario del protocolo.

destinatarios, en los que, nuevamente, uno de ellos no satisface la confidencialidad y el otro depende de una TTP. En [8], [9] se proponen soluciones para múltiples destinatarios que también requieren la participación de una TTP. En [10] se presenta una solución diseñada específicamente para escenarios con varios destinatarios, pero su especificación es incompleta: no se detalla cómo el destinatario verifica que ha recibido un mensaje del remitente. Además, la cancelación del proceso de entrega implica costes adicionales. En [11], se propone una mejora en términos de coste respecto a [10], pero los pasos de la entrega equitativa y la prueba generada son los mismos, y no se incluye un análisis de seguridad. En [12] se presenta una solución basada en la blockchain de Bitcoin, lo que dificulta su extensión a servicios más avanzados, como escenarios con múltiples destinatarios. En [13], se proponen varias soluciones también basadas en la blockchain de Bitcoin, algunas de ellas incompletas o incorrectamente especificadas. En una propuesta de [13] y en [14] se impone una fecha límite para su ejecución, lo que compromete la entrega equitativa. Todas las propuestas anteriores permiten el rechazo selectivo de mensajes por parte de los destinatarios basándose en la identidad del remitente. Las propuestas [15] y [16] abordan este problema mediante el anonimato parcial del remitente, aunque a costa de incorporar TTPs a la especificación.

En conclusión, el estado del arte sobre la entrega de documentos con acuse de recibo mediante blockchain presenta diversas deficiencias y limitaciones. Aunque se han propuesto múltiples enfoques para garantizar la equidad y la confidencialidad en el proceso de entrega, ninguno de ellos satisface plenamente todos los requisitos necesarios.

III. Resumen del protocolo propuesto

El protocolo de entrega de documentos con acuse de recibo (ver Figura 1), es un sistema que permite a un remitente (A) enviar un documento digital (M) a un destinatario (B) y obtener un acuse de recibo como evidencia de entrega. El protocolo consta de tres pasos y utiliza un *smart contract* (SC) para gestionar el proceso de entrega y resolver cualquier disputa que pueda surgir. La Tabla I ofrece un resumen de la notación utilizada en la descripción del protocolo. A continuación, se indican los pasos definidos por el protocolo:

1. *Mensaje off-chain*. El remitente (A) envía un mensaje cifrado al destinatario (B) mediante una clave simétrica, junto con una firma.
2. *Aceptar entrega*. El destinatario (B) puede aceptar o rechazar el mensaje. Si opta por aceptarlo, envía un reconocimiento de aceptación a la blockchain, que actúa como evidencia de que B está dispuesto a recibir el documento.
3. *Publicar clave*. El remitente (A) publica la clave de cifrado en una transacción *on-chain*, protegiéndola para que solo el destinatario (B) tenga acceso.

Tabla I
Notación para el protocolo de entrega equitativa de recibos.

A	Remitente
B	Destinatario
SC	Contrato inteligente
$@X$	Dirección de blockchain de X
M	Documento a entregar
$h()$	Función hash unidireccional resistente a colisiones
id	Identificador del proceso de entrega del documento
$Sig_X(a)$	Firma del elemento a realizada por la entidad X
K	Clave simétrica
$E_K(a)$	Función de cifrado simétrico sobre el elemento a utilizando la clave simétrica K
$D_K(a)$	Función de descifrado simétrico sobre el elemento a utilizando la clave simétrica K

III-A. Requisitos de seguridad

La solución debe cumplir con los siguientes requisitos de seguridad para proporcionar una entrega segura [6], [12]:

- *Efectividad*. El protocolo no requiere la participación de una TTP en ninguno de los pasos.
- *Equidad*. Una vez completada la ejecución del protocolo, o bien el destinatario recibe el documento y el remitente obtiene el acuse de recibo, o ninguna de las partes obtiene ninguno de ellos.
- *Timeliness*. Ambas partes pueden ejecutar el protocolo dentro de un tiempo finito, manteniendo el cumplimiento del requisito de equidad.
- *No repudio*. Ninguna de las partes puede negar su participación en la ejecución del protocolo: el destinatario no puede negar que aceptó la entrega y el remitente no puede negar que proporcionó la clave para acceder al documento.
- *Confidencialidad*. Únicamente el remitente y el destinatario tienen acceso al documento digital. Además, la información publicada en la blockchain no debe permitir el acceso al contenido a terceras partes.

III-B. Especificación del protocolo

La Figura 2 muestra los tres pasos definidos por el protocolo y permite visualizar de manera clara la interacción entre las partes y el flujo de información durante todo el proceso de entrega. Las líneas sólidas representan los mensajes enviados por el remitente o el destinatario, mientras que las líneas discontinuas indican los eventos generados por el *smart contract* tras la ejecución exitosa de una función. El código del *smart contract* fue desplegado en la red de pruebas Sepolia [17].

A continuación, se detallan los tres pasos del protocolo y se explican los parámetros involucrados en los mismos.

- **Paso 1: *Mensaje off-chain***. El remitente A envía a B un mensaje *off-chain* cifrado que contiene la siguiente información:

$$id, t_m, t_d, C, h, f(C)$$

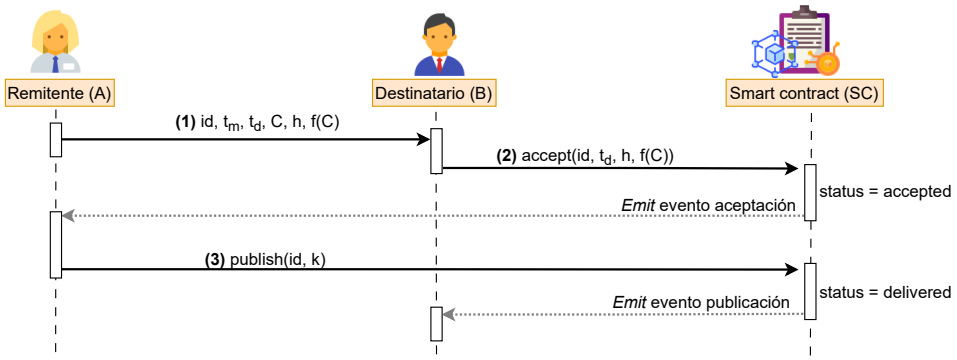


Figura 2. Especificación del protocolo.

donde,

- id: identificador único para la entrega, calculado como $id = H(@A, @B, @SC, t_m, t_d, h)$.
- t_m: *timestamp* de creación del mensaje.
- t_d: plazo límite para completar la entrega.
- C: mensaje M cifrado con una clave simétrica K, denotado como $C = E_K(M)$.
- h: hash de C, es decir, $h = H(C)$.
- $f(C)$: firma digital de A sobre la información anterior, calculada como $f(C) = Sig_A(id, t_m, t_d, h)$.

Con esta información, B puede decidir aceptar o rechazar el mensaje. Si no lo acepta, A observará que el plazo expira.

- **Paso 2: *Aceptar entrega***. Si B acepta recibir la entrega, genera una transacción *on-chain* invocando a la función de aceptación ($accept$) definida en el *smart contract*:

$$tx_{acceptB} = accept(id, t_d, h, f(C))$$

La función del *smart contract* comprueba que t_d no haya expirado y verifica el resto de la información proporcionada. Si la validación es satisfactoria, procede a modificar el estado de la entrega a $accepted$ y genera un evento que contiene la evidencia de aceptación ($tx_{acceptB}$).

- **Paso 3: *Publicar clave***. En este punto, B ha solicitado el documento, por lo que A debe proporcionar la clave K. Tras detectar el evento generado en el paso anterior, A publica la clave K mediante una transacción *on-chain* que invoca la función de publicación ($publish$) del *smart contract*:

$$tx_{publishA} = publish(id, k)$$

El parámetro k contiene el material criptográfico necesario para que solo B pueda recuperar la clave K (por ejemplo, K cifrada con la clave pública de B), la cual le permite obtener el documento original M descifrando C mediante la operación $M = D_K(C)$. El *smart contract* verifica la información proporcionada y valida que t_d no haya expirado. Si todo es correcto, procede a modificar el estado de la entrega a $delivered$ y genera un evento que contiene la evidencia de publicación ($tx_{publishA}$).

Al final de la ejecución exitosa del protocolo, los participantes disponen de las siguientes evidencias:

- A obtiene el acuse de recibo de B ($tx_{acceptB}$), el cual ha sido publicado en la blockchain como resultado de ejecutar la función de aceptación.
- A tiene una transacción de blockchain que confirma la publicación de la clave K, firmada con su clave privada y asociada a su dirección de blockchain ($@A$).
- B tiene C y k, lo que le permite acceder al contenido del documento. También tiene la firma digital de A ($f(C)$), obtenida en el Paso 1, así como la transacción generada por la función de publicación, firmada con la clave privada asociada a la dirección de blockchain de A ($@A$).
- Tanto A como B pueden consultar el estado de la entrega del documento proporcionando el identificador (id) al *smart contract*, que devuelve $undefined$, $accepted$ o $delivered$ según el estado del proceso. El significado de cada estado es el siguiente:
  - $undefined$: valor por defecto que devuelve el *smart contract* previo a la ejecución satisfactoria de la función de aceptación.
  - $accepted$: indica que el destinatario ha ejecutado correctamente la función $accept$ del contrato.
  - $delivered$: indica que el remitente ha publicado la clave K a través de la ejecución de la función $publish$ del contrato.

III-C. *Análisis de seguridad*

A continuación se evalúa el cumplimiento de los requisitos de seguridad presentados en el apartado III-A:

- *Efectividad*. El protocolo cumple el requisito, ya que no requiere del uso de una TTP durante ningún paso de su ejecución.
- *Equidad*. Para verificar este requisito es necesario analizar las diferentes situaciones que pueden ocurrir durante la ejecución del protocolo:
  1. *Situación 1*. Solo se ejecuta el Paso 1. El remitente no tiene el acuse de recibo y el destinatario no dispone del contenido del documento.
  2. *Situación 2*. Se ejecutan los Pasos 1 y 2. El destinatario no tiene acceso al contenido del documento y el remitente no dispone del acuse de recibo (únicamente tiene conocimiento del deseo del destinatario de recibir el documento).
  3. *Situación 3*. Se ejecutan todos los pasos del protocolo. Distinguimos dos casos posibles:
     - *Situación 3.1*. El destinatario tiene acceso al documento, ya que dispone del documento cifrado

(Paso 1) y la clave de descifrado (Paso 3). El remitente cuenta con el acuse de recibo, ya que dispone de la aceptación del destinatario (Paso 2) y de la prueba de publicación de la información necesaria para recuperar la clave (Paso 3).

- *Situación 3.2*. El destinatario tiene una clave que no le permite acceder al contenido del documento, pero dispone de la evidencia para probarlo y demostrar que no recibió el documento. Por su parte, el remitente tiene la aceptación del destinatario (Paso 2), pero la evidencia publicada en el Paso 3 muestra que fue deshonesto y, por lo tanto, no tiene evidencia de haber entregado el documento.

En conclusión, se satisface el requisito de la equidad.

- *Timeliness*. Se garantiza el cumplimiento del requisito, puesto que la ejecución del protocolo finaliza cuando expira t_d, momento a partir del cual no pueden ejecutarse los pasos 2 y 3, o bien cuando la entrega se completa antes de dicha expiración.
- *No repudio*. El remitente firma el documento cifrado ($f(C)$) y los datos necesarios para recuperar la clave de descifrado (k, registrada en la transacción). Esta firma impide que pueda negar haber enviado dicha información. Por su parte, el destinatario firma la aceptación para recibir el documento mediante la ejecución de la función $accept$, por lo que no puede negar haber aceptado recibirlo. Por consiguiente, se asegura el requisito.
- *Confidencialidad*. El remitente envía el documento cifrado con una clave secreta K ($C = E_K(M)$) y publica los datos para recuperar la clave secreta en la blockchain, de manera que sólo el destinatario pueda recuperarlo mediante su clave privada. En consecuencia, se garantiza el requisito.

IV. Conclusiones

Este trabajo presenta un protocolo seguro y equitativo basado en blockchain para la entrega de documentos digitales con acuse de recibo entre un remitente y un destinatario. Tras analizar la literatura sobre las soluciones basadas en la tecnología blockchain, se descubrió que ninguna de ellas satisface plenamente todos los requisitos necesarios. Por un lado, algunas propuestas no garantizan la equidad ni la confidencialidad del proceso de entrega debido a la forma como gestionan la expiración de los plazos. Por otro lado, y con el objetivo de superar esta limitación, otras soluciones recurren a la dependencia de TTPs, lo que a su vez introduce un punto único de fallo que puede comprometer la equidad del proceso de entrega. A diferencia de la literatura previa, el protocolo presentado satisface todos los requisitos necesarios de seguridad, especialmente en lo que respecta al cumplimiento conjunto de las propiedades de equidad en el intercambio y de confidencialidad del documento a entregar, evitando comportamientos maliciosos por parte de los participantes durante el proceso. Además, no depende del uso de TTPs en ningún paso, lo que elimina los potenciales problemas asociados a su utilización. Para conseguir estos objetivos, el protocolo se basa en el diseño de un *smart contract* compatible con blockchains EVM, que gestiona de forma autónoma el registro y generación de evidencias durante el proceso de entrega. El protocolo resulta aplicable en el contexto actual de progresiva digitalización de los procedimientos administrativos, como las citaciones, las notificaciones de sanciones o la entrega certificada. Para estos servicios, resulta fundamental garantizar la seguridad, la equidad y la confidencialidad del proceso de entrega.

Como trabajo futuro se plantea extender el protocolo a escenarios multiparte, en los que un mismo documento debe ser entregado a varios destinatarios.

Agradecimientos

Este trabajo ha sido financiado por el proyecto BLOB-SEC (Proyecto PID2021-122394OB-I00 financiado por MICIU/AEI/10.13039/501100011033 y por "FEDER/EU").

Referencias

[1] M. F. Hinarejos, J. L. Ferrer-Gomila , A. P. Isern-Deyà and G. F. Chévez-Alvarado: "A Blockchain-Based Protocol for Fair Delivery for Receipts", *Future Internet*, vol. 17, 2025.

[2] A. Bahreman and J. D. Tygar: "Certified Electronic Mail", *Proceedings of the Network and Distributed System Security Conference (NDSS)*, pp. 3–19, 1994.

[3] A. Tauber, P. Kustor and B. Karning: "Cross-border certified electronic mailing: A European perspective", *Computer Law & Security Review*, vol. 29, pp. 28–39, 2013.

[4] G. Wood: "Ethereum: A Secure Decentralised Generalised Transaction Ledger EIP-150 Revision (759dccd - 2017-08-07)". https://ethereum.github.io/yellowpaper/paper.pdf (accedido el 30 de Octubre 2025).

[5] Ethereum.org: "The Community-Run Technology Powering the Cryptocurrency Ether (ETH) and Thousands of Decentralized Applications". https://ethereum.org/ (accedido el 30 de Octubre 2025).

[6] M. Mut-Puigserver, M. M. Payeras-Capellà and M. A. Cabot-Nadal: "Blockchain-based fair certified notifications", *Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2018 International Workshops, DPM 2018 and CBT 2018*, pp. 20–37, 2018.

[7] M. M. Payeras-Capellà, M. Mut-Puigserver and M. A. Cabot-Nadal: "Blockchain-based system for multiparty electronic registered delivery services", *IEEE Access*, vol. 7, pp. 95825–95843, 2019.

[8] M. Payeras-Capellà, M. Mut-Puigserver and M. À. Cabot-Nadal: "Smart contract for multiparty fair certified notifications", *Proceedings of the 2018 Sixth International Symposium on Computing and Networking Workshops (CANDARW)*, pp. 459–465, 2018.

[9] M. A. Cabot-Nadal, M. Mut-Puigserver, M. M. Payeras-Capellà and R. Pericàs-Gornals: "Confidential Certified Notification Protocol using Rejectable Soulbound Tokens and Identity-Based Cryptography", *IEEE Access*, vol. 11, pp. 142495–142514, 2023.

[10] M. Mut-Puigserver, M. A. Cabot-Nadal and M. M. Payeras-Capellà: "Removing the Trusted Third Party in a Confidential Multiparty Registered eDelivery Protocol Using Blockchain", *IEEE Access*, vol. 8, pp. 106855–106871, 2020.

[11] M. Mut-Puigserver, R. Pericàs-Gornals, M. M. Payeras-Capellà and M. À. Cabot-Nadal: "Improving the Efficiency of a Blockchain-Based Confidential Registered e-Delivery Protocol", *Advanced Information Systems Engineering Workshops*, pp. 5–17, 2022.

[12] M. F. Hinarejos, J. L. Ferrer-Gomila and L. Huguet-Rotger: "A solution for secure certified electronic mail using Blockchain as a secure message board", *IEEE Access*, vol. 7, pp. 31330–31341, 2019.

[13] K. Elmaghraby and T. Dimitriou: "Blockchain-Based Fair and Secure Certified Electronic Mail Without a TTP", *IEEE Access*, vol. 9, pp. 100708–100724, 2021.

[14] M. F. Hinarejos and J. L. Ferrer-Gomila: "A solution for secure multiparty certified electronic mail using blockchain", *IEEE Access*, vol. 8, pp. 102997–103006, 2020.

[15] S. Kremer and O. Markowitch: "Selective Receipt in Certified E-mail", *Progress in Cryptology—INDOCRYPT 2001*, pp. 136–148, 2001.

[16] M. M. Payeras-Capellà, M. Mut-Puigserver, J. L. Ferrer-Gomila and L. Huguet-Rotger: "No Author Based Selective Receipt in an Efficient Certified E-mail Protocol", *Proceedings of the Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, pp. 387–392, 2009.

[17] Sepolia contract: https://sepolia.etherscan.io/address/0x24a835494a443e59eA034547e8447DF937924Df9 (accedido el 07 de Enero 2026).

Tweak Meets Hashing: Improving Padding-Free Sponges

Mario Marhuenda Beltrán
Radboud University
The Netherlands
mmarhuenda@cs.ru.nl

***Abstract*—The sponge is a popular construction for hashing and keyed hashing, and the duplex for authenticated encryption. They are proven to behave like a random oracle up to approximately $2^{c/2}$ queries, where c is the so-called capacity. In the arithmetization-oriented setting, which operates on elements from a finite field of size p, security is guaranteed up to $p^{c/2}$. Classically, to hash securely, the sponge needs to injectively pad the message. Correspondingly, authenticated encryption schemes often flip bits in the inner part to ensure domain separation. While these bit manipulations have little (but non-zero) effect on the efficiency and security in case of a field of size 2, they become more profound for larger fields. For example, `ReinforcedConcrete` operates on a field with $p \approx 2^{256}$, absorbs 2 elements per permutation evaluation, and has a capacity $c = 1$. Consequently, injective padding results in superfluous permutation evaluations half of the time, and domain separation in the inner part would reduce the capacity to 0 and thus void security. Alternatives to padding have been proposed, such as `SAFE` [1], `GSponge` [2] and `sponge-pi` [3]. These solutions either use some kind of length encoding, or resort to so-called non-cryptographic permutations to transform the inner state.**

Iwata and Minematsu introduced the primitive of tweakable permutation. In this work, we leverage this primitive to encode the length into the permutation, this allows, at no security overhead, to absorb an extra $c/2$ bits on the first absorption, and to squeeze $c/2 - \log_2(c)$ extra bits on every squeeze, an optimization that was lost in `SAFE`, while achieving concretely better security than `sponge-pi`.

Finally, the tweak allows for a more fine-grained security bound, with applications to zero-knowledge proof systems.

***Index Terms*—sponge, field elements, padding, SAFE, indifferentiability**

I. Introduction

The sponge hash function construction of Bertoni et al. [4] has been the inspiration of dozens of symmetric cryptographic modes for hashing and authenticated encryption in the last decades. It operates on a b-bit state, which is split into an outer part of size r (the *rate*) and inner part of size c (the *capacity*). Internally, it uses a cryptographic b-bit permutation $\mathcal{P}$. To hash a message M, it is first *injectively padded* to a string of size a multiple of r bits, subsequently cut into r-bit blocks, and these blocks are then added to the outer part of the state, interleaved by an evaluation of $\mathcal{P}$ on the state. After the last message block is absorbed, the digest is squeezed from the outer part of the state in r-bit blocks, again interleaved by evaluations of $\mathcal{P}$. If we assume that $\mathcal{P}$ is a random permutation, this construction is proven to behave like a random oracle in the indifferentiability framework [5], [6] as long as the total number of permutation evaluations is at most $2^{c/2}$ [7]. Later, Naito and Ohta proved that the same bound holds, even if the initial absorption is $r + c/2$ bits, and squeezing is performed at $r + c/2 - \log_2(c)$ bits at a time [8].

The sponge construction is the core behind the Keccak hash function [9], later adopted to the SHA-3 hash function standard [10]. This standard operates on a state of $b = 1600$ bits, and has a capacity ranging from $c = 448$ (with rate $r = 1152$) to $c = 1024$ (with rate $r = 576$). Another famous hash function based on the sponge is Ascon-Hash, from the Ascon suite [11], recently selected for standardization as lightweight cryptographic hashing and authenticated encryption scheme [12]. This upcoming standard operates on a state of $b = 320$ bits, with a capacity ranging from $c = 192$ to $c = 256$. Additionally, the sponge has a sister construction, the duplex [13], which allows interleaving of absorb and squeeze calls, which has also enjoyed great popularity.

A. The Cost of Padding

In the binary setting, the message expansion incurred by the padding is not so much of an efficiency penalty. Indeed, the minimum injective padding, or 10^*-padding, adds a single 1 and a sufficient number of 0's so that the message M is of size a multiple of r bits. For a sponge evaluation, this incurs an extra permutation evaluation *only if* the unpadded M turns out to have a size divisible by r. In other words, padding is not the bottleneck in bit-oriented cryptographic schemes. We see a comparable phenomenon in keyed applications of the sponge.

However, the situation drastically changes when we consider schemes that are evaluated on prime field elements rather than bits, so-called arithmetization-oriented or finite-field friendly schemes. The rise of these applications in the last years, notably in zero-knowledge proof systems [14], fully homomorphic encryption [15], and multiparty computation in the head (MPCiTH) [16], has led to a large number of cryptographic hash functions, such as `Poseidon` [17], `Poseidon2` [18], `Anemoi` and `Jive` [19], `MiMC` [20], `ReinforcedConcrete` [21], `XHash` [22], Tip5 [23], and `Monolith` [24]. Some of these designs propose permutations that operate over a large field, $\mathbb{F}_p$ for a large p, and are instantiations of the sponge.

As the security proof of the sponge construction is field-agnostic, the bound carries over mutatis mutandis. Notably: if the permutation does not operate on $\mathbb{F}_2$ but rather $\mathbb{F}_p$, the construction is secure as long as the number of permutation evaluations is at most $p^{c/2}$. Naturally, in field-oriented designs, b, c, and r are smaller. For example, `Poseidon` [17] takes $p \geq 2^{32}$ (requiring $c = 8$ to achieve approximately 128-bit security), `XHash` [22] takes $p \approx 2^{64}$, and Reinforced Concrete [21] even operates with $p \approx 2^{256}$ and operates on a state of $b = 3$ elements, split into a capacity of $c = 1$

and a rate of $r = 2$. If such a function is used for general-purpose hashing, a superfluous permutation evaluation just to process the padding has to be made for half of the messages on average. This can be quite costly [25].

B. The Rise of Padding-Free Sponges

To solve the issue of padding, and various other issues in arithmetization-oriented hashing, Aumasson et al. [26] proposed SAFE: Sponge API for Field Elements. SAFE is a generic API for sponge functions specifically tailored towards its use on field elements. In a nutshell, SAFE requires the user to first *hash* the input-output pattern IO of the sponge evaluation into a digest, that can be placed in the inner part of the initial state, and subsequently it allows for element-wise absorption and squeezing as long as these operations obey the initial input-output pattern IO. The hash function evaluation can then additionally hash certain other bit-based data or a domain separator D. This construction was recently proven secure up to $p^{c/2}$ queries by Khovratovich et al. [1], and SAFE has been implemented in the Neptune hash framework and some zero-knowledge proof projects [27], [28]. A comparable construction has recently been introduced by Ashur and Singh Bhati [2].

The core idea of the input-output pattern is that, in many applications, such as commitment schemes, Fiat-Shamir protocols, and verifiable encryption of cryptocurrency transactions, the exact input-output pattern is known and fixed in advance, and even stays constant for many evaluations. By thus hashing this pattern into the inner state, one de facto considers sponge hashing with a prefix-free padding, and no 10^*-padding needs to be performed. Of course, for general-purpose applications, using the input-output pattern, or prefix-free padding in general, is not an ideal solution.

A different, more flexible approach is that of Beltrán et al. [3], which uses non-cryptographic permutations (NCPs) to permute the capacity, allowing for pseudo-domain separation, an idea first exploited by Hirose in [29] and later in [30].

C. Tweakable Permutations

Iwata and Minematsu [31] introduced the primitive of tweakable permutation, which can be seen as a collection of permutations indexed by a so-called *tweak*. Seemingly, this provides a primitive that lies in strength somewhat between a permutation and a tweakable block cipher. Yet, tweakable permutations can be easily built from permutation via the tweakable Even-Mansour construction [32], in particular, the cost of adding the tweak is small, but as we will show, it can result in considerable security gains.

D. Our Contributions

In this work, we consider an alternative approach: We investigate the security of `tsponge`, a sponge based construction, requiring no padding, only a tweak that is used for pseudo domain separation. We will also show that in some practical applications there is a big security gain, because this pseudo domain separation also means that collisions originating from different tweaks do not degrade security.

E. Outline

The remainder of this paper is organized as follows. First, in Section II-A we present our novel `tsponge` construction, second, in Section III, we present its security analysis, and lastly in Section III-A we conclude and present interesting future work directions.

F. Preliminaries

1) Notation: We denote $\boldsymbol{A} \sqcup \boldsymbol{B}$ as the disjoint union of sets $\boldsymbol{A}$ and $\boldsymbol{B}$. The empty set is denoted by ϵ. If $\boldsymbol{S}$ is a finite set, we write $x \xleftarrow{\$} \boldsymbol{S}$ to denote that x is sampled uniformly from $\boldsymbol{S}$.

Let $n \in \mathbb{N}$ such that $n > 1$. We will abbreviate the set $\{1, \ldots, n\}$ as $[n]$. Given $k < n$, we denote by $(n)_k$ the falling factorial of n of depth k, i.e. $\prod_{i=0}^{k-1}(n-i)$.

We denote $\mathbb{F}_p^{\leq a} = \bigcup_{i=0}^{a} \mathbb{F}_p^i$, $\mathbb{F}_p^* = \bigcup_{i=0}^{\infty} \mathbb{F}_p^i$, and $\mathbb{F}_p^\infty = \prod_{i=1}^{\infty} \mathbb{F}_p^i$. I.e., $\mathbb{F}_p^*$ is the set of all *finite* tuples of elements of $\mathbb{F}_p$ of any length, and $\mathbb{F}_p^\infty$ the set of all *infinite* tuples of elements of $\mathbb{F}_p$. Note that $\mathbb{F}_p^*$ contains the empty tuple, which we also denote ϵ. We denote by $\langle n \rangle_p^\ell$ the encoding of the positive integer n as an element of $\mathbb{F}^\ell$.

Let $s = (s_1, s_2, \ldots, s_n)$ and $s' = (s'_1, s'_2, \ldots, s'_m) \in \mathbb{F}_p^*$, and $\ell, \ell' \in \mathbb{N}$ with $1 \leq \ell \leq \ell'$. We denote by $s \| s'$ the tuple $(s_1, \ldots, s_n, s'_1, \ldots, s'_m)$. In particular, if $s = (1)$ and s' is of the form 0^α, we abbreviate the concatenation as 10^α. Moreover, if $n = m$, we denote $s \oplus s'$ as the element-wise addition of s and s'. Moreover, $s[\ell : \ell']$ refers to the empty tuple if $\ell > n$, and $(s_\ell, \ldots, s_{\max\{\ell', n\}})$ otherwise. If $\ell \leq n$, we define $\texttt{left}_\ell(s)$ to be $s[1 : \ell]$ and $\texttt{right}_\ell(s)$ to be $s[n - \ell : n]$.

We denote by $\texttt{pad}_{r_I, r_A}$ the function that takes as input a tuple $M \in \mathbb{F}_p^*$, and returns a tuple of field blocks $\overline{M} \in \mathbb{F}_p^{\leq r_I} \times (\mathbb{F}_p^{\leq r_A})^*$ such that only the last block of the tuple may be incomplete. $\texttt{pad}_{r_I, r_A}(\cdot)$. Additionally, we will overload the notation by denoting $\texttt{pad}_r(\cdot) = \texttt{pad}_{r,r}(\cdot)$ and $\texttt{pad}_r(\cdot) = \texttt{pad}_{r,r}(\cdot)$.

Let $\mathsf{RO}^\infty : \mathbb{F}_p^* \to \mathbb{F}_p^\infty$ denote a random oracle in the sense of Bellare and Rogaway [33]. From RO^∞ we build the function $\mathsf{RO} : \mathbb{F}_p^* \times \mathbb{N} \to \mathbb{F}_p^*$, which, on input (M, n), returns $\mathsf{RO}^\infty(M)[1 : n]$. We will abuse notation and refer to RO as a random oracle.

G. Tweakable Permutation

We end the preliminaries by defining the concept of tweakable permutations: Let p be a prime and let $n, \tau \geq 1, s \geq 1$. A tweakable permutation $\mathsf{TP} : \mathbb{F}_p^b \times \mathbb{F}_p^\tau \to \mathbb{F}_p^b$ is a function that satisfies the following condition: for each $t \in \mathbb{F}_p^\tau$, $\mathsf{TP}(\cdot, t) : \mathbb{F}_p^b \to \mathbb{F}_p^b$ is invertible (i.e. a permutation).

Furthermore, if each function $\mathsf{TP}(\cdot, t)$ is independent uniformly random, then we say that TP is a `TURP`, or tweakable uniformly random permutation. We denote by $\texttt{tperm}(n, \tau)$ the space of tweakable uniformly random permutations with tweak size τ and permutation size n.

There are a few ways to build a `TURP` using a random permutation as a primitive. One possible way to set up a tweakable permutation is by adapting the `TWEAKEY` Framework proposed by Jean et al. [34], while another possible way, is to use the tweakable Even-Mansour construction, or `XPX` [32]. This shows that, security-wise, the notion of a `TURP` is equivalent in strength to that of a random permutation.

II. Tweakable Sponge Hash Function

In this section, we introduce the tweakable-sponge $\texttt{tsponge}$. $\texttt{tsponge}$ relies on a random permutation, and encoding function $\texttt{H} : \mathbb{F}_p^* \mapsto \mathbb{F}_p^\tau$, which is indicated by the superscript: $\texttt{tsponge}^{\texttt{H}}$. Furthermore, it takes as input a message $M \in \mathbb{F}_p^*$ a tweak $t \in \mathbb{F}_p^\tau$ and outputs a digest $Z \in \mathbb{F}_p^n$.

$\texttt{tsponge}^{\texttt{H}}$ uses the sponge, together with a tweakable permutation and the modifications suggested by Hirose [29], [3]. At a high level $\texttt{tsponge}^{\texttt{H}}$ operates in two phases: absorb and squeeze. We simplify the following for the sake of exposition. During these phases, the permutation is split into two parts: the rate and the capacity. During the absorption phase, the message is added and is split into blocks of size equal to the rate, using the function $\texttt{ZPad}$.

We remark that despite its name, $\texttt{ZPad}$ does not result in any expansion of the message. Next the message is xored into the rate part of the state, while permuting the state during each step. Finally, during squeezing the rate is released as output.

A formal description of $\texttt{tsponge}^{\texttt{H}}$ is given in Algorithm 1 and Fig. 1. Furthermore, the rate and capacity during the first evaluation, absorbing and squeezing may differ between the first evaluation, the absorbing phase, and the squeezing phase. We denote this respectively by r_I, c_I, r_A, c_A and r_S, c_S. The reason for this is twofold: Firstly, during the squeezing phase, it is not possible to control the capacity, hence its size may be decreased, allowing for a larger throughput. Secondly, during the first call the capacity has to maintain a certain fixed value, therefore, its size may be decreased as well, allowing for faster absorption. In general, $r_I = r_S = r_A + c_A/2$.

a) Choice of $\texttt{H}$*:* In order to properly instantiate $\texttt{tsponge}$, we need to choose an encoding function $\texttt{H}$. In fact, the only requirement for $\texttt{H}$ is that no messages with the same length of their last block, map to the same value. More formally, we require that for any $M, M' \in \mathbb{F}_p^*$ padded to $M_1 \| \cdots \| M_\ell$ and $M'_1 \| \cdots \| M'_{\ell'}$ such that if $\ell = \ell'$, and $|M_\ell| \neq |M'| \text{mod} r_A$, we have $\texttt{H}(M) \neq \texttt{H}(M')$. Of course, one simple way to satisfy this requirement is to define at $\texttt{H} = \langle \ell \rangle_p^{r_A}$.

Algorithm 1 $\texttt{tsponge}$

Function $\texttt{tsponge}^{\texttt{H}}$
Input: $(IV, M, n) \in \mathbb{F}_p^b \times \mathbb{F}_p^* \times \mathbb{N}_+$
Output: $Z \in \mathbb{F}_p^*$

$S \leftarrow IV$
$Z \leftarrow \epsilon$
$t \leftarrow \texttt{H}(M)$
$(M_1, \ldots, M_\ell), \leftarrow \texttt{pad}_{r_I, r_A}(M)$
for $1 \leq i \leq \ell$ **do**
    $S \leftarrow \mathcal{P}\left(S \oplus (M_i \| 0^*), t\right)$
end for
for $1 \leq i \leq \lceil \frac{n}{r} \rceil$ **do**
    $Z \leftarrow Z \| \texttt{left}_{r_S}(S)$
    $S \leftarrow \mathcal{P}(S, t)$
end for
return $Z[1:n]$

A. Indifferentiability

The indifferentiability framework was first introduced by Maurer et al. [5], and refined to the context of hash functions by Coron et al. [6]. Translating into our setting, given a $\texttt{tsponge}^{\texttt{H}}$, relying on a tweakable permutation $\mathcal{TP}$, intuitively, we say that it is indifferentiable if there exists a simulator, $\texttt{S}$, so that $(\texttt{tsponge}^{\texttt{H}}, \mathcal{TP})$ is hard to distinguish from $(\mathsf{RO}, \texttt{S}^{\mathsf{RO}})$, where access to $\mathcal{TP}$ also allows access to its inverse function, $\texttt{S}^{\mathsf{RO}}$ indicates that $\texttt{S}$ has oracle access to RO, and analogously with $\texttt{tsponge}^{\texttt{H}}$. We usually refer to $(\texttt{tsponge}^{\texttt{H}}, \mathcal{TP})$ as the real world and to $(\mathsf{RO}, \texttt{S}^{\mathsf{RO}})$ as the ideal world. More precisely, given an adversary,$\mathcal{A}$ interacting with either world, the indifferentiability advantage of $\mathcal{A}$ with respect to $\texttt{S}^{\mathsf{RO}}$ is:

$$\mathsf{Adv}^{\texttt{if}}_{\texttt{C}}(\mathcal{A}) \triangleq \left| \Pr\left[\mathcal{A}[\mathsf{RO}, \texttt{S}^{\mathsf{RO}}] \to 1\right] - \Pr\left[\mathcal{A}[\texttt{tsponge}, \mathcal{TP}] \to 1\right] \right| ,$$

where the resources of $\mathcal{A}$ are the number of queries made to the inner primitive (respectively the simulator) and to the construction (respectively RO). Queries to $\texttt{tsponge}^{\texttt{H}}$ are counted by how many permutation queries it requires to answer them in the real world. Queries to RO are counted by how many permutation queries *would have been required* in order to answer them in the real world, had the same query been made.

III. Security of tsponge

We will state the security claim for $\texttt{tsponge}$. In short, we have the following result:

Theorem 1: Let $\mathcal{N}, \mu \in \mathbb{N}$ be positive integers. There exists a simulator $\texttt{S}$ such that, for any distinguisher $\mathcal{A}$ making at most $\mathcal{N}$ permutation evaluations, and repeating the same tweak at most μ times, we have:

$$\mathbf{Adv}^{\text{iff}}_{\texttt{C},\texttt{S}}(\mathcal{A}) \leq \mathcal{O}\left(\frac{\mu\mathcal{N}}{p^{c_A}} + \frac{\binom{\mathcal{N}}{2}}{p^b}\right).$$

The proof is not very involved, but technical, and far from trivial. Suffice it to say it is inspired by the proof of Naito and Ohta [8], with the addition of improvements made by [1], and the improved multicollision technique of [35], [36]. The security improvement comes in setting where μ is restrained, for example, if the messages length follow some kind of distribution, where the mode does not have a high probability. A natural question is the following: can we encode an IO pattern à la SAFE [1] into the tweak, and thus obtain a padding-free sponge, where we can mix absorb and squeeze?. In fact, we can.

Corollary 1: Let $\mathcal{N}$, $\texttt{S}$ and $\mathcal{A}$ be as Theorem 1, and $\texttt{H}$ be a random function. There exists a simulator $\texttt{S}$ such that, for any distinguisher $\mathcal{A}$ making at most $\mathcal{N}$ permutation evaluations.

$$\mathbf{Adv}^{\text{iff}}_{\texttt{C},\texttt{S}}(\mathcal{A}) \leq \mathcal{O}\left(\frac{2\mathcal{N}^2}{p^{c_A+\tau}} + \frac{(3\tau \log p + 4)\mathcal{N}}{p^{c_A}} + \frac{\binom{\mathcal{N}}{2}}{p^b}\right).$$

a) Remark on the interpretation of the bound: Notice that, for the bound Corollary 1 to apply, we first need to find a hash function, $\texttt{H}$ that remains indifferentiable from a random oracle up to, *at least*, the same bound. This is far from trivial and the main drawback of $\texttt{tsponge}$. For example, if $\texttt{H}$ was initialized with a sponge of the capactiy c, then at most $c/2$ bist of security can be achieved. So $\texttt{tsponge}$ would be more useful in the combiner setting.

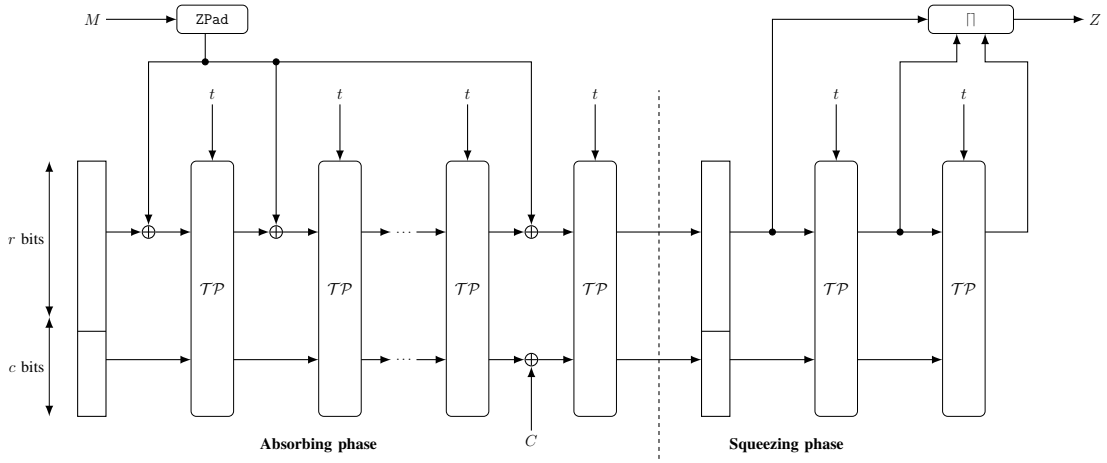


Fig. 1. Description of the tsponge construction.

Before we proceed with the proof sketch, we will need the following result:

Lemma 1: Let $q, R \in \mathbb{N}$. Consider the experiment of throwing uniformly at random q balls in R bins. For $u \in [R]$, denote by S_u the size of the bin number u. Then,

$$\mathbb{E}\left[\max_{u\in[R]} S_u\right] \leq \frac{2q}{R} + 3\ln(R) + 4\,.$$

A weakened version of this result was first proposed by Choi et al. [35], which was later improved by Lefevre and Mennink [36], and a result can be also found in [36].

Proof Sketch: Consider the real world $W_R = (\texttt{tsponge}, \mathcal{TP})$ and the ideal world $W_I = (\mathsf{RO}, \mathsf{S}^{\mathsf{RO}})$, consider an adversary $\mathcal{A}$ interacting with either world, and let Δ be its indifferentiability advantage. Let Γ be the transcript of the interaction, that is, the set $\{t_i, x_i, y_i\}_{i=1}^{\mathcal{N}}$, where x_i are the queries $\mathcal{A}$ made to the permutation with tweak t_i (respectively the simulator), and y_i the corresponding answers.

We can define a bad event, **Bad**, that is, a particular set of interactions between $\mathcal{A}$ and the real world that, if neither of them occurs, then the real world and the ideal world are identical distributed, then we have by the fundamental lemma of game playing [37] that $\Delta \leq \Pr[\textbf{Bad}]$. Defining this event, and proving that the distribution of W_R conditioned on $\neg$**Bad** is equal to that of W_I conditioned on $\neg$**Bad** is an important aspect of the proof, but one that will take too much space. Suffice to say, that the argument made on [3] or [8] or [1] can be adapted to our setting almost mutatis mutandis.

Skipping this allows us to move to the interesting part of the proof, which is to bound $\Pr$ **Bad**. We define the random variable $\mu_{\texttt{max}}$ as:

$$\mu_{\texttt{max}} = \max \#\left\{t_i \in \mathbb{F}_p^\tau \mid \exists\, x_i, y_i : (t_i, x_i, y_i) \in \Gamma\right\}\,.$$

That is, $\mu_{\texttt{max}}$ is the maximum number of queries made by $\mathcal{A}$ to the permutation (respectively simulator) with the same tweak.

Hence, we have:

$$\begin{aligned}
\Delta &\leq \Pr[\textbf{Bad}] \\
&= \sum_{i=1}^{\mathcal{N}} \Pr[\textbf{Bad}_i \mid \mu_{\texttt{max}} = i]\Pr[\mu_{\texttt{max}} = i] && (1)\\
&\leq \mathcal{O}\left(\frac{\binom{\mathcal{N}}{2}}{p^b} + \sum_{i=1}^{\mathcal{N}} \frac{\mathcal{N}}{p^{c_A}} i \Pr[\mu_{\texttt{max}} = i]\right) && (2)\\
&\leq \mathcal{O}\left(\frac{\binom{\mathcal{N}}{2}}{p^b} + \frac{\mathbb{E}[\mu_{\texttt{max}}]\mathcal{N}}{p^{c_A}}\right), && (3)
\end{aligned}$$

where (1) follows by the law of total probability and (2) follows by the definition of expectation. In order to bound $\mu_{\texttt{max}}$, we notice that we have a bin and balls problem, with $\mathcal{N}$ balls and p^τ bins, hence, by Lemma 1, we have:

$$\mathbb{E}[\mu_{\texttt{max}}] \leq \frac{2\mathcal{N}}{p^\tau} + 3\tau\log p + 4\,. \quad (4)$$

Putting (3) and (4) together concludes the proof of Corollary 1. ∎

A. Conclusion

In this work we have introduced a novel sponge construction that rides on the recent trend of tweakable permutations. Finding a good instantiation tsponge remains an open problem.

B. Future Work

The combination of tweakable permutations has been shown to be quite powerful, as has been shown in this work and also in [38]. It would be interesting to see whether other constructions, such as the duplex, or authenticated encryption schemes, can also benefit from this primitive. Another interesting direction is whether it is possible to encode extra information in the tweak, for example, the nonce in sponge base AEAD schemes, such as MonkeySponge, SpongeWrap and MonkeySpongeWrap [39].

Finally, formal verification in the symmetric cryptography setting has been notoriously lacking, only the Naito and Ohta [8] construction has been formally verified. This was achieved by Almeida et al. [40], using EasyCrypt [41]. EasyCrypt is specially well suited for game based proofs,

however, it is not known how to formalize proofs using the H-Coefficient technique [42], let alone the chi-squared technique [43] in EasyCrypt, possibly requiring other techniques.

ACKNOWLEDGMENTS. The author is supported by the Netherlands Organisation for Scientific Research (NWO) under grant VI.Vidi.203.099. The author would like to thank Bart Mennink and Charlotte Lefevre for the fruitful discussions and their overall guidance. Furthemore, the author would like to thank the reviewers for their insightful comments and suggestions.

REFERENCES

[1] D. Khovratovich, M. M. Beltrán, and B. Mennink, "Generic security of the SAFE API and its applications," ser. Lecture Notes in Computer Science, vol. 14445, 2023, pp. 301–327.

[2] T. Ashur and A. S. Bhati, "Generalized indifferentiable sponge and its application to polygon miden VM," Cryptology ePrint Archive, Paper 2024/911, 2024. [Online]. Available: https://eprint.iacr.org/2024/911

[3] C. Lefevre, M. M. Beltrán, and B. Mennink, "To pad or not to pad? padding-free arithmetization-oriented sponges," *IACR Trans. Symmetric Cryptol.*, vol. 2025, no. 1, pp. 97–137, 2025.

[4] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Sponge Functions," 2007.

[5] U. M. Maurer, R. Renner, and C. Holenstein, "Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology," ser. Lecture Notes in Computer Science, M. Naor, Ed., vol. 2951, pp. 21–39.

[6] J. Coron, Y. Dodis, C. Malinaud, and P. Puniya, "Merkle-Damgård Revisited: How to Construct a Hash Function," ser. Lecture Notes in Computer Science, V. Shoup, Ed., vol. 3621, 2005, pp. 430–448.

[7] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "On the Indifferentiability of the Sponge Construction," ser. Lecture Notes in Computer Science, N. P. Smart, Ed., vol. 4965, pp. 181–197.

[8] Y. Naito and K. Ohta, "Improved Indifferentiable Security Analysis of PHOTON," ser. Lecture Notes in Computer Science, vol. 8642, 2014, pp. 340–357.

[9] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Keccak," in *International Conference on the Theory and Application of Cryptographic Techniques*, 2013.

[10] National Institute of Standards and Technology, "FIPS PUB 202: SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions," August 2015. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf

[11] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, "Ascon v1.2," Winning Submission to NIST Lightweight Cryptography, 2021.

[12] NIST, "Lightweight Cryptography," February 2019, https://csrc.nist.gov/Projects/Lightweight-Cryptography.

[13] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche, "Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications," ser. Lecture Notes in Computer Science, vol. 7118, 2011, pp. 320–337.

[14] A. Fiat and A. Shamir, "How to Prove Yourself: Practical Solutions to Identification and Signature Problems." in *Crypto*, vol. 86. Springer, 1986, pp. 186–194.

[15] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Stanford University, USA, 2009.

[16] J. Katz, V. Kolesnikov, and X. Wang, "Improved Non-Interactive Zero Knowledge with Applications to Post-Quantum Signatures," 2018, https://eprint.iacr.org/2018/475. [Online]. Available: https://eprint.iacr.org/2018/475

[17] L. Grassi, D. Khovratovich, C. Rechberger, A. Roy, and M. Schofnegger, "Poseidon: A new hash function for zero-knowledge proof systems." USENIX Association, 2021, pp. 519–535. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/grassi

[18] L. Grassi, K. Koschatko, and C. Rechberger, "Poseidon and neptune: Gröbner basis cryptanalysis exploiting subspace trails," *IACR Trans. Symmetric Cryptol.*, vol. 2025, no. 2, pp. 34–86, 2025.

[19] C. Bouvier, P. Briaud, P. Chaidos, L. Perrin, R. Salen, V. Velichkov, and D. Willems, "New Design Techniques for Efficient Arithmetization-Oriented Hash Functions: Anemoi Permutations and Jive Compression Mode," ser. Lecture Notes in Computer Science, vol. 14083, 2023, pp. 507–539.

[20] M. R. Albrecht, L. Grassi, C. Rechberger, A. Roy, and T. Tiessen, "MiMC: Efficient Encryption and Cryptographic Hashing with Minimal Multiplicative Complexity," ser. Lecture Notes in Computer Science, vol. 10031, 2016, pp. 191–219.

[21] L. Grassi, D. Khovratovich, R. Lüftenegger, C. Rechberger, M. Schofnegger, and R. Walch, "Reinforced Concrete: A Fast Hash Function for Verifiable Computation." Association for Computing Machinery, 2022, p. 1323–1335.

[22] T. Ashur, A. S. Bhati, A. Kindi, M. Mahzoun, and L. Perrin, "XHash: Efficient STARK-friendly Hash Function," Cryptology ePrint Archive, Paper 2023/1045, 2023.

[23] A. Szepieniec, A. Lemmens, J. F. Sauer, B. Threadbare, and Al-Kindi, "The Tip5 Hash Function for Recursive STARKs," Cryptology ePrint Archive, Paper 2023/107, 2023.

[24] L. Grassi, D. Khovratovich, R. Lüftenegger, C. Rechberger, M. Schofnegger, and R. Walch, "Monolith: Circuit-Friendly Hash Functions with New Nonlinear Layers for Fast and Constant-Time Implementations," *IACR Trans. Symmetric Cryptol.*, vol. 2024, no. 3, pp. 44–83, 2024.

[25] D. Hopwood, S. Bowe, T. Hornby, and N. Wilcox, "ZCash protocol specification," 2023, https://github.com/zcash/zips/blob/master/protocol/protocol.pdf.

[26] J. Aumasson, D. Khovratovich, and P. Quine, "SAFE: Sponge API for Field Elements," Cryptology ePrint Archive, Paper 2023/522, 2023, https://eprint.iacr.org/2023/522. [Online]. Available: https://eprint.iacr.org/2023/522

[27] S. Setty, "Nova: open source implementation."

[28] M. Maller and D. Khovratovich, "Baloo: open source implementation," 2022, https://github.com/mmaller/caulk-dev/tree/main/baloo.

[29] S. Hirose, "Sequential Hashing with Minimum Padding," *Cryptogr.*, vol. 2, no. 2, p. 11, 2018. [Online]. Available: https://doi.org/10.3390/cryptography2020011

[30] A. Gunsing, "Block-cipher-based tree hashing," ser. Lecture Notes in Computer Science, Y. Dodis and T. Shrimpton, Eds., vol. 13510. Springer, 2022, pp. 205–233.

[31] K. Minematsu and T. Iwata, "Tweak-length extension for tweakable blockciphers," *IACR Cryptol. ePrint Arch.*, p. 888, 2015. [Online]. Available: http://eprint.iacr.org/2015/888

[32] B. Mennink, "XPX: generalized tweakable even-mansour with improved security guarantees," ser. Lecture Notes in Computer Science, M. Robshaw and J. Katz, Eds., vol. 9814, 2016, pp. 64–94.

[33] M. Bellare and P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols," pp. 62–73.

[34] J. Jean, I. Nikolic, and T. Peyrin, "Tweaks and keys for block ciphers: the TWEAKEY framework," *IACR Cryptol. ePrint Arch.*, p. 831, 2014.

[35] W. Choi, B. Lee, and J. Lee, "Indifferentiability of Truncated Random Permutations," ser. Lecture Notes in Computer Science, S. D. Galbraith and S. Moriai, Eds., vol. 11921, 2019, pp. 175–195.

[36] C. Lefevre and B. Mennink, "Generic Security of the Ascon Mode: On the Power of Key Blinding," ser. Lecture Notes in Computer Science, M. Eichlseder and S. Gambs, Eds., 2024.

[37] M. Bellare and P. Rogaway, "Code-Based Game-Playing Proofs and the Security of Triple Encryption," Cryptology ePrint Archive, Paper 2004/331, 2004, https://eprint.iacr.org/2004/331. [Online]. Available: https://eprint.iacr.org/2004/331

[38] T. Moss, L. Grassi, M. Steiner, F. Schmid, and H. Yan, "psquare-hash: A family of tweakable hash functions for physically secure pq signatures," unplublished manuscript, 2025.

[39] B. Mennink, "Understanding the Duplex and Its Security," *IACR Transactions on Symmetric Cryptology*, vol. 2023, no. 2, p. 1–46, Jun. 2023. [Online]. Available: https://tosc.iacr.org/index.php/ToSC/article/view/10976

[40] J. B. Almeida, C. Baritel-Ruet, M. Barbosa, G. Barthe, F. Dupressoir, B. Grégoire, V. Laporte, T. Oliveira, A. Stoughton, and P. Strub, "Machine-checked proofs for cryptographic standards: Indifferentiability of sponge and secure high-assurance implementations of SHA-3." ACM, 2019, pp. 1607–1622.

[41] M. Barbosa, G. Barthe, K. Bhargavan, B. Blanchet, C. Cremers, K. Liao, and B. Parno, "Sok: Computer-aided cryptography." IEEE, pp. 777–795.

[42] J. Patarin, "The "Coefficients H" Technique," ser. Lecture Notes in Computer Science, vol. 5381. Springer, 2008, pp. 328–345.

[43] W. Dai, V. T. Hoang, and S. Tessaro, "Information-theoretic indistinguishability via the chi-squared method," in *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*, ser. Lecture Notes in Computer Science, J. Katz and H. Shacham, Eds., vol. 10403. Springer, 2017, pp. 497–523. [Online]. Available: https://doi.org/10.1007/978-3-319-63697-9_17

Un marco de trabajo para la trazabilidad y el control de riesgos en la ciberseguridad de la cadena de suministro de software

Carlos Villarrubia*, Antonio Santos-Olmo†, Juan Manuel Vara*, Eduardo Fernández-Medina†
*Universidad Rey Juan Carlos, Móstoles, Madrid, España
carlos.villarrubia@urjc.es, juanmanuel.vara@urjc.es
†Universidad de Castilla-La Mancha, Ciudad Real, España
antonio.santosolmo@uclm.es, eduardo.fdezmedina@uclm.es

***Resumen*—La creciente complejidad del desarrollo de software y la fuerte dependencia de componentes de terceros han ampliado los ataques dirigidos a la cadena de suministro de software, comprometiendo la integridad, trazabilidad y confiabilidad de los artefactos software. No obstante, la literatura existente carece de una solución holística que integre el conjunto de características relevantes. Este trabajo aborda esta carencia proponiendo un marco de trabajo de código abierto basado en blockchain que integra la totalidad de las características relevantes. El marco de trabajo, resultado de una revisión sistemática de la literatura, proporciona una solución integral y ágil, alineada con las principales normativas de seguridad en la cadena de suministro de software. Asimismo, el marco habilita una trazabilidad de extremo a extremo e incorpora métricas de riesgo, así como una evaluación federada entre actores.**

***Index Terms*—Cadena de suministro de software, trazabilidad, gestión de dependencias, gestión de riesgos, seguridad**

I. Introducción

La cadena de suministro de software es el conjunto estructurado de procesos, recursos y actores involucrados en el diseño, construcción, validación y distribución de un producto de software [1]. Esta cadena suele incorporar tanto artefactos como dependencias desarrolladas internamente, así como otros obtenidos de repositorios públicos o servicios comerciales [2].

La eficiencia de la cadena se ha fortalecido recientemente por la adopción de prácticas como Desarrollo y Operaciones (DevOps), Integración Continua y Despliegue Continuo (CI/CD), junto con la automatización, lo que permite gestionar de manera continua las distintas etapas del proceso de desarrollo [3]. Estas prácticas reducen la intervención manual y la posibilidad de errores durante dicho proceso [4].

No obstante, la complejidad de la cadena de suministro introduce riesgos que pueden comprometer la trazabilidad, la integridad y la seguridad del software. Cada evento en la cadena de suministro, como la inclusión de un paquete, el resultado de una prueba o la publicación de un artefacto, puede convertirse en un punto vulnerable si no se registra y verifica adecuadamente [5]. La Figura 1 muestra el crecimiento sustancial de la cantidad de amenazas detectadas en la cadena de suministro de software en los últimos años.

En respuesta a estos riesgos, la comunidad ha comenzado a promover soluciones como el Software Bill of Materials (SBOM), que representa el inventario de los componentes y sus versiones en cada producto de software, facilitando el análisis de vulnerabilidades y el cumplimiento normativo [6]. Otras soluciones, como el marco de trabajo [7] han surgido para abordar la seguridad de la cadena de suministro desde distintas perspectivas, como la trazabilidad, la gestión de dependencias o el análisis de riesgos. No obstante, estos marco de trabajos tienen limitaciones compartidas, como una cobertura parcial de la cadena de suministro, la ausencia de validación o que no son de código abierto. Asimismo, muchas organizaciones, en particular las pequeñas y medianas empresas (PYMEs), se enfrentan a dificultades para adoptar soluciones, debido a que no disponen de tantos recursos económicos como las grandes empresas [8].

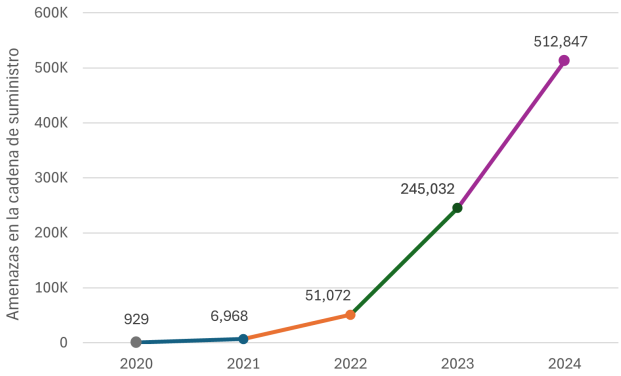


Figura 1: Amenazas detectadas anualmente en la cadena de suministro de software [1].

En este contexto, la tecnología blockchain ha emergido como una solución prometedora para reforzar la trazabilidad y garantizar la integridad en la cadena de suministro de software [9]. La naturaleza descentralizada, transparente e inmutable la convierte en una tecnología idónea para registrar y verificar cada evento dentro del desarrollo de software, eliminando la necesidad de confiar en una única entidad. Asimismo, la blockchain permite que cualquier intento de alteración quede evidenciado en los registros [10].

En este trabajo, se lleva a cabo una revisión sistemática de la literatura y un análisis de las propuestas existentes para reforzar la trazabilidad, control de riesgos y ciberseguridad en la cadena de suministro de software, la cual evidencia una necesidad en la comunidad científica de soluciones integrales.

[1]Datos obtenidos de los informes anuales publicados en *https://www.sonatype.com/*.

Como consecuencia, se presenta un marco de trabajo basado en blockchain, que actúa como un sistema integral de trazabilidad, control de riesgos y seguridad para la cadena de suministro de software. El marco de trabajo permite registrar las dependencias, los resultados de compilación y de pruebas, las entregas de artefactos, así como los procesos e interacciones humanas en toda la cadena de suministro. Todos estos eventos se vinculan mediante identificadores criptográficos y quedan registrados en la cadena de bloques, permitiendo su verificación posterior por cualquier parte interesada.

El resto del trabajo se estructura de la siguiente manera: en la Sección II se describen los conceptos relevantes relacionados con la cadena de suministro de software y trazabilidad; en la Sección III se presenta y analiza la revisión sistemática de la literatura realizada; en la Sección IV se describe la propuesta central de este trabajo y, finalmente, la Sección V se resume las principales aportaciones y el trabajo futuro derivado.

II. Antecedentes

Esta sección presenta los conceptos que comprenden el contenido del trabajo y la problemática asociada.

II-A. Cadena de suministro de software

La cadena de suministro en el desarrollo de software se caracteriza por la integración de múltiples actores, tecnologías y procesos que abarcan desde la escritura del código hasta la entrega del producto o servicio final [1]. No obstante, su integración y automatización pueden variar dependiendo de la metodología de desarrollo adoptada. Asimismo, esta cadena comprende tanto componentes desarrollados internamente como aquellos obtenidos de fuentes externas, todos integrados mediante herramientas y prácticas específicas [11].

En su inicio, la cadena incluye el código fuente, que puede ser propio o incorporado desde bibliotecas, marcos de trabajo y paquetes de terceros. Este código es gestionado en sistemas de control de versiones y constituye la base funcional del software.

A continuación, las herramientas de desarrollo intervienen, para permitir la edición, compilación, integración y pruebas del software. Estas herramientas forman parte del entorno técnico sobre el cual se construye el producto de software.

Posteriormente, el software es empaquetado y preparado para su uso mediante mecanismos de construcción y distribución, que garantizan su integridad y disponibilidad. Esto puede involucrar contenedores, repositorios, gestores de paquetes y otras plataformas que facilitan la entrega a entornos de ejecución o directamente al usuario final.

Finalmente, la cadena emplea componentes de infraestructura operativa, los cuales permiten desplegar y mantener el software en producción. Esto incluye servicios en la nube, entornos virtualizados y configuraciones automatizadas [1], [2].

II-B. Trazabilidad en la cadena de suministro de software

La trazabilidad en la cadena de suministro de software se refiere a la capacidad de identificar, registrar y rastrear el origen, la evolución y el uso de todos los componentes involucrados en el desarrollo y entrega de una solución digital [12]. De tal modo que se garantice la seguridad, la calidad y el cumplimiento normativo de la solución, permitiendo conocer en todo momento qué componentes fueron utilizados. La trazabilidad también facilita la auditoría, el control de cambios y de riesgos. Para ello, se emplean herramientas como sistemas de gestión de código fuente, automatización de integración y entrega continua, seguimiento de incidencias y generación de documentos como el SBOM, que detalla los componentes y sus relaciones [6].

Las principales limitaciones de la trazabilidad son la dispersión e inconsistencia de la información, la posibilidad de manipulación de registros y la falta de visibilidad integral en la cadena de suministro. La dependencia de múltiples herramientas desconectadas dificulta garantizar la autenticidad, seguridad y licenciamiento de los componentes, especialmente en entornos que integran software de terceros o código abierto [13].

II-C. Blockchain para la trazabilidad en la cadena de suministro de software

La tecnología blockchain consiste en un sistema digital descentralizado que permite el registro seguro, transparente e inmutable de información en un entorno distribuido [14]. La cadena funciona como un libro contable público y compartido entre múltiples nodos de una red, donde cada bloque almacena un conjunto de datos, generalmente transacciones, que se enlazan entre sí de forma cronológica mediante algoritmos criptográficos. Esta secuencia de bloques está protegida contra alteraciones, ya que cualquier intento de modificar un bloque implicaría rehacer todos los bloques posteriores [9].

No obstante, la adopción de la blockchain no es suficiente por sí sola para abordar los retos de la cadena de suministro de software. La trazabilidad que proporciona debe articularse con mecanismos de gestión e identificación de riesgos de ciberseguridad a lo largo de toda la cadena. Esto implica evaluar dependencias y proveedores, aplicar políticas de admisión, gestionar vulnerabilidades, verificar firmas y procedencia, y conservar evidencias de cumplimiento, todo ello respaldado por procesos organizativos y controles de gobernanza [13].

III. Revisión de la literatura

Esta sección presenta y analiza los resultados de la revisión de la literatura realizada, la cual se enfoca en la trazabilidad, control de riesgos y ciberseguridad de la cadena de suministro de software.

III-A. Proceso de revisión

La revisión sistemática de la literatura es un proceso estructurado para localizar, evaluar e interpretar de manera rigurosa estudios relevantes relacionados con una pregunta, temática o fenómeno específico, garantizando un enfoque metodológico reproducible. Dada la abundancia de fuentes y datos disponibles, resulta imprescindible llevar a cabo este proceso con rigurosidad [15].

La siguiente pregunta de investigación se definió para este estudio, con el objetivo de guiar todo el proceso de revisión. Asimismo, esta pregunta permitió identificar los estudios primarios sobre la trazabilidad de la cadena de suministro enfocada en el desarrollo de software.

> ¿Qué esfuerzos se han realizado sobre la trazabilidad en la cadena de suministro de software para garantizar su seguridad y control de riesgos?

Posteriormente, se definieron las áreas temáticas: cadena de suministro y seguridad. Para cada una de estas áreas, se seleccionó un conjunto de palabras clave, procurando maximizar tanto la relevancia como la cobertura de los estudios susceptibles de ser identificados. La Tabla I muestra las palabras clave seleccionadas, organizadas según su área temática correspondiente.

Tabla I: Keywords by area.

Supply chain	**Security**
Software supply chain	Dependency management
Supply chain security	Bill of materials
Open source supply chain	SBOM
	DevSecOps
	Risk management
	Traceability

Estas palabras clave se combinaron mediante operadores lógicos para construir la cadena de búsqueda que permitió recuperar los estudios pertinentes. La cadena empleada fue la siguiente:

> (“software supply chain” OR “supply chain security” OR “open source supply chain”) AND (“dependency management” OR “bill of materials” OR “SBOM” OR “risk management” OR “DevSecOps” OR “traceability”)

La cadena de búsqueda se adaptó y ejecutó en las bibliotecas digitales ACM, IEEE Xplore, Science Direct, Scopus y Web of Science, obteniendo un total de 577 estudios. A partir de esta selección, se aplicaron criterios de inclusión y exclusión obteniendo un total de 35 estudios relevantes, lo que representa el 6.1 % de los resultados identificados inicialmente.

Finalmente, se eliminaron los estudios duplicados que se encontraban en varias bibliotecas digitales. En concreto, el conjunto final estaba formado por 19 estudios primarios, que representan el 54.3 % del total de estudios relevantes inicialmente identificados.

III-B. Discusión y síntesis de los estudios primarios

Una vez identificados los estudios primarios, se determinó cuántos correspondían a cada una de las categorías definidas. En concreto, se determinó una mayor concentración de estudios en las categorías *Herramientas* y *Marcos de trabajo* con 6 cada una, *Enfoques* con 4 y *Otros* con 3. Esta distribución revela una fragmentación temática y la ausencia de una categoría que claramente domine en la literatura.

Posteriormente, se seleccionaron un conjunto de características que se consideraron las más relevantes para la cadena de suministro de software en este contexto. La Figura 2 muestra la contribución de los estudios primarios analizados en relación con cada una de las características seleccionadas. Cada característica fue evaluada indicando su grado de cobertura. Las características fueron *Evaluación Continua del Riesgo (ECR)*, *Colaboración entre Actores & Flujo de Información (CAF)*, *Cultura & Concienciación en Seguridad (CCS)*, *Habilitación Tecnológica Avanzada (HTA)*, *Cumplimiento Normativo & Políticas (CNP)*, *Interoperabilidad & Integración (II)*, *SBOM & Transparencia de Componentes (STC)*, *Trazabilidad Extremo a Extremo (TEE)*, *Automatización & Integración en CI/CD (AI)*, *Gobernanza & Responsabilidad (GR)*, *Respuesta & Recuperación ante Incidentes (RRI)* y *Código Abierto (CA)*.

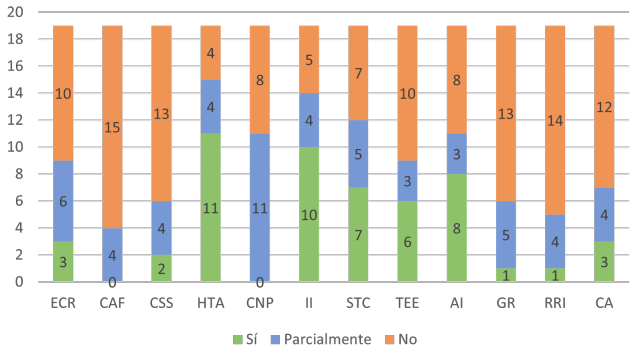


Figura 2: Contribuciones de las características de los estudios primarios.

No obstante, no se ha identificado ningún estudio que considere de manera holística la totalidad de las características definidas como relevantes. Como consecuencia de esta ausencia, se presenta a continuación un marco de trabajo para la trazabilidad, control de riesgos y seguridad en la cadena de suministro de software, que aborda de manera integral cada una de estas características.

IV. MARCO DE TRABAJO PROPUESTO

El marco de trabajo propuesto constituye una solución integral para garantizar la trazabilidad, control de riesgos y seguridad en toda la cadena de suministro de software. Mediante el uso de tecnología blockchain como registro inmutable de evidencias, el marco permite registrar, verificar y auditar todos los componentes de la cadena de suministro, como la validación de dependencias, la generación de artefactos y la ejecución de pruebas, garantizando transparencia, integridad y confianza.

El marco de trabajo propuesto se organiza en cinco etapas funcionales, enlazadas por retroalimentaciones ágiles que permiten la mejora continua. Asimismo, el marco permite la colaboración controlada entre todos los actores y se alinea con las principales normativas de seguridad de la cadena de suministro de software. Las etapas son las siguientes:

- *Etapa 1. Planificación y Adquisición*: identifica, evalúa y selecciona de manera segura todos los componentes, dependencias y proveedores que formarán parte del ecosistema de software.
- *Etapa 2. Desarrollo y Construcción*: crea y ensambla el software mediante prácticas de codificación segura, asegurando la integridad de los entornos y artefactos generados.
- *Etapa 3. Distribución y Despliegue*: empaqueta, almacena y entrega los artefactos de software garantizando su autenticidad, integridad y protección frente a manipulaciones.
- *Etapa 4. Operación y Mantenimiento*: supervisión continua, detección de vulnerabilidades y aplicación de actualizaciones seguras para preservar la resiliencia del software en producción.
- *Etapa 5. Gobernanza y Cumplimiento*: establece las políticas, auditorías y mecanismos de trazabilidad que

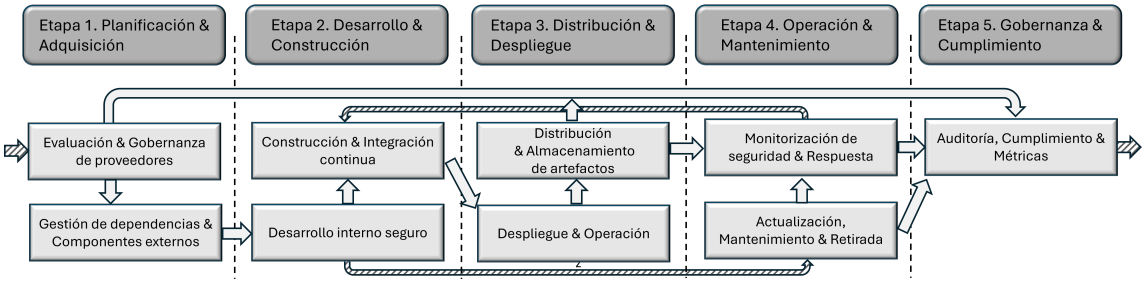


Figura 3: Flujo de los procesos y su correspondencia con las etapas en el marco de trabajo.

sustentan la gestión de riesgos y la rendición de cuentas a lo largo de toda la cadena de suministro.

En conjunto, el marco de trabajo transforma la seguridad de la cadena de suministro como un marco medible y verificable mediante la automatización, facilitando la adopción, interoperabilidad y reducción del tiempo de respuesta ante amenazas.

IV-A. Procesos del marco de trabajo

El marco de trabajo está formado por nueve procesos operativos que se ejecutan de manera secuencial, con retroalimentaciones ágiles DevSecOp entre ellos. La Figura 3 muestra el esquema de flujo de procesos y sus retroalimentaciones. El flujo principal se articula de extremo a extremo de la cadena de suministro de software e incorpora dos retroalimentaciones. La primera retroalimentación, delimitada por flechas continuas, ejecuta de nuevo la cadena de suministro una vez finalizada. La segunda retroalimentación, representada por flechas discontinuas, itera sobre los procesos que abarcan desde la construcción hasta la monitorización del software. Los procesos son los siguientes:

- *Evaluación y Gobernanza de proveedores*: asegura que todos los actores externos que intervienen en la cadena de suministro cumplan con estándares de seguridad y madurez organizativa. Este proceso de control previene insuficiencias de seguridad, incumplimientos normativos y vulnerabilidades en su propia cadena de suministro.
- *Gestión de dependencias y Componentes externos*: selección y adquisición de librerías, marco de trabajos, contenedores y servicios de terceros, tanto de código abierto como propietarios. Los riesgos asociados se mitigan creando y manteniendo un SBOM que documenta versiones, hashes y firmas de cada componente, junto con la evidencia de su procedencia, canal de distribución y responsable de aprobación.
- *Desarrollo interno seguro*: cada commit se firma digitalmente e identifica a su autor, garantizando autenticidad y rendición de cuentas. Los análisis estáticos y las revisiones de código se registran como evidencias de cumplimiento, junto con la documentación de vulnerabilidades detectadas y su resolución.
- *Construcción e Integración continua*: establece mecanismos de aislamiento, verificación y firma digital que aseguran la integridad de los entornos de build y de los pipelines CI/CD. Los riesgos se registran con la configuración, los responsables y artefactos generados, vinculando cada elemento con su código y dependencias.
- *Distribución y Almacenamiento de artefactos*: regulación del almacenamiento y distribución mediante políticas de integridad y control de acceso en repositorios internos o externos. Cada publicación queda asociada a un responsable exigiendo verificación contra él en la descarga o uso.
- *Despliegue y Operación*: asegura que cada versión desplegada sea validada antes de su activación, previniendo configuraciones inseguras o despliegues automáticos no verificados. Se registran versiones, entornos y responsables, junto con controles de integridad.
- *Monitorización de seguridad y Respuesta*: detención de vulnerabilidades, anomalías y ataques en tiempo real. Los riesgos de nuevas vulnerabilidades se mitigan mediante alertas. Cada evento de seguridad genera una traza vulnerabilidad–componente–versión–respuesta anclada en la blockchain.
- *Actualización, Mantenimiento y Retirada*: garantiza que el software en uso permanezca seguro y conforme a los estándares definidos. Se distribuyen parches y actualizaciones con firmas y hashes verificables. Todas las actualizaciones aplicadas, sus validaciones y las versiones retiradas se registran en la blockchain.
- *Auditoría, Cumplimiento y Métricas*: registro centralizado de evidencias que habilita la auditoría integral de toda la cadena de suministro. Se establecen métricas de madurez y riesgo, unificando trazabilidad y cumplimiento.

Cada uno de estos procesos consume y genera artefactos verificables, registrados en la blockchain mediante transacciones firmadas e invocaciones a contratos inteligentes.

IV-B. Capas del marco de trabajo

El marco de trabajo se articula en torno a una arquitectura modular compuesta por cinco capas funcionales y una capa transversal de trazabilidad federada. Cada capa añade un nivel complementario de control y verificación, y la trazabilidad actúa como hilo conductor que interconecta los procesos, las evidencias y los actores involucrados.

La **primera capa, de procesos**, integra los nueve procesos definidos en el marco de trabajo, proporcionando una estructura homogénea y verificable para su gestión. Su función consiste en proporcionar un mapa de referencia del flujo de la cadena de suministro.

La **segunda capa, de controles y requisitos**, traduce los procesos en mecanismos concretos de seguridad, alineados con los principales estándares internacionales. En esta capa se agrupan los controles técnicos y organizativos derivados de normativas y directrices ampliamente aceptadas en la cadena de suministro de software.

La **tercera capa, de evidencias y trazabilidad**, garantiza que cada evento o cambio quede registrado, autenticado y sea verificable. Esta capa permite auditorías automáticas y análisis retrospectivos. La trazabilidad proporciona un contexto verificable que enlaza cada artefacto con su origen, modificación y responsable, facilitando la detección temprana de anomalías o brechas de integridad.

La **cuarta capa, de evaluación y gobernanza del riesgo**, integra los resultados de las anteriores capas para proporcionar una visión de la seguridad actual de la cadena de suministro. Esta capa utiliza métricas y auditorías para correlacionar los riesgos derivados de cada proceso y transformarlos en indicadores de desempeño. De esta manera, se facilita la toma de decisiones estratégicas basada en evidencias y se fomenta la mejora continua de la seguridad.

La **quinta capa, de infraestructura tecnológica y herramientas**, proporciona el soporte operativo y los componentes tecnológicos para la implementación y automatización de los procesos, controles y mecanismos de trazabilidad definidos en las capas superiores. Incluye un sistema CI/CD que permite la ejecución controlada de los flujos de desarrollo y despliegue, y soluciones de monitorización y orquestación de seguridad.

Finalmente, la **capa transversal de trazabilidad y evaluación federada** posibilita la interoperabilidad y el intercambio seguro de información entre los diferentes actores de la cadena de suministro. Este módulo introduce mecanismos, como registros distribuidos, que permiten compartir evidencias y evaluaciones sin exponer información sensible.

IV-C. Actores y atribución de responsabilidades

En el marco de trabajo, los procesos están asociados a actores funcionales, desempeñados tanto por humanos como por sistemas tecnológicos. Esta asignación permite distribuir responsabilidades, reforzar la trazabilidad, facilitar la rendición de cuentas mediante evidencias verificables y mejorar la seguridad. La Tabla II muestra los actores definidos y su responsabilidad principal a lo largo de la cadena de suministro.

Tabla II: Actores y sus funciones en el marco de trabajo.

Actor	Función en la cadena de suministro
Desarrollador	Crea y mantiene código fuente
DevOps / Integrador	Automatiza los procesos de build y despliegue
Proveedor	Suministra librerías, SDKs o contenedores
Gestor de seguridad	Supervisa la postura de seguridad global
Auditor interno	Evalúa cumplimiento y madurez organizacional
Cliente	Consume el producto final
Autoridad / Certificador	Entidad independiente (sectorial o estatal)
Nodo federado	Participante en el ecosistema distribuido

Todos los actores operan bajo esquemas de identidad verificable, y sus interacciones quedan registradas como evidencias inmutables en la blockchain. Esta estructuración funcional permite identificar puntos críticos donde la intervención humana es necesaria y donde la automatización no compromete la toma de decisiones. De este modo, el marco de trabajo no solo implementa trazabilidad técnica, sino que habilita un modelo de gobernanza operativa, auditable y escalable para entornos de desarrollo seguros. La Figura 4 resume la correspondencia entre los roles y la arquitectura por capas del marco de trabajo.

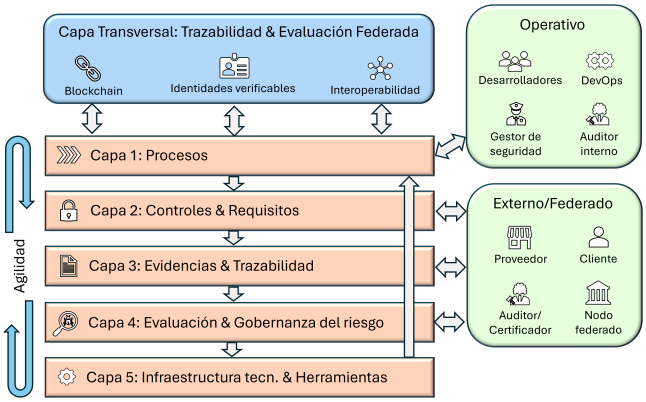


Figura 4: Arquitectura por capas del marco de trabajo y roles implicados.

IV-D. Métricas del marco de trabajo

El marco de trabajo proporciona métricas que permiten monitorizar y gestionar los riegos de manera continua, así como fomentar la seguridad en toda la cadena de suministro. Estas métricas son las siguientes: Cobertura de trazabilidad, Tasa de validación, Participación en decisiones, Índice de descentralización, Tiempo medio de actualización de los contratos, Tiempo de recuperación y Punto de recuperación.

Cabe destacar que el umbral de aceptación y la frecuencia de evaluación de estas métricas deben ajustarse dependiendo del tamaño, nivel de madurez y necesidades de cada organización. Esta flexibilidad garantiza que el marco de trabajo sea aplicable tanto en entornos corporativos complejos como en organizaciones con recursos limitados.

IV-E. Normativas alineadas

El marco de trabajo considera varias de las normativas y estándares internacionales que han emergido en los últimos años para fortalecer la seguridad, trazabilidad y gobernanza de la cadena de suministro de software, especialmente frente a los crecientes riesgos asociados. Algunas de estas normativas y estándares son la *ISO/IEC 27036*, orientada a la gestión de riesgos relacionados con proveedores y provisión de servicios [16]; la *ISO/IEC 20243*, centrada en la prevención de productos y software alterados y maliciosos [17]; y la *NIST SP 800-161 Rev. 1*, que promueve buenas prácticas para la gestión de riesgos [18].

El diseño facilita la adaptación a nuevas disposiciones o actualizaciones de directivas regulatorias. De este modo, el marco de trabajo no solo implementa controles técnicos, sino que también contribuye a consolidar una cultura organizacional orientada a la seguridad y al cumplimiento de la regulación vigente. La frecuencia y el alcance de estas iniciativas pueden adaptarse a las características y necesidades específicas de cada organización.

IV-F. Alineamiento con las características clave

El diseño modular del marco de trabajo permite mapear de manera explícita en qué medida cada etapa contribuye al cumplimiento de las características relevantes que fueron definidas para una cadena de suministro de software. La Tabla III muestra la correspondencia para cada una de las características.

Tabla III: Características incluidas en cada etapa del marco de trabajo (S=Sí; P=Parcial; N=No).

Etapa	ECR	CAF	CSS	HTA	CNP	II	STC	TEE	AI	GR	RRI	CA
Planificación & Adquisición	S	S	P	P	P	P	P	S	P	P	P	S
Desarrollo & Construcción	S	S	P	S	P	S	S	S	S	S	P	S
Distribución & Despliegue	S	S	P	S	P	S	S	S	S	S	P	S
Operación & Mantenimiento	S	S	P	S	P	S	S	S	P	S	S	S
Gobernanza & Cumplimiento	S	S	P	P	P	S	P	S	P	S	S	S

El marco de trabajo propuesto integra el conjunto de características clave identificadas para la cadena de suministro de software. Las características ECR, CAF, TEE y CAA presentan una cobertura completa y transversal en todas las etapas, mientras que HTA, II, STC, AI, GR e RRI muestran una presencia casi total o combinan etapas con cobertura plena y parcial. Finalmente, las características CSS y CNP se consideran parcialmente cubiertas en todas las etapas, actuando como capacidades de soporte de naturaleza más contextual y dependiente de la organización, pero igualmente integradas en el marco de trabajo propuesto.

V. Conclusiones y trabajo futuro

Este trabajo ha presentado un marco de trabajo basado en tecnología blockchain para la trazabilidad, el control de riesgos y la ciberseguridad en la cadena de suministro de software. El marco permite registrar, verificar y auditar cada artefacto y evento dentro de la cadena de suministro, garantizando la trazabilidad y la integridad. Para ello, integra mecanismos de generación de SBOM, validación criptográfica, firma digital y almacenamiento inmutable mediante contratos inteligentes. Asimismo, su diseño modular, la definición de roles, métricas y su interoperabilidad con herramientas CI/CD favorecen la adopción en organizaciones heterogéneas, con distintos niveles de madurez y disponibilidad de recursos.

El diseño de este marco de trabajo es a consecuencia de una revisión sistemática de la literatura en el contexto de la cadena de suministro de software. Dicha revisión ha permitido identificar debilidades recurrentes en las soluciones existentes, entre ellas la falta de trazabilidad de extremo a extremo, la ausencia de una gestión de riesgos continua y la limitada transparencia sobre los componentes utilizados.

Como trabajo futuro, se plantea la validación empírica del marco de trabajo en entornos reales. Asimismo, se explorará la integración de capacidades avanzadas, como la inteligencia artificial (IA) generativa, para la automatización de tareas y la adaptación dinámica

Agradecimientos

Esta publicación no habría sido posible sin el apoyo de los siguientes proyectos: Di4SPDS (PCI2023145980-2), financiado por MCIN/AEI/10.13039/501100011033 Unión Europea (Chist-Era Program); KOSMOSUCLM (PID2024-155363OB-C44) financiado por MCIN/AEI/10.13039/501100011033/ FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la Junta de Comunidades de Castilla La Mancha, el Fondo Europeo de Desarrollo Regional FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; RED2024-154240-T financiado por MICIU/AEI/10.13039/501100011033, SerDigital (PID2020-117244RB-100), financiado por el Ministerio de Ciencia e Innovación a través del proyecto; SIC-SPAIN 4.0 (Safer Internet Centre Spain 4.0), financiado por la Comisión Europea a través del proyecto; DESAFíOCM (TEC-2024/COM-235), financiado por la Comunidad de Madrid; GESMOD (2024/SOLCON-137672), financiado por la Universidad Rey Juan Carlos a través de la convocatoria IMPULSO.

Referencias

[1] H. Stadtler, H. Stadtler, C. Kilger, C. Kilger, H. Meyr, and H. Meyr, *Supply chain management and advanced planning: concepts, models, software, and case studies*. Springer, 2015.

[2] R. He, H. He, Y. Zhang, and M. Zhou, "Automating dependency updates in practice: An exploratory study on github dependabot," *IEEE Transactions on Software Engineering*, vol. 49, no. 8, pp. 4004–4022, 2023.

[3] G. Kim, J. Humble, P. Debois, J. Willis, and N. Forsgren, *The DevOps handbook: How to create world-class agility, reliability, & security in technology organizations*. It Revolution, 2021.

[4] L. Chen, "Continuous delivery: Huge benefits, but challenges too," *IEEE software*, vol. 32, no. 2, pp. 50–54, 2015.

[5] Y. Shen, X. Gao, H. Sun, and Y. Guo, "Understanding vulnerabilities in software supply chains," *Empirical Software Engineering*, vol. 30, no. 1, pp. 1–38, 2025.

[6] C. Crossley, *Software Supply Chain Security: Securing the End-to-end Supply Chain for Software, Firmware, and Hardware*. O'Reilly Media, Inc., 2024.

[7] X. Feng, Y. Gao, J. Wu, and L. Shi, "Research on Security Risk Identification and Evaluation in Open-Source Software Supply Chain," in *Proceedings of the 2025 International Conference on Digital Management and Information Technology*, 2025, pp. 257–266.

[8] G. H. Sagala and D. Őri, "Toward SMEs digital transformation success: a systematic literature review," *Information Systems and e-Business Management*, vol. 22, no. 4, pp. 667–719, 2024.

[9] J. Wen and X. Fan, "Container Image Supply Chain Security Framework Based on Blockchain and Dynamic Analysis," in *2025 Information Science Frontier Forum and the Academic Conference on Information Security and Intelligent Control (ISF)*. IEEE, 2025, pp. 15–19.

[10] M. M. Queiroz, R. Telles, and S. H. Bonilla, "Blockchain and supply chain management integration: a systematic review of the literature," *Supply chain management: An international journal*, vol. 25, no. 2, pp. 241–254, 2020.

[11] D. Blanchard, *Supply chain management best practices*. John Wiley & Sons, 2021.

[12] I. Bashir, *Mastering blockchain*. Packt Publishing Ltd, 2017.

[13] L. Williams, G. Benedetti, S. Hamer, R. Paramitha, I. Rahman, M. Tamanna, G. Tystahl, N. Zahan, P. Morrison, Y. Acar *et al.*, "Research Directions in Software Supply Chain Security," *ACM Transactions on Software Engineering and Methodology*, vol. 34, no. 5, pp. 1–38, 2025.

[14] M. Di Pierro, "What is the blockchain?" *Computing in Science & Engineering*, vol. 19, no. 5, pp. 92–95, 2017.

[15] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele University*, vol. 33, no. 2004, pp. 1–26, 2004.

[16] International Organization for Standardization and International Electrotechnical Commission, "ISO/IEC 27036 (Series) – Cybersecurity – Supplier Relationships," 2021.

[17] International Organization for Standardization, "ISO/IEC 20243 (Series) – Information technology — Open Trusted Technology ProviderTM Standard (O-TTPS)," 2023.

[18] J. Boyens, C. Paulsen, R. Moorthy, N. Bartol, and S. A. Shankles, "Supply chain risk management practices for federal information systems and organizations," *NIST Special publication*, vol. 800, no. 161, p. 32, 2015.

Providing economic incentives to reveal the existence of a CRQC

Roger Sala-Mimó, Jordi Herrera-Joancomartí
Departament d'Enginyeria de la Informació i les Comunicacions
Universitat Autònoma de Barcelona (UAB)
Roger.Sala@uab.cat, Jordi.Herrera@uab.cat

***Abstract*—A Cryptographically Relevant Quantum Computer (CRQC) is a quantum computer capable of running algorithms at a scale large enough to break widely used classical public-key cryptography. It is difficult to assess when a CRQC will emerge because reliable information in this area is scarce and often distorted. Public announcements from companies are hard to validate, and there is a clear incentive to exaggerate progress to attract investment. At the same time, governments are unlikely to disclose the true state of their quantum capabilities for national security reasons. To mitigate this situation, we propose the design and implementation of a honeypot for quantum computers, a standard-like Bitcoin address in which some capital may be deposited as an economic incentive for a CRQC to reveal itself. The Bitcoin honeypot address is deliberately weakened to a smaller elliptic curve key. The protocol distributes key generation among multiple participants, ensuring that no single actor can gain an unfair advantage. If the funds are ever retrieved, it would strongly indicate the presence of a CRQC capable of solving the Elliptic Curve Discrete Logarithm Problem in the smaller curve, still providing some reaction time for larger curves. The system is implemented in Python and deployed on the Bitcoin testnet, ensuring compatibility, correctness, robustness, and public verifiability.**

***Index Terms*—Bitcoin, Honeypot, Quantum Computer, Distributed Protocol, Elliptic Curve.**

I. Introduction

Recent quantum computing advances [1] challenge widely used cryptographic algorithms. The main difference between classical and quantum computers is that quantum computers use qubits instead of bits, enabling them to store and process exponentially more information. Although quantum computers can represent binary outcomes upon measurement, qubits handle and manipulate information in fundamentally different ways compared to classical systems.

Qubits can exist in superposition, representing multiple states simultaneously. This property allows quantum computers to exponentially increase their ability to process and store information compared to classical bits, which can only represent one possibility at a time. Quantum circuits, made up of logical qubits, can process enormous datasets simultaneously, improving computational efficiency by many orders of magnitude for specific problems [2]. This advancement could significantly impact various fields, including cryptography. For instance, Elliptic Curve Digital Signature Algorithm (ECDSA) [3] and Schnorr [3] signatures could be broken by a Cryptographically Relevant Quantum Computer (CRQC) running Shor's algorithm [4], which could potentially solve the Elliptic Curve Discrete Logarithm Problem (ECDLP) [3] that secures them.

Assessing CRQC emergence is difficult due to scarce and often distorted information. Public announcements from companies are hard to validate, and there is a clear incentive to exaggerate progress to attract investment, attention, or government contracts. At the same time, governments are unlikely to disclose the true state of their quantum capabilities, both for national security reasons and to avoid revealing strategic advantages. As a result, most available news provides an incomplete or unreliable picture, making any prediction about the arrival of a CRQC highly uncertain.

To overcome such uncertainty, this research designs a mechanism that provides an economic incentive to a CRQC operator to reveal its existence. This incentive is provided in the form of a bitcoin economic bounty, deposited in a standard Bitcoin address. By implementing such a honeypot Bitcoin address, we intent to provide an early warning system for quantum advancements, enabling the development of preventive measures. If a CRQC breaks the honeypot keys, it would signal imminent risk to standard cryptography protocols, prompting a response.

The contribution of the paper is the design and implementation of a distributed protocol that creates a standard Bitcoin address, that by default has 128 bits of security, but with the special property that its security is reduced to 96 bits. The distributed protocol ensures that no participant of the protocol gains any advantage from their involvement in the address creation to retrieve the funds of the address, forcing any movement of funds within the honeypot can only result from an external entity with access to a CRQC.

The rest of the paper is organized as follows. Section II provides the state of the art on quantum computing and weakened elliptic curves. In Section III, the proposed protocol and its phases are introduced. Section IV analyzes how the protocol satisfies the required properties. The protocol implementation and tests are described in Section V. Finally, Section VI presents the conclusions and future work.

II. State of the Art

A. Quantum Computers

Quantum computing has made significant advancements over the past decade, but current systems remain far from being capable of breaking modern cryptographic algorithms such as Bitcoin's, that is based on the ECDLP on `secp256k1`. This section reviews the most relevant quantum computing platforms, focusing on their current qubit capacities, error rates, and roadmaps towards achieving large-scale, fault-tolerant quantum computation [5].

IBM is one of the leading companies in quantum computing research. As of 2024, IBM's *Heron* processor reached 133 physical superconducting qubits [6]. Their development

roadmap outlines several key milestones and beyond 2033 their goal is to reach 2,000 logical qubits with full error correction and one billion gate operations.

Google has made significant advancements in quantum computing with its *Willow* processor, featuring 105 qubits [7]. Google's quantum computing roadmap is divided into six milestones and the 6th milestone represents their long-term goal of 1,000,000 physical qubits, achieving a logical error rate of 10^{-13} and incorporating error correction.

The University of Science and Technology of China has made notable advancements in quantum computing with the *Zuchongzhi 3.0* processor, which features 105 transmon qubits. These qubits are constructed in a way to reduce sensitivity to noise [8]. There are no publicly available detailed roadmaps for China's quantum computing development.

D-Wave has developed the *Advantage* quantum computer, which is the world's largest quantum annealing system, featuring over 5,000 qubits. However, D-Wave's qubits are not universal like those in IBM, Google or Microsoft's systems. Instead, they are specialized for specific types of problems, such as optimization tasks, and are not suitable for general quantum algorithms or gate-based quantum computing. Unlike companies such as IBM or Google, D-Wave does not have a clearly defined roadmap with specific numerical targets for increasing qubit counts or achieving milestones in error correction [9].

Microsoft has introduced the *Majorana 1* quantum processing unit (QPU), the world's first quantum processor powered by topological qubits. This architecture is designed to scale up to a million qubits on a single chip. Topological qubits are expected to be more robust against noise and error, which is a significant challenge in current quantum computing systems. Microsoft focuses on developing its topological qubit technology, which they believe will enable large-scale, fault-tolerant quantum computation in the future [10].

B. Estimation of Qubits to Solve the ECDLP on `secp256k1`

The number of qubits required to solve the ECDLP on `secp256k1` remains an active area of research. Below, we summarize recent estimates from various studies [11].

A 2022 study from the University of Sussex estimates that breaking `secp256k1` within 1 to 8 hours would require between 13 and 300 million physical qubits [12]. Similarly, Webber et al. [13] estimate that approximately 317 million physical qubits are needed for a one-hour attack. If the attack duration is extended to one day, this number drops to about 13 million physical qubits.

Kudelski Security suggests that, at least, 2,619 logical qubits and roughly 7.43 million physical qubits would be needed to complete an attack within 24 hours under similar error rates [14]. An independent cryptographic analysis estimates between 1,500 and 3,000 logical qubits could solve the ECDLP in under 10 minutes under ideal conditions, which corresponds to millions of physical qubits when accounting for error correction overhead [15].

In summary, while estimates vary widely, the consensus is that quantum threats to Bitcoin's security via solving the ECDLP on `secp256k1` are still many technological generations away.

C. Smaller Elliptic Curves

To deliberately weaken the cryptographic strength of Bitcoin without altering its transaction structure, one approach involves using a smaller elliptic curve. Several standards offer such smaller curves, including those by Brainpool, Certicom (SEC 2), and NIST.

The NIST standard (FIPS PUB 186-4, 2013) [3] defines five elliptic curves over $\mathbb{Z}_p$ with prime sizes of 192, 224, 256, 384, and 521 bits. Curves are named using the prefix `P-`, indicating their definition over prime fields. These curves were generated using a verifiably random algorithm with SHA-1 as the hash function. Examples of smaller curves in this set include `P-192` and `P-224`.

The Brainpool standard [3], published in 2005, defines a set of elliptic curves known as the Brainpool Standard Curves and Curve Generation. It includes seven curves, among which `brainpoolP160r1`, `brainpoolP192r1`, and `brainpoolP224r1` are of particular interest for reduced security configurations. Brainpool curve names follow a consistent format: the prefix `brainpool`, followed by a `P` indicating the use of a prime field $\mathbb{Z}_p$, a number specifying the prime size, a letter `r` denoting a random curve, and a sequence number.

Certicom's SEC 2 standard (version 2.0, 2010) [3] describes eight elliptic curves over $\mathbb{Z}_p$ using prime field sizes of 192, 224, 256, 384, and 521 bits. For each size (except 384 and 521), two types of curves are offered: verifiably pseudo-random (denoted `r`) and Koblitz curves (denoted `k`). Examples of smaller ones include `secp192k1`, `secp192r1`, `secp224k1`, and `secp224r1`.

The Koblitz curves over $\mathbb{Z}_p$ are defined using parameters that allow efficient point doubling. The ones included in this standard have been selected by iteratively searching for efficient parameters until a curve of prime order is found.

III. A Bitcoin Honeypot Address Protocol

The aim of the Bitcoin Honeypot Address Protocol (BHAP) is to construct a Bitcoin honeypot for quantum computers that could serve as an alert for when a quantum computer is capable of solving the ECDLP on `secp192r1`, specifically using Shor's algorithm, meaning it is approaching the capability to solve the ECDLP on `secp256k1`, the elliptic curve Bitcoin uses. As we will describe in detail later on, to provide the desired security properties, the BHAP is a distributed protocol performed by n participants. For efficiency reasons, we present here the simplified version that uses a coordinator.

The honeypot address generated by the BHAP should satisfy the following properties:

1) **Compatibility:** The honeypot should be implemented in a way that appears as a standard Taproot address fully compatible with Bitcoin but having the security level of `secp192r1`.
2) **Correctness:** Even with up to k (with $k < n$) malicious participants in the BHAP, an attacker with a quantum computer with capacity to break the ECDLP over `secp192r1` should be able to retrieve the funds in the honeypot address.
3) **Robustness:** The private key of the honeypot address is completely unknown providing that the BHAP has a single honest participant.

4) **Verifiability:** All the data related to the honeypot address creation must be publicly verifiable.

As we already mention, the BHAP is a distributed protocol with n participants plus a coordinator. This protocol assumes an authentic communication channel for correct execution.

The key naming convention used in this protocol is the following:

- (P_b, d_b): (public, private) keys using the `secp256k1` curve (Bitcoin standard).
- (P_w, d_w): (public, private) keys using the `secp192r1` curve (weakened for quantum feasibility).

The subindex b and w stand for *bitcoin* and *weak* respectively.

A. BHAP description

1) Phase 1: Key Pair Generation: In this first phase of the protocol, each participant $i \in (1, \cdots, n)$ independently generates a random Bitcoin private key, d_{b_i} and then calculates its corresponding Bitcoin public key, P_{b_i} using the `secp256k1` curve.

After the key pair generation process, all $i \in (1, \cdots, n)$ participants need to share their public key P_{b_i} with the coordinator in order to proceed to the next step of the protocol.

2) Phase 2: Public Key Aggregation: In this second phase, the coordinator receives all participants' public keys P_{b_i} for $i \in (1, \cdots, n)$, and aggregates them into a single public key $P_b = \mathsf{KeyAgg}_{i=1}^{n}(P_{b_i})$ via elliptic curve point addition [3].

3) Phase 3: Weaker Key Generation: After obtaining the aggregated public key $P_b = \mathsf{KeyAgg}_{i=1}^{n}(P_{b_i})$, the coordinator needs to generate a `secp192r1` public key.

To generate this weaker key, the coordinator hashes with SHA-256 the aggregated public key P_b, truncates the result to the most significant 24 bytes, and uses these 24 bytes as the x-coordinate to try to derive a point on the `secp192r1` curve, if no valid point is found, the coordinator hashes again and repeats the process until a valid point on the curve is found. Upon completion, the coordinator obtains the weak public key on `secp192r1`, P_w.

After the weaker key is generated, the coordinator shares P_b, P_w and the number of hashes performed to derive P_w from P_b with all participants $i \in (1, \cdots, n)$.

4) Phase 4: ECIES encryption of d_{b_i} with P_w: Once each participant $i \in (1, \cdots, n)$ receives P_b, P_w, and the number of hashes. They must encrypt their d_{b_i} with ECIES [3] and AES-CBC using P_w.

Participants will also need to perform a Zero-Knowledge Proof [3] to prove that the ciphertext corresponds to a private key d_{b_i} which in turn corresponds to the public key P_{b_i}.

In this protocol, the encryption method used is ECIES [3], as it is the standard for elliptic curve cryptography, typically used in combination with AES. AES will operate in CBC mode, as it produces fewer outputs than GCM, which includes an authentication tag, which is not required in this protocol, since an authentic communication channel is assumed and minimizing the amount of data to be stored is prioritized.

After the ECIES encryption, the participants $i \in (1, \cdots, n)$ will obtain a ciphertext $c_i = \mathsf{Enc}_{P_w}(d_{b_i})$, an initialization vector IV_i, and an ephemeral public key P_{e_i}. All these outputs, $E_i = (P_{b_i}, c_i, IV_i, P_{e_i})$, for $i \in (1, \cdots n)$, must be sent to the coordinator to include them into the honeypot commitment.

5) Phase 5: Honeypot funding and data commitment: This is the final step of BHAP. In this phase, the coordinator receives all participants' outputs $E_i = (P_{b_i}, c_i, IV_i, P_{e_i})$ for $i \in (1, \cdots, n)$ and creates a commitment with all the honeypot data. This includes the aggregated public key P_b, the weak public key P_w, the number of hashes performed to derive P_w from P_b, and the outputs $E_i = (P_{b_i}, c_i, IV_i, P_{e_i})$ for each participant $i \in (1, \cdots n)$. All this data is hashed and tagged to produce a taproot tweak, which is then used to modify P_b resulting in the tweaked public key P_{bt} [16], which is then converted into the final honeypot address H_A, which will be the address where the funds will be sent.

The coordinator will then create the initial funding transaction, including an `OP_RETURN` [17] output containing a link to a IPFS file with all relevant information, details, and source code.

IV. BHAP properties analysis

In this section, we analyze how the described protocol ensures the properties described at Section III.

1) Compatibilty: In Phase 4, the Compatibility property is satisfied because when the ECIES encrypted values of d_{b_i} are shared, they are protected by 96 bits of security (`secp192r1`) instead of the 128 bits provided by `secp256k1`, while P_b remains fully compatible with Bitcoin while using a weaker security than Bitcoin's. The address generated in Phase 5 also satisfies the Compatibility property, as it encapsulates Bitcoin private keys d_{b_i} using a weaker security key P_w while maintaining compatibility with Bitcoin.

2) Correctness: In Phase 4 the Correctness property will also be satisfied, preventing any malicious participant from submitting a ciphertext unrelated to their encrypted d_{b_i} corresponding to their public key P_{b_i}. The content of what all participants encrypt must be the private key corresponding to their public key, ensuring that a quantum computer can decrypt all of them and redeem the funds. This can be done with Zero-Knowledge Proofs [3], but it is beyond the scope of this paper, and it is not included in the implementation provided with the paper.

3) Robustness: In Phase 2, the aggregated public key P_b that includes all participants public keys P_{b_i}, satisfies the Robustness property, such that, if only one participant is honest, any participant will have zero advantage in front of the others to obtain the aggregated private key d_b and move the honeypot funds.

4) Verifiability: As mentioned in the Verifiability property, all data related to the honeypot must be verifiable by anyone. This requirement is met, as all relevant information is shared by both the participants and the coordinator. Any participant can obtain all public keys P_{b_i} for $i \in (1, \cdots, n)$ and perform the key aggregation themselves to verify that the aggregated public key P_b and weaker public key P_w provided by the coordinator are correct.

Subsequently, the coordinator in Phase 5 can verify all Zero-Knowledge Proofs submitted by participants, along with the ECIES outputs E_i, to confirm their validity. The `secp192r1` public key will be the one used for the ECIES encryption and, in fact, the one that the quantum computer must break by solving the ECDLP using Shor's algorithm.

This public key must be generated or derived in a manner that satisfies the Verifiability property ensuring that the `secp192r1` public key is not arbitrarily chosen or hidden, but instead has a verifiable relationship with the previous data.

Once the coordinator generates the honeypot's taproot address H_A, all participants can repeat the process to verify that the result matches the coordinator's.

Similarly, the attacker can repeat all verification steps to ensure they are not being misled before attempting to recover the private key d_w corresponding to P_w.

All the data related to the honeypot must be verifiable by anyone step by step without hidden or opaque components. All this data will be committed into a taproot tweak, and the `OP_RETURN` of the funding transaction will include a link to a IPFS file with all the information and source code.

A. Redemption of the honeypot funds

BHAP properties ensure that if only one participant is honest, no participant will have an advantage over the others, nor will non-participants have any advantage in obtaining the aggregated private key d_b and move the honeypot funds. Thus, if the funds are moved, it indicates a high probability of quantum computer involvement, the discovery of a new algorithm to solve the ECDLP with a standard computer, or even someone extremely lucky who managed to compute d_b using existing algorithms. Although it would be impossible to determine which occurred, we can be confident that no participant had an advantage over the others, which is the key point. With a sufficiently large number of participants, it becomes highly probable that at least one is honest.

In a hypothetical future scenario where a Quantum Computer is capable of solving the ECDLP on `secp192r1`, its owner(s) may discover this honeypot, will follow the `OP_RETURN` link to the associated IPFS file and verify all the provided data. After verification, they will run Shor's algorithm on their Quantum Computer to recover the private key d_w corresponding to P_w. Once d_w is obtained, each participant's ciphertext c_i for $i \in (1, \cdots, n)$ can be decrypted using ECIES decryption giving $d_{b_i} = \mathsf{Dec}_{d_w}(E_i)$. All recovered private keys d_{b_i} for $i \in (1, \cdots, n)$ are then aggregated to reconstruct the final private key $d_b = \mathsf{KeyAgg}_{i=1}^{n}(d_{b_i})$.

Once d_b is reconstructed, it can be imported into a wallet to spend all the funds from the honeypot.

V. BHAP and attack Implementation

The BHAP protocol was implemented in Python through five scripts that perform the functions described previously. The implementation utilizes several existing Python libraries, which will be detailed in the following sections. During implementation, various potential threats and adversarial scenarios were identified. To address these issues, multiple tests were conducted, which will be discussed in a later section. All source code is publicly available on GitHub: `https://github.com/rogersala21/TFG-BHQC`.

In the following sections, the code is explained from the perspective of each role. For clarification, the naming convention for each script is `xy_NameOfTheScript` where `x` is a letter representing the role—participant (`p`), coordinator (`c`), or attacker (`a`)—and `y` is the protocol step number (`1-5`).

A. Participants

1) Key Pair Generator: This is the first step of the implementation and corresponds to Phase 1 of the protocol. The script is named `p1_KeyPairGenerator.py`. All participants $i \in (1, \cdots, n)$ must execute this script to generate their respective private key d_{b_i} and public key P_{b_i}. The script outputs two text files—one for d_{b_i} and another for P_{b_i}—which are stored in `"../outputs/participant/keys"`. A unique identifier is appended to each filename to facilitate the coordinator's management of the received public keys.

The script utilizes the `secrets` [18] library to generate the private key and `bitcoinutils` [19] to derive the corresponding public key.

2) Key encryption: This corresponds to the third step of the implementation and to Phase 4 of the protocol. The script is named `p3_ReducedSecurityKeyCypher.py`. Upon receiving the `"aggregation_output.txt"` file, participants must save it in `"../outputs/participant"` and execute this script. The script first verifies the contents of `"aggregation_output.txt"` by extracting the aggregated public key P_b and the number of hashes, performing the same derivation process as the coordinator, and comparing the result with the received P_w. Once this verification succeeds, the script proceeds to encrypt d_{b_i} using P_w with ECIES and AES-CBC. The resulting output $E_i = (P_{b_i}, c_i, IV_i, P_{e_i})$ for $i \in (1, \cdots n)$, is saved in `"../outputs/participant/ecies_output"`. The unique identifier used for the public and private key files is appended to these outputs to facilitate the work of the coordinator.

This script uses the `tinyec` [20] library for elliptic curve operations on `secp192r1`, and `cryptography` [21] for the ECIES encryption.

B. Coordinator

1) Public Key Aggregator: This corresponds to the second and third steps of the implementation and to Phase 2 and Phase 3 of the protocol. The script is named `c2_PublicKeyAggregator.py`. When the coordinator receives all participants' public keys P_{b_i}, they are saved in `"../outputs/coordinator/key_agg_input"`. The coordinator executes the script, which aggregates all public keys P_{b_i} within this directory. The script then hashes the aggregated public key P_{b_i} using SHA-256, truncates the hash to the most significant 24 bytes, and attempts to derive a valid point on the `secp192r1` curve using these bytes as the x-coordinate. If no valid point is found, the hash process repeats until a valid point on the curve is derived. Once completed, the script saves the results—including the aggregated public key P_b, the number of hashes performed, and the weakened public key P_w—to `"../outputs/coordinator/key_agg_output/aggregation_output.txt"`.

The curve `secp192r1` was chosen because it is the smallest curve in the Certicom SEC 2 standard, thereby offering the lowest level of security, and because it was the one for which we found Python libraries with the necessary functionality. The choice of `secp192r1` over `secp192k1` was also influenced by the fact that the libraries we found most suitable supported `secp192r1`.

This script utilizes the `coincurve` [22] library for key aggregation, `hashlib` for hashing, `tinyec` [20] for elliptic curve operations on `secp192r1`, and `cryptography` [21] for key serialization and formatting.

2) Honeypot Commitment: This corresponds to the fourth step of the implementation and to Phase 5 of the protocol. The script is named `c4_HoneypotCommitment.py`. Upon receiving all participants' ECIES outputs, the coordinator saves them in `"../outputs/coordinator/honeypot_commitment"`. The script then creates a commitment by combining data from `"../outputs/coordinator/key_agg_output"` and `"../outputs/coordinator/honeypot_commitment"`.This commitment includes the aggregated public key P_b, the number of hashes performed, the weakened public key P_w, and the set of participant outputs $E_i = (P_{b_i}, c_i, IV_i, P_{e_i})$ for $i \in (1, \cdots n)$. The commitment is tagged and hashed to generate a tweak value used to tweak P_b, producing the tweaked public key P_{bt}. This key is then converted into the actual Taproot address of the honeypot, denoted as H_A. The address H_A is saved in `"../outputs/coordinator/honeypot_address.txt"`. Afterward, the script prompts the coordinator to create a funding transaction embedding an `OP_RETURN` output, which includes a link to a IPFS file containing all relevant information. To create this transaction, the coordinator will need to provide a UTXO as the input and the private key in WIF format to sign the transaction. The script also provides a direct link to mempool.space with the signed transaction ready for broadcast.

This script relies on the `bitcoinutils` [19] library for creating the commitment, tweaking the public key, generating the honeypot address, and constructing the `OP_RETURN` transaction.

C. Attacker

This is the final step of the protocol, corresponding to the Redemption phase of the honeypot funds in a hypothetical post-quantum scenario. The script is named `a5_WalletReadyKeyRetriever.py`. In this hypothetical scenario, the owner of a quantum computer will access the honeypot information through the IPFS file linked in the `OP_RETURN` funding transaction. The attacker then executes the script, which guides them step by step through the process. The script first instructs the attacker to save the participants' public keys P_{b_i}, for $i \in (1, \cdots n)$ into `"../outputs/coordinator/key_agg_input"` and all ECIES outputs into `"../outputs/coordinator/honeypot_commitment"`. It also prompts the attacker to enter the honeypot address H_A. Using these inputs, the script internally runs `c2_PublicKeyAggregator.py` and `c4_HoneypotCommitment.py` to recompute the aggregated and weakened public keys, and then verifies that the user-entered address H_A matches the address computed from the honeypot data. If the addresses coincide, the script asks for the private key d_w corresponding to the weakened public key P_w to be saved in `"../attacker/privatekey.txt"` in PEM format (SEC1, unencrypted). It verifies that this private key indeed generates P_w. Next, the script iterates over all ECIES output files, decrypts each ciphertext to recover the private keys d_{b_i}, and aggregates them to reconstruct the final private key d_b. The script then applies the same tweak operation as the coordinator to obtain the tweaked private key d_{bt}, which corresponds to the honeypot address H_A. Finally, it confirms that H_A derived from d_{bt} and the corresponding public key P_{bt} matches the expected honeypot address. Upon successful verification, the script generates a wallet descriptor ready to import into Bitcoin Core [23] and saves it to `"../attacker/bitcoin_core_import.txt"` for easy copy-pasting.

This script uses the same libraries as the other scripts for cryptographic operations. Additionally, it utilizes `descriptor.py` [24] located in `src/modules`, which handles the creation of the wallet descriptor compatible with Bitcoin Core.

D. Tests

Before implementing the protocol, several potential weak points in the code were identified and thoroughly tested. For the key aggregation process, we generated and aggregated key pairs and three-key sets and performed a signature test to check the correctness of the key aggregation at a large scale, giving a success rate of 100%. (This test can be found in `"/tests/test_key_aggregation.py"`) Another similar test was performed, but with an added final addition of all key pairs together to get an aggregated key from 2000 keys, and all three-key sets together to get an aggregated key from 3000 keys. The results also yielded a 100% success rate. (This test can be found in `"/tests/test_key_aggregation_2.py"`)

To test the derivation of P_b into P_w and verify the correctness of the code, we performed many iterations to check how many hashes were needed to get a valid P_w and the maximum number of hashes required. The largest test executed was for 15,000,000 samples and took nearly 2 hours to complete. This test generates a random P_b and performs the entire derivation process for each iteration. All the performed tests had a success rate of 100% (This test can be found in `"/tests/test_hash_to_secp192_pubkey.py"`).

Some other possible code failures that were considered include the possibility that when aggregating all the keys, the resulting aggregated key would be the point at infinity $\mathcal{O}$, meaning the private key would be zero. Another similar possibility is if the aggregated public key is the generator point G, meaning the private key is 1, which leads to the same issue as before. These two possibilities are extremely unlikely to happen, but if that were the case, the entire protocol must be restarted from the very beginning.

To test this scenario, `"/tests/invalidpoint finder.py"` was created; it is the same code as `"/tests/test_hash_to_secp192_pubkey.py"` but designed to find invalid points on the `secp192r1` curve, to verify that the process works correctly when it fails to find a valid point. The only difference is that it forces `is_quadratic_residue` to be true, so it will always try to create a Public Key, even if the `x_candidate` is not valid. After running this test, we found that the library returns an error if the point is invalid, if the point is the point at infinity $\mathcal{O}$, or if the point is the generator G; in the latter

case, it directly throws an exception saying that the point cannot be the same as the generator.

To ensure robustness, we added two functions to `c2_PublicKeyAggregator.py` that verify the aggregated point is different from the point at infinity $\mathcal{O}$ and also different from the generator point G.

The final test performed was to verify whether the entire protocol implementation was correct. As we do not have a quantum computer, we added a temporary bypass in every script where P_w and d_w were used, and replaced them with a P_w for which we knew d_w.

VI. Conclusions and future work

This paper addresses the uncertainty around the emergence of a CRQC and its potential impact on breaking ECDLP. Because reliable information about a quantum computer is limited, we aim to create an economic incentive for a CRQC to reveal its capabilities. To do so, we introduce a Bitcoin honeypot that intentionally relies on a smaller elliptic curve while remaining fully compatible with the Bitcoin protocol.

We design a distributed protocol that lets multiple participants create a Taproot address derived from an aggregated public key. Each participant encrypts their private key using ECIES under a weakened key derived from the aggregated one on the `secp192r1` curve.

The protocol satisfies four properties: compatibility, since the address is indistinguishable from a standard Taproot one; correctness, as anyone who can break the ECDLP on `secp192r1` can recover all keys and redeem the honeypot funds; robustness, ensuring no participant gains an advantage from being part of the protocol; and verifiability, enabling public verification of all the data and steps performed. These properties make the honeypot a practical mechanism for detecting early quantum capabilities.

Although the proposed protocol provides an approach to building a Bitcoin honeypot for quantum computers, several aspects can be improved or extended in future work. The main weakness of the proposed protocol is that any dishonest participant could encrypt arbitrary data in the `ECIES encryption` step instead of their actual private key d_{b_i}. This would break the correctness property, rendering the honeypot permanently inaccessible, or at least inaccessible under the assumptions of this protocol—thus undermining its entire purpose.

To mitigate this risk, a future improvement would involve using a Zero-Knowledge proof to verify that the encrypted content is indeed the participant's private key d_{b_i} corresponding to their public key P_{b_i} without revealing the private key itself. This would ensure the integrity of the encryption process without compromising the keys, thereby preserving the correctness property.

Acknowledgements

This work is funded by the projects PID2021-125962OB-C33 SECURING/NET and PID2024-156914OB-C43 SAFE-BLOCKCHAIN, funded by the Ministerio de Ciencia e Innovación, la Agencia Estatal de Investigación and the European Regional Development Fund (ERDF); the DANGER Strategic Project of Cybersecurity C062/23, funded by the Spanish National Institute of Cybersecurity through the European Union - NextGenerationEU and the Recovery, Transformation and Resilience Plan; and the Catalan Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) grant SGR2021-00643.

References

[1] Veritasium, "How Quantum Computers Break The Internet... Starting Now," Accessed: 30 June 2025. [Online]. Available: https://www.youtube.com/watch?v=-UrdExQW0cs

[2] IBM, "What Is Quantum Computing? | IBM," Accessed: 30 June 2025. [Online]. Available: https://www.ibm.com/think/topics/quantum-computing

[3] J. Herrera-Joancomartí and C. Pérez-Sol *La criptografia que et cal saber*, J. Herrera-Joancomartí and C. Pérez-Sol Eds., 2023. [Online]. Available: https://criptografia.cat/

[4] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, p. 1484–1509, Oct. 1997.

[5] D. Willsch, P. Hanussek, G. Hoever, M. Willsch, F. Jin, H. D. Raedt, and K. Michielsen, "The state of factoring on quantum computers," 2025. [Online]. Available: https://arxiv.org/abs/2410.14397

[6] "Ibm quantum roadmap," Accessed: 30 June 2025. [Online]. Available: https://www.ibm.com/roadmaps/quantum/

[7] "Google Quantum AI," Accessed: 30 June 2025. [Online]. Available: https://quantumai.google/

[8] "Chinese quantum processor is 1 quadrillion times faster than the best supercomputer — and it rivals Google's breakthrough Willow chip," Accessed: 30 June 2025. [Online]. Available: https://www.livescience.com/technology/computing/china-achieves-quantum-supremacy-claim-with-new-chip-1-quadrillion-times-faster-than-the-most-powerful-supercomputers

[9] "The Advantage™ Quantum Computer | D-Wave," Accessed: 30 June 2025. [Online]. Available: https://www.dwavequantum.com/solutions-and-products/systems/

[10] N. C. M. Latinoamérica, "Microsoft presenta Majorana 1, el primer procesador cuántico del mundo impulsado por qubits topológicos," Accessed: 30 June 2025. [Online]. Available: https://news.microsoft.com/source/latam/noticias-de-microsoft/microsoft-presenta-majorana-1-el-primer-procesador-cuantico-del-mundo-impulsado-por-qubits-topologicos/

[11] C. Gidney and M. Ekerå, "How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits," *Quantum*, vol. 5, p. 433, Apr. 2021.

[12] "Will Quantum Computing Break Bitcoin? | River Learn - Bitcoin Technology," Accessed: 30 June 2025. [Online]. Available: https://river.com/learn/will-quantum-computing-break-bitcoin/

[13] M. Webber, V. Elfving, S. Weidt, and W. K. Hensinger, "The impact of hardware specifications on reaching quantum advantage in the fault tolerant regime," *AVS Quantum Science*, vol. 4, no. 1, Jan. 2022.

[14] T. Gagliardoni, "Quantum Attack Resource Estimate: Using Shor's Algorithm to Break RSA vs DH/DSA VS ECC," Aug. 2021. [Online]. Available: https://research.kudelskisecurity.com/2021/08/24/quantum-attack-resource-estimate-using-shors-algorithm-to-break-rsa-vs-dh-dsa-vs-ecc/

[15] S. Holmes and L. Chen, "Assessment of quantum threat to bitcoin and derived cryptocurrencies," Cryptology ePrint Archive, Paper 2021/967, 2021. [Online]. Available: https://eprint.iacr.org/2021/967

[16] andozw, "What does it mean to tweak a public key?" Accessed: 30 June 2025. [Online]. Available: https://bitcoin.stackexchange.com/q/113232

[17] ChristianOConnor, "How do I use python-bitcoin-utils to add an OP_RETURN message into a bitcoin transaction?" Accessed: 30 June 2025. [Online]. Available: https://bitcoin.stackexchange.com/q/117930

[18] "PEP 506 – Adding A Secrets Module To The Standard Library | peps.python.org," Accessed: 30 June 2025. [Online]. Available: https://peps.python.org/pep-0506/

[19] "bitcoin-utils: Bitcoin utility functions," Accessed: 30 June 2025. [Online]. Available: https://github.com/karask/python-bitcoin-utils

[20] alexmgr, "alexmgr/tinyec," Accessed: 30 June 2025. [Online]. Available: https://github.com/alexmgr/tinyec

[21] "pyca/cryptography," Accessed: 30 June 2025. [Online]. Available: https://github.com/pyca/cryptography

[22] "coincurve: Safest and fastest Python library for secp256k1 elliptic curve operations," Accessed: 30 June 2025. [Online]. Available: https://ofek.dev/coincurve/

[23] "Bitcoin Core," Accessed: 30 June 2025. [Online]. Available: https://bitcoincore.org/

[24] bitcoin, "bitcoin/../test_framework/descriptors.py at master · bitcoin/bitcoin," Accessed: 30 June 2025. [Online]. Available: https://github.com/bitcoin/bitcoin/blob/master/test/functional/test_framework/descriptors.py

Análisis de mecanismos de privacidad basados en la perturbación de los enlaces en grafos

Víctor Rubio-Jornet
Universitat Politècnica de Catalunya
victor.rubio.jornet@upc.edu

Javier Parra-Arnau
Universitat Politècnica de Catalunya
javier.parra@upc.edu

Jordi Forné
Universitat Politècnica de Catalunya
jordi.forne@upc.edu

***Resumen*—El uso de los grafos se ha extendido debido a las nuevas aplicaciones que se les han dado en disciplinas como biología, ingeniería de redes, análisis de datos, etc. De la misma manera se ha incrementado la necesidad de proteger los datos – muchas veces sensibles – que se incluyen en ellos. Pese a haber diversas maneras de privatizar los grafos, este artículo se centra en aquellas soluciones que enmascaran los enlaces de los grafos usando ϵ-Differential Privacy. Específicamente, nuestro trabajo clasifica y analiza las soluciones existentes, identificando fortalezas y desventajas de cada una y relacionando una comparación exhaustiva. De las soluciones recopiladas se ha observado que ninguna ha seleccionado unos grafos suficientemente diversos durante la evaluación. Consecuentemente, no se menciona la capacidad de cada solución de garantizar la privacidad para aquellos conjuntos de datos omitidos. Por esta razón, hemos definido un conjunto de mejores prácticas al seleccionar los grafos para evaluar la utilidad.**

***Index Terms*—grafos, edge-DP, privacidad diferencial**

I. Introducción

Los grafos se usan, entre otras disciplinas, para modelar la relaciones interpersonales en redes sociales o las conexiones entre dispositivos en la redes de comunicación. Por eso, el incremento en el uso de las redes sociales y redes telemáticas ha hecho que las investigaciones relacionadas con los grafos hayan incrementado de la misma manera. El contenido de estos datos puede ser de alta sensibilidad, como en redes de telecomunicación críticas. Además, los datos provenientes de redes sociales pueden revelar mucha información sobre los usuarios que los componen. Por ejemplo, en Jernigan y Mistree [1] consiguen identificar la orientación sexual de estudiantes del Massachussetts Institue of Technology (MIT) a través de la red social Facebook mediante el análisis de las relaciones interpersonales.

Actualmente existen dos maneras principales de publicar grafos sensibles: mediante la generación sintética, o mediante el enmascaramiento. La primera se centra en parametrizar grafos mediante la aproximación de características que han sido privatizadas, como el número de enlaces, el peso de los vértices, o la distribución del grado de los vértices. En cambio, el enmascaramiento se centra en perturbar los vértices o los enlaces del grafo principalmente añadiendo ruido para garantizar un valor ϵ de Privacidad Diferencial [2], con sus siglas en inglés: ϵ-DP. De esta manera, la literatura de privatización de grafos mediante ϵ-DP se separa principalmente en dos casos: Node-DP [3] y Edge-DP [4], en los que se añade ruido en los vértices o en los enlaces del grafo respectivamente.

En este artículo nos centramos en el uso de ϵ-DP para enmascarar los enlaces de los grafos. En concreto, nuestro objetivo es analizar las técnicas actuales de privatización de grafos mediante Edge-DP para ver qué beneficios aportan cada una y entender en qué casos se debe escoger una sobre otra, centrándonos principalmente en la complejidad. Adicionalmente, tras analizar los grafos publicados en las evaluaciones se ha visto una disparidad en las elecciones para los diferentes mecanismos del estado del arte, por lo que proponemos una metodología para seleccionar el conjunto de datos a evaluar. También hemos seleccionado un listado de grafos como recomendación para realizar las evaluaciones.

II. Antecedentes

Los grafos nos permiten representar relaciones de datos complejos mediante el uso vértices y enlaces. Adicionalmente, los grafos se pueden clasificar de diversas maneras, por ejemplo: si los vértices tienen pesos, si los enlaces son unidireccionales o bidireccionales, o si los vértices se pueden auto-referenciar. En esta sección introducimos la definición de grafo simple: unidireccional, sin pesos en los vértices y sin auto-referencias, ya que es el que nos centraremos en el resto del trabajo:

Definición 1 (Grafo simple): Un grafo simple G se forma por una dupla $G = (V, E)$ de vértices (V) y enlaces ($E = \{v_i, v_j\}$), que representan la conexión entre dos vértices, y el enlace $\{v_i, v_j\}$ es equivalente al enlace $\{v_j, v_i\}$.

Adicionalmente, los grafos tienen asociada una matriz de adyacencia que permite representar de forma simple las conexiones entre vértices. Como decíamos anteriormente, en este artículo nos centramos en los grafos simples, es decir: unidireccionales, sin pesos en los vértices y sin auto-referencias, por lo que la matriz de adyacencias es simétrica y la diagonal principal es siempre 0.

Definición 2 (Matriz de adyacencia): La matriz de adyacencia en un grafo simple unidireccional $A = (a_{ij})_{n\times n}$ de G se define:

$$A = \begin{cases} 1 & if\ v_i, v_j \in E \\ 0 & if\ v_i, v_j \notin E. \end{cases} \quad (1)$$

II-A. ϵ-Privacidad Diferencial

Mucha de la literatura que se centra en enmascarar grafos se enfoca en la noción de privacidad diferencial. Esta se definió inicialmente para escenarios interactivos en los que un curador de datos provee una interfaz para que los usuarios hagan consultas. Posteriormente, esta noción se ha adaptado para otros muchos casos. Concretamente ϵ-DP, se define:

Definición 3 (ϵ-Privacidad Diferencial): Un algoritmo $\mathcal{K} : \mathcal{D} \to \mathcal{S}$, con $\mathcal{D}$ siendo el universo de bases de datos, es ϵ-diferencialmente privado (ϵ-DP) si para todas las bases de datos vecinas $D, D' \in \mathcal{D}$ y para todas $S \subseteq \mathcal{S}$,

$$P\{\mathcal{K}(D) \in S\} \leq e^{\epsilon} \times P\{\mathcal{K}(D') \in S\} \quad (2)$$

Para garantizar ϵ-DP existen dos métodos principales que la mayoría de mecanismos usan: Laplaciano y Exponencial. Ambos se basan en añadir ruido, Laplaciano y Gaussiano respectivamente, a una función de consulta f, tal que $a = f(X)$, siendo X la base de datos. Además, se define la sensitividad global para cada función de consulta Δf usada en dos bases de datos D, D' , o su siglas en inglés GS:

$$\Delta f = \max_{x,y,d(x,y)=1} ||f(x) - f(y)||,\ x \in D,\ y \in D' \quad (3)$$

En la *Def. 3*, un elemento clave es el significado de *bases de datos vecinas*, o nivel de granularidad. En el caso de los grafos este elemento se ramifica principalmente en dos opciones: Node-DP [3] y Edge-DP [4]. El primero define grafos vecinos como dos grafos que difieren en: un vértice y todos los enlaces que inciden en él[1]. Formalmente,

$$|V(G) \oplus V(G')| = 1, \text{ y } E(G) \oplus E(G') = E_v, \quad (4)$$

en el que $\{v\} = V(G) \oplus V(G')$ es el vértice en el que difieren, y E_v son los enlaces que inciden en v. En cambio, Edge-DP indica que dos grafos son vecinos si difieren únicamente en un solo enlace. Aunque en Hay, et al. [3] la definición dice que los grafos G y G' pueden diferir por un vértice aislado. Entonces, indican que dos grafos G y G' son vecinos si:

$$|V \oplus V'| + |E \oplus E'| = 1. \quad (5)$$

III. Análisis de los mecanismos

Tal y como se ha introducido en la Sección I, existen maneras diferentes de añadir privacidad a los grafos. Diversos trabajos se centran en la generación sintética como manera de publicar grafos [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. Otros usan Node-DP como método de enmascaramiento [15], [16], [17], [18]. En esta sección definimos y analizamos las técnicas de enmascaramiento basadas en Edge-DP: PiG [19], TmF [20], PLT-PiG [21], EdgeFlipShrink [22] y LapGraph [23]. En cada uno de los siguientes subapartados presentamos primero cada uno de los mecanismos y posteriormente analizamos su comportamiento para diferentes tipos de grafos, las fortalezas y desventajas, y los grafos usados durante la evaluación.

III-A. PiG

PiG [19], también llamado EdgeRand o EdgeFlip en otros artículos, define un parámetro s que permite decidir con qué probabilidad se perturba una celda de la matriz A. Entonces, la probabilidad de que una celda sea preservada es de $1 - s$. En el caso en que la celda deba ser perturbada se escoge el valor con probabilidad $p(0) = p(1) = 0{,}5$. PiG cumple los requisitos para ser ϵ-DP si se satisface:

$$s \geq \frac{2}{1 + e^{\epsilon}}, s \in (0, 1], \epsilon \in (0, \infty) \quad (6)$$

PiG permite estimar el número esperado de enlaces del grafo perturbado $E[|E_P|]$, definido como:

$$E[|E_P|] = (1 - \frac{1}{2} \cdot s) \cdot m + (\frac{n(n-1)}{2} - m) \cdot \frac{1}{2} \cdot s \quad (7)$$

[1]Para grafos bidireccionales serian enlaces incidentes y salientes

En la que $1 - \frac{1}{2} \cdot s$ es la probabilidad de que se preserve la celda después de usar el mecanismo, $\frac{1}{2} \cdot s$ es la probabilidad de que se modifique la celda, n es el número de vértices y m es el número de enlaces en el grafo original. De la misma manera, PiG permite estimar la densidad final del grafo:

$$E[d_P] = \frac{E[|E_P|]}{\frac{n \cdot (n-1)}{2}} = (1 - s) \cdot d + \frac{s}{2} \quad (8)$$

Viendo la *Ec. 8* podemos observar que con "privacidad máxima", es decir $s \to 1$ y $\epsilon \to 0$ la densidad esperada del grafo cada vez se acercará más a $0{,}5$. Por lo que al usar un presupuesto de privacidad bajo, es decir mayor privacidad, todos los grafos acaban homogeneizándose en cuanto al a densidad. Si es necesario un presupuesto de privacidad demasiado bajo, en grafos con muchos vértices vemos que se incrementa mucho la cantidad de enlaces que acaba teniendo el grafo perturbado, haciendo que el uso de este algoritmo sea desaconsejado en estos casos. Además, este mecanismo considera independiente cada celda de la matriz de adyacencia, por lo que es posible que surja un grafo sin ningún enlace al perturbar un grafo con algún enlace. Finalmente, la evaluación realizada durante el artículo se basa en ver el efecto de usar PiG antes de realizar un agrupamiento por comunidades. En el análisis usan dos grafos: uno con pocos vértices, y otro con una cantidad media, ambos con poca densidad. Comparado con el resto de las propuestas la evaluación realizada por los autores es poco diversa, ya que únicamente usan dos grafos con poca variabilidad en las propiedades.

III-B. Top-m Filter (TmF)

TmF [20], se basa en la idea de añadir ruido de Laplace de manera independiente a cada uno de las celdas de la matriz de adyacencia A de un grafo G. TmF primero se separa la $\epsilon = \epsilon_1 + \epsilon_2$, la primera se usará para añadir ruido a las celdas de la matriz A y la segunda se usa para enmascarar el número de enlaces $\tilde{m} = m + Lap(1/\epsilon_2)$.

A continuación se identifica la probabilidad de obtener una celda 0, que viene definida por el área bajo la curva de la distribución de una variable aleatoria de Laplace centrada en 0 con desviación estándar proporcional a ϵ_1. La probabilidad para obtener una celda 1 viene definida de la misma manera, pero con una variable aleatoria de Laplace centrada en 1. Se define un límite θ que controla el número de celdas 0 y celdas 1 que se obtienen a posteriori. Entonces, se encuentran dos posibles casos:

Caso 1: $0 \leq \theta \leq 1$: el número esperado de celdas 1 es:

$$m_1 = \frac{m}{2}(2 - e^{-\epsilon_1(1-\theta)}) \quad (9)$$

El número esperado de celdas 0 es:

$$m_0 = (\frac{n(n-1)}{2}) - m)\frac{1}{2}e^{-\epsilon_1\theta} \quad (10)$$

Sabiendo que $m_0 + m_1 = E[\tilde{m}] = m$, encontramos el valor del límite θ:

$$\theta = \frac{1}{2\epsilon_1} ln(\frac{n(n-1)}{2m} - 1) + \frac{1}{2} \quad (11)$$

Caso 2: $1 \leq \theta$: De la misma manera encontramos θ

$$\theta = \frac{1}{\epsilon_1} ln(\frac{n(n-1)}{4m} + \frac{1}{2}(e^{\epsilon_1} - 1)) \quad (12)$$

Se puede observar que la θ depende del número de enlaces totales y del presupuesto de seguridad ϵ_1. Para saber que caso se encuentra cada celda debemos identificar el valor límite de ϵ_1 que diferencia los dos casos:

$$\epsilon_t = ln(\frac{n(n-1)}{2m} - 1) \tag{13}$$

Una vez encontrados los parámetros en las *Ec. 11, 12 y 13* se separa el algoritmo en dos partes: procesado de las celdas 1 y procesado de las celdas 0. En el primero se añade ruido de Laplace de manera individual a cada celda de la matriz A que contiene un enlace. Únicamente si el valor ruidoso de la celda supera el límite estimado de θ se selecciona como enlace. Tras finalizar esta parte se identifican los enlaces faltantes hasta cumplir el número de enlaces deseado $\tilde{m}$ y se añaden de manera aleatoria hasta llegar al cupo.

Las ventajas principales de esta solución son la poca complejidad, $O(m)$, y que generalmente se respeta uno de los parámetros del grafo original, que es el número de enlaces. Desafortunadamente, TmF no permite *priorizar* otros parámetros del grafo como la distribución de los grados de los nodos. Además, un problema importante es que está focalizado específicamente para grafos poco densos, ya que cuantos más enlaces tenga un grafo más lento será el algoritmo. La evaluación se realiza comparando seis grafos pequeños y medianos. TmF presenta resultados buenos en regímenes de ϵ altos para los grafos presentados, pese a que en casos en que el presupuesto de privacidad es bajo, y especialmente con grafos grandes empeora considerablemente en las métricas de utilidad. Durante la evaluación se menciona el uso de dos grafos de gran tamaño: "amazon" y "youtube", pero no se encuentran en la comparativa de utilidad, pese a que si se menciona el tiempo de cómputo necesario para cada uno.

III-C. PLT-PiG

PLT-PiG [21], "Partial Linear Time PiG" es una variante del mecanismo PiG [19] adaptado junto a metodología del mecanismo TmF [20]. Con las modificaciones realizadas se pretende solucionar uno de los problemas principales de PiG: la complejidad cuadrática. Para solucionar este problema, en los casos en las que el presupuesto de seguridad ϵ usado durante el algoritmo es superior al valor límite ϵ_t (*Ec. 13*), se procesan independientemente las celdas 0 de las celdas 1 de la matriz de adyacencia. De esta manera, se añade el enlace para las celdas 1 con probabilidad $1-\tilde{s},\ \tilde{s} = \frac{2}{e^{\epsilon}+1}$. Notar que $\tilde{s}$ no es la misma que s, esta segunda es un parámetro de diseño y la primera se determina teniendo antes ϵ: $\epsilon = ln(\frac{2}{s}-1)-\epsilon_2$. A continuación, se calculan los enlaces que faltan para cambiar de 0 a 1, $n_0 = (\frac{n(n-1)}{2}-\tilde{m})\frac{\tilde{s}}{2}$ y se añaden de manera aleatoria del conjunto de celdas 0.

Gracias a la modificación añadida, la complejidad de PLT-PiG disminuye en los casos en que $s \leq \frac{4m}{n(n-1)}$, en los que tiene un tiempo lineal. Además, para este rango de s se puede acotar el número esperado de enlaces: $E[|E_P|] \leq 2m$.

Pese a poder acotar el número esperado de enlaces final, no se pueden priorizar o limitar ningún parámetro como la distribución de grados de los vértices o el número de enlaces final. Adicionalmente, PLT-PiG hereda la problemática de PiG en la que, para valores pequeños de ϵ, genera muchos enlaces $E[|E_P|]_{|\tilde{s}=1} = \frac{n(n-1)}{4}$ y puede no funcionar para algunos grafos. La evaluación realizada consta de 9 grafos de tamaños variados y densidad baja-media. En los resultados del artículo original se observa la principal limitación, tanto de PiG como de PLT-PiG: no se puede usar en grafos con ϵ relativamente bajas.

III-D. EdgeFlipShrink

EdgeFlipShrink [22] surge como una respuesta a la problemática generada por PLT-PiG i PiG, en los que el número esperado de enlaces incrementa de manera exponencial cuando disminuye la ϵ.

Este mecanismo mantiene un esquema muy similar a los anteriores. Primero se enmascara m, $\tilde{m} = m + Lap(1/\epsilon_2)$. Se calcula $\tilde{s} = \frac{2}{1+e^{\epsilon}}$, obteniendo ϵ de la misma manera que en la sección anterior. A partir del número esperado de enlaces que se obtiene usando PiG, $m_{PiG} = (1-\tilde{s})\tilde{m} + \frac{n(n-1)}{4}\tilde{s}$, se calcula la probabilidad de encogerse $p = \frac{\tilde{m}}{m_{PiG}}$. Ahora la probabilidad de añadir un enlace existente pasa a ser de $\frac{1-\tilde{s}}{2} \cdot p$. El resto se realiza de la misma manera: se obtiene el número de enlaces faltantes hasta cumplir el esperado $\tilde{m}$ y se obtienen del grupo de celdas 0 de la matriz de adyacencias original.

Al cambiar la probabilidad de mantener un enlace existente de $\frac{1-\tilde{s}}{2}$ de EdgeFlip a $\frac{1-\tilde{s}}{2} \cdot p$ se consigue limitar el número esperado de enlaces para que sea $E[|E_P|] = m$. De esta manera, se obtiene una mejora de la utilidad medida para el número de enlaces del grafo perturbado, comparado con la propuesta original PiG. En la evaluación de este mecanismo se han usado dos grafos medianos y tres grafos con gran número de vértices, pese a que hay poca participación de grafos con densidades altas.

III-E. LapGraph

En LapGraph [23] se analizan las limitaciones de PiG[19] y PLT-PiG [21], principalmente la escalabilidad con el número de vértices, y se propone un nuevo mecanismo que permite trabajar con grafos de gran tamaño. Primero, se divide el presupuesto $\epsilon = \epsilon_1 + \epsilon_2$. ϵ_1 se usa para añadir ruido al número de enlaces $\tilde{m} = m + Lap(1/\epsilon_1)$. Después, se añade ruido de manera independiente a cada enlace $\tilde{A}_{ij} = A_{ij} + Lap(1/\epsilon_2)$. A continuación se seleccionan aquellos $\tilde{m}$ enlaces mayores y se binariza la matriz $\tilde{A}$ para obtener 1s en aquellos $\tilde{m}$ enlaces y 0 en el resto.

El mecanismo es evaluado empíricamente en el artículo original mediante un ataque de re-identificación llamado *LinkTeller*. Este trabajo contiene también un número más elevado de conjuntos de datos, 11 en total, con grafos de tamaño grande y densidades medias-bajas. Desgraciadamente, LapGraph tiene una complejidad $O(|V|^2)$. Por eso, este mecanismo tiene mayor dificultad en casos con grafos de alto número de vértices, debido a que se deben perturbar todos, para luego seleccionar los $\tilde{m}$ más grandes. Además, también disminuye su rendimiento con el aumento de la densidad del grafo, ya que se deben seleccionar aquellas $\tilde{m}$ celdas más grandes. Finalmente, este mecanismo solo permite priorizar una métrica de utilidad, el número de enlaces.

Tabla I
MECANISMOS DE PRIVACIDAD PARA GRAFOS SEGÚN: LA COMPLEJIDAD, SI HAN PRIORIZADO ALGUNA MÉTRICA DE UTILIDAD DURANTE LA PRIVATIZACIÓN Y LA DIVERSIDAD ENCONTRADA EN LOS GRAFOS USADOS PARA LA EVALUACIÓN.

Nombre	Métrica de utilidad priorizada	Diversidad de grafos en la evaluación	Complejidad	Fortalezas	Desventajas
PiG [19]	-	Baja	$O(\|V\|^2)$	▪ Capacidad de estimar $E[\|E_P\|]$	▪ A mayor privacidad mayor número de enlaces finales. ▪ Imposible de usar en grafos grandes y/o densos.
TmF [20]	$E[\|E_P\|]$	Baja	$O(\|E\|)$	▪ Poca complejidad. ▪ Bueno con grafos poco densos.	▪ Complejidad escala con el número de enlaces por lo que es peor con grafos densos.
PLT-PiG [21]	-	Media	$\begin{cases} O(\|E\|) \; if \; s \leq \frac{4m}{n(n-1)} \\ O(\|V\|^2) \; if \; s \geq \frac{4m}{n(n-1)} \end{cases}$	▪ Capacidad de estimar $E[\|E_P\|]$. ▪ Complejidad lineal en algunos valores de s	▪ A mayor privacidad mayor número de enlaces finales. ▪ Imposible de usar en grafos grandes o densos con valores bajos de ϵ.
EdgeFlipShrink [22]	$E[\|E_P\|]$	Baja	$O(\|E\|)$	▪ Bueno con grafos poco densos.	▪ Complejidad depende del número de enlaces.
LapGraph [23]	$E[\|E_P\|]$	Media	$O(\|V\|^2)$	▪ Bueno con grafos poco densos y medianos/pequeños.	▪ Empeora considerablemente con grafos grandes y densos.

IV. DISCUSIÓN

Los mecanismos de privacidad para grafos analizados contienen una problemática generalizada: únicamente se permite priorizar una métrica de utilidad durante la privatización. Es decir, la mayoría de los mecanismos evaluados – TmF, EdgeFlipShrink o LapGraph – usan el número esperado de enlaces $E[|E_P|]$, siendo E_P los enlaces del grafo perturbado, como limitación en el algoritmo. El resto, PiG y PLT-PiG, permiten computar $E[|E_P|]$ en función del parámetro de seguridad s, para poder hacer una estimación. Poder priorizar otras métricas de utilidad, como la distribución de los grados o el coeficiente de agrupamiento, podría ser beneficioso para mejorar el uso que se les puede dar a los grafos privatizados. Adicionalmente, un único mecanismo, LapGraph, evalúa, mediante un ataque, la privacidad empírica obtenida.

Adicionalmente, el mecanismo PiG [19], además de las problemáticas ya mencionadas, escala su complejidad de manera cuadrática con el número de enlaces, por lo que no es adecuado para grafos densos. PLT-PiG [21], permite reducir la complejidad del mecanismo anterior, pese a que en casos con ϵ pequeñas se incrementa el número de vértices esperados de manera exponencial. A pesar de las mejoras introducidas, en casos en que $s \geq \frac{4m}{n(n-1)}$ se sigue usando PiG. EdgeFlipShrink [22] usa la misma idea de "girar" un enlace que los dos anteriores. En este caso, antes de comenzar el algoritmo, se actualizan las probabilidades de girar un enlace usando el número de enlaces ruidoso y el número de enlaces que debería ser obtenido en PiG. De esta manera, el número de enlaces queda limitado a $\tilde{m} = m + Lap(1/\epsilon_2)$. Por otro lado, TmF [20] permite privatizar grafos con poca complejidad mientras que se mantiene el número esperado de enlaces. La complejidad aumenta con el número de enlaces del grafo original, por lo que para grafos muy grandes y densos es desaconsejable. Finalmente, LapGraph [23] usa un enfoque simplista que permite obtener buen rendimiento en grafos poco densos y medianos y priorizar el número de enlaces esperados durante la privatización. Desgraciadamente, grafos muy grandes y densos empeoran su usabilidad.

Una nota importante a realizar en este análisis es la poca diversidad en el uso de grafos para evaluar los algoritmos. Todos los mecanismos han evaluado sus resultados en un conjunto limitado de datos, principalmente en grafos con densidades bajas y número de vértices medio-bajo. Estas configuraciones ensalzan los resultados de mecanismos con alta complejidad, mientras que rara vez se menciona que no es posible su uso en otras configuraciones más demandantes.

Tabla II
NÚMERO DE VÉRTICES Y DENSIDAD POR GRUPO

Grupo	Densidad	Número de vértices
1	$\leq 0{,}0615$	≤ 64
2	$\leq 0{,}0108$	≥ 2642
3	$\geq 0{,}2222$	≤ 10
4	$\geq 0{,}0983$	≥ 93

Hemos señalado la diversidad de grafos durante la evaluación de cada mecanismo en tres categorías: diversidad alta, media o baja:

Diversidad Alta: Al menos se han usado 8 grafos con representación de:

- Grafos con una cantidad pequeña o mediana de vértices y densidad baja
- Grafos con una cantidad pequeña o mediana de vértices y densidad alta
- Grafos con una gran cantidad de vértices independientemente de la densidad

Diversidad Media: Al menos se han usado 6 grafos con representación de:

- Grafos con una cantidad pequeña o mediana de vértices y densidad baja
- Grafos con una gran cantidad de vértices independientemente de la densidad

Diversidad Baja: Los que no han cumplido ninguno de los requisitos se han incluido en la categoría diversidad baja.

Estas tres categorías se han definido teniendo en cuenta que es muy difícil encontrar grafos con gran cantidad de vértices y densidades altas y por eso no ha sido considerado un requisito indispensable. Finalmente, para poder comparar de manera más fácil los mecanismos hemos hecho un resumen en la *Tab. I*.

IV-A. Mejores practicas al seleccionar los conjuntos de datos

Al ver durante el análisis un desbalance en los conjuntos de grafos usados para evaluar los mecanismos de privacidad, realizamos una recomendación de mejores prácticas al seleccionar los conjuntos de grafos. Para obtener una cantidad balanceada de grafos en el conjunto de evaluación se recomienda seguir las líneas definidas en la categoría de diversidad alta. Es decir, al menos usar 8 grafos de 3 categorías: (1) grafos con una cantidad pequeña o mediana de vértices y densidad baja, (2) grafos con una cantidad pequeña o mediana de vértices y densidad alta y (3) grafos con una gran cantidad de vértices independientemente de la densidad. Adicionalmente, usar grafos provenientes de diferentes disciplinas es beneficioso para el análisis ya que las diferencias en topología o en las comunidades existentes variarán. Para ayudar al lector a seleccionar los posibles grafos para realizar evaluaciones hemos realizado una selección de posibles grafos en la *Tab. III*. Se ha realizado una primera selección de $13{,}121$ grafos procedentes de 15 conjuntos de datos diferentes, con números de enlaces variando de $|E| = 1$ a $|E| = 3 \cdot 10^6$, y números de vértices en rangos de $|V| = 2$ a $|V| = 1 \cdot 10^6$. De este listado se han hecho 4 grupos:

- Grupo 1: Densidad baja, número de vértices bajo
- Grupo 2: Densidad baja, número de vértices alto
- Grupo 3: Densidad alta, número de vértices bajo
- Grupo 4: Densidad alta, número de vértices alto

En la *Tab. II* se pueden ver los límites que se han puesto para seleccionar los grupos. La selección ha sido el resultado de un amplio análisis en el que se ha priorizado además de estos límites, la diversidad de conjuntos de datos. Si nos fijamos en el Grupo 4 de las *Tab. II y III* podemos ver que la densidad es mucho menor que del Grupo 3, pese a ser ambos de densidad "alta". Esto es debido a que, como se ha dicho anteriormente, existen pocos grafos simples que contengan un número muy elevado de vértices y una alta densidad a la vez. De esta manera se ha hecho una propuesta con un número moderado de vértices y una densidad moderada.

V. CONCLUSIONES

La necesidad de proteger los datos que se incluyen en los grafos ha aumentado debido a los nuevos casos de uso encontrados en disciplinas como: la biología, el estudio de redes sociales, etc. Por eso, en este artículo nos hemos centrado en analizar las soluciones existentes de privatización de grafos mediante Edge-DP, una variante de ϵ-DP que se centra en enmascarar los enlaces de los grafos para garantizar la privacidad. De las cinco soluciones encontradas tres de ellas (TmF [20], EdgeFlipShrink [22] y LapGraph [23]) permiten priorizar la métrica del número de enlaces final durante la privatización para mejorar los resultados de utilidad. Las dos restantes (PiG [19] y PLT-PiG [21]), se centran en intercambiar enlaces, por lo que el número de enlaces final depende de los parámetros de privacidad de cada uno de los mecanismos. Adicionalmente, hemos observado que la complejidad de Pig y PLT-PiG es bastante alta y en algunos casos puede hacer que no sean utilizables con algunos grafos. Finalmente, se ha detectado durante el análisis de las soluciones que los grafos usados para las evaluaciones de la utilidad tenían poca variabilidad en las características. Es decir, la mayoría tenían un número de vértices mediano y bajo y poca densidad. Por esta razón, hemos definido un conjunto de mejores prácticas al seleccionar grafos para la evaluación de los mecanismos de privacidad. También hemos realizado una recomendación de 21 grafos con diferentes características y de diferentes orígenes que permiten evaluar de manera más exhaustiva la utilidad de las nuevas propuestas.

AGRADECIMIENTOS

Javier Parra-Arnau es el beneficiario de un contrato postdoctoral Ramón y Cajal (ref. RYC2021-034256-I) financiado por el Ministerio de Ciencia e Innovación de España y la Unión Europea — "NextGenerationEU/PRTR (Plan de Recuperación, Transformación y Resiliencia). Este trabajo ha sido parcialmente financiado por el Gobierno de España mediante el proyecto "DIstributed Smart Communications with Verifiable EneRgy-optimal Yields (DISCOVERY)"PID2023-148716OB-C32, y por la Generalitat de Catalunya, mediante la ayuda AGAUR "2021 SGR 01413". Víctor Rubio Jornet cuenta con el apoyo del programa predoctoral AGAUR-FI ayudas (2024 FI-1 00762) Joan Oró de la Secretaría de Universidades e Investigación del Departamento de Investigación y Universidades de la Generalitat de Catalunya y del Fondo Social Europeo Plus.

Tabla III
SELECCIÓN DE GRAFOS RECOMENDADOS

Grupo	Conjunto de datos	Grafo	Densidad	Número de vértices
1	webcompute	job-I2avF_RB8Ik7a36-Q7teww	0.060000	26
1	north	g.34.7	0.058824	34
1	Rome	grafo11544.34	0.060606	34
1	scotch	ar_m8x8	0.055556	64
2	MID	midb	0.000042	11,766
2	arxiv	CA-HepPh	0.001644	12,008
2	econ	econ-poli-large	0.000272	15,575
2	gnutella	p2p-Gnutella25	0.000213	22,687
2	storylines	war	0.000040	24,056
2	scotch	fe_144	0.000103	144,649
2	Cisco	g4_network	0.000008	278,739
2	TmF2	com-youtube	0.000005	1,134,890
3	webcompute	job-133	0.500000	3
3	scotch	nd_7.10	0.476190	7
3	Rome	grafo758.10	0.222222	10
3	north	g.10.74	0.266667	10
4	north	g.44.20	0.098309	44
4	scotch	nd_48.283	0.250887	48
4	Rome	grafo3666.53	0.067489	53
4	webcompute	job-132	0.065766	75
4	Cisco	g1_network	0.101911	1,447

REFERENCIAS

[1] C. Jernigan and B. F. Mistree, "Gaydar: Facebook friendships expose sexual orientation," *First Monday*, 9 2009.

[2] C. Dwork, "Differential Privacy," in *LNCS 4052 - Automata, Languages and Programming*, 2006, pp. 1–12.

[3] M. Hay, C. Li, G. Miklau, and D. Jensen, "Accurate estimation of the degree distribution of private networks," in *Proceedings - IEEE International Conference on Data Mining, ICDM*, 12 2009, pp. 169–178.

[4] V. Karwa, S. Raskhodnikova, A. Smith, and G. Yaroslavtsev, "Private Analysis of Graph Structure," *ACM Transactions on Database Systems*, pp. 1146–1157, 9 2011.

[5] Y. Wang and X. Wu, "Preserving Differential Privacy in Degree-Correlation based Graph Generation," *Transactions on data privacy*, vol. 6, no. 2, pp. 127–145, 8 2013. [Online]. Available: http://www.cs.cmu.edu/enron/

[6] D. Mir and R. N. Wright, "A differentially private estimator for the stochastic Kronecker graph model," in *Proceedings of the 2012 Joint EDBTICDT Workshops*. ACM Digital Library, 3 2012, pp. 167–176.

[7] D. J. Mir and R. N. Wright, "A differentially private graph estimator," in *ICDM Workshops 2009 - IEEE International Conference on Data Mining*, 2009, pp. 122–129.

[8] X. Chen, S. Mauw, and Y. Ramírez-Cruz, "Publishing Community-Preserving Attributed Social Graphs with a Differential Privacy Guarantee," in *Proceedings on Privacy Enhancing Technologies*, vol. 2020, no. 4. Privacy Enhancing Technologies Symposium Advisory Board, 10 2020, pp. 131–152.

[9] M. Eliáš, E. Michael Kapralov, and E. Janardhan Kulkarni Microsoo Research Redmond Yin Tat Lee, "Differentially Private Release of Synthetic Graphs," in *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2019.

[10] Q. Xiao, R. Chen, and K. L. Tan, "Differentially private network data release via structural inference," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 8 2014, pp. 911–920.

[11] L. Hou, W. Ni, S. Zhang, N. Fu, and D. Zhang, "Block-HRG: Block-based differentially private IoT networks release," *Ad Hoc Networks*, vol. 140, 3 2023.

[12] C. Wei, S. Ji, C. Liu, W. Chen, and T. Wang, "AsgLDP: Collecting and Generating Decentralized Attributed Graphs with Local Differential Privacy," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3239–3254, 2020.

[13] Y. Quan, Z. Zhang, L. Du, M. Cheng, P. Cheng, and M. Sun, "PrivGraph: Differentially Private Graph Data Publication by Exploiting Community Information," in *32nd USENIX Conference on Security Symposium (SEC '23)*. USENIX Association, 8 2023.

[14] R. Chen, B. C. Fung, P. S. Yu, and B. C. Desai, "Correlated network data publication via differential privacy," *VLDB Journal*, vol. 23, no. 4, pp. 653–676, 8 2014.

[15] S. P. Kasiviswanathan, K. Nissim, S. Raskhodnikova, and A. Smith, "Analyzing Graphs with Node Differential Privacy," in *10th Theory of Pryptgraphy Conference, TCC2013*, ser. Lecture Notes in Computer Science, A. Sahai, Ed., vol. 7785. Berlin, Heidelberg: Springer Berlin Heidelberg, 3 2013. [Online]. Available: http://link.springer.com/10.1007/978-3-642-36594-2

[16] W. Y. Day, N. Li, and M. Lyu, "Publishing graph degree distribution with node differential privacy," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, vol. 26-June-2016. Association for Computing Machinery, 6 2016, pp. 123–138.

[17] X. Jian, Y. Wang, and L. Chen, "Publishing Graphs under Node Differential Privacy," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 4164–4177, 4 2023.

[18] S. Zhang, W. Ni, and N. Fu, "Community Preserved Social Graph Publishing with Node Differential Privacy," in *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 11 2020. [Online]. Available: http://arxiv.org/abs/2101.01450

[19] Y. Mülle, C. Clifton, and K. Böhm, "Privacy-Integrated Graph Clustering Through Differential Privacy," in *EDBT/ICDT Workshops*, 3 2015, pp. 247–254.

[20] H.-H. Nguyen, "Social Graph Anonymization," Ph.D. dissertation, Université de Lorraine, 2016. [Online]. Available: http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm

[21] H. H. Nguyen, M. Rusinowitch, H. H. Nguyen, A. Imine, and M. Rusinowitch, "Network structure release under differential privacy," *Transactions on Data Privacy*, 12 2016. [Online]. Available: https://www.researchgate.net/publication/312063774

[22] H. H. Nguyen, A. Imine, and M. Rusinowitch, "Detecting Communities under Differential Privacy," in *WPES '16: Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society*, 10 2016, pp. 83–93. [Online]. Available: http://arxiv.org/abs/1607.02060

[23] F. Wu, Y. Long, C. Zhang, and B. Li, "LinkTeller: Recovering Private Edges from Graph Neural Networks via Influence Analysis," in *2022 IEEE Symposium on Security and Privacy (SP)*, 8 2021. [Online]. Available: http://arxiv.org/abs/2108.06504

Evaluación comparativa de esquemas ZKP para credenciales anónimas en clave eIDAS2.0

Alfonso Camblor-García
Gradiant
atlanTTic-Universidad de Vigo
acamblor@gradiant.org
alfonso.camblor@uvigo.gal

Xavier Martínez-Luaña
Gradiant
atlanTTic-Universidad de Vigo
xmartinez@gradiant.org
xamartinez@alumnado.uvigo.gal

Manuel Fernández-Veiga
atlanTTic-Universidad de Vigo
mveiga@det.uvigo.es

Pablo Santos-Cabaleiro
Gradiant
pscabaleiro@gradiant.org

***Abstract*—Este trabajo presenta un benchmarking comparativo de los esquemas BBS, Longfellow-zk y Crescent Credentials aplicados a credenciales anónimas en el contexto eIDAS 2.0. Se evalúan tiempos de firma, generación y verificación de pruebas, así como el tamaño de los artefactos y su escalabilidad según el número de atributos divulgados. Los resultados muestran que Longfellow-zk ofrece la mejor eficiencia temporal y la única seguridad post-cuántica entre los esquemas analizados, aunque con pruebas de gran tamaño que limitan su uso práctico. BBS destaca por su compacidad y su avance hacia la estandarización, pero presenta mayores latencias y ausencia de protección post-cuántica. Crescent proporciona un equilibrio intermedio con tamaños moderados y compatibilidad con credenciales existentes. El estudio evidencia que ningún esquema satisface simultáneamente eficiencia, compacidad y resiliencia post-cuántica, subrayando la necesidad de explorar ZKPs alternativos, especialmente los basados en retículos.**

***Index Terms*—Pruebas de Conocimiento Cero, Privacidad, Criptografía, Credenciales anónimas**

I. Introducción

La aprobación del Reglamento eIDAS2 [19] marca un punto de inflexión en la gestión de la identidad digital dentro de la Unión Europea. Con él, emerge un nuevo paradigma de identidad descentralizada, en el que la European Digital Identity Wallet (EUDI Wallet) [**?**] y las credenciales verificables [20] se convierten en el eje central de la interacción entre ciudadanos, administraciones públicas y proveedores de servicios. Esta transformación pretende dotar a los usuarios de un control sin precedentes sobre sus atributos e identificadores, reforzando simultáneamente la soberanía digital europea.

Sin embargo, este avance regulatorio viene acompañado de un desafío técnico de gran envergadura: eIDAS2 exige niveles de privacidad extremadamente elevados, tanto en la emisión como en el uso de credenciales digitales. En particular, impone que los usuarios puedan demostrar atributos concretos sin revelar información adicional [18], evitando cualquier forma de correlación entre transacciones, servicios o emisores. Este requisito no solo tensiona los sistemas actuales de gestión de identidad, sino que obliga a revisar y evolucionar los esquemas de Pruebas de Conocimiento Cero (ZKPs) [1] empleados hasta ahora.

En este contexto, las credenciales anónimas —capaces de combinar verificabilidad, no correlación y privacidad formalmente demostrable— se convierten en un componente indispensable. No obstante, los esquemas ZKP tradicionales no fueron diseñados para funcionar en ecosistemas interoperables, masivos y altamente regulados como el que propone eIDAS2. La necesidad de soportar múltiples atributos, revocación eficiente, dispositivos móviles y restricciones criptográficas específicas del reglamento exige nuevas aproximaciones que optimicen tanto la seguridad como el rendimiento.

Este trabajo se sitúa precisamente en esa intersección crítica: la búsqueda de mecanismos criptográficos avanzados que permitan implementar credenciales anónimas compatibles con las exigencias de eIDAS2, con un énfasis particular en la eficiencia, la escalabilidad y la experiencia de usuario. A través de un análisis exhaustivo y una evaluación comparativa, exploramos el potencial de los esquemas modernos de ZKPs para responder a los retos que plantea la identidad digital europea de próxima generación.

En este contexto, las contribuciones principales de este trabajo son tres. Primero, realizamos un benchmarking experimental de los esquemas BBS, Longfellow-zk y Crescent en escenarios realistas de divulgación selectiva de atributos en el contexto eIDAS2.0. Segundo, ofrecemos un análisis comparativo de su rendimiento y eficiencia —tiempos de firma, generación y verificación, tamaño de pruebas y escalabilidad según el número de atributos— proporcionando criterios prácticos de selección según las necesidades del despliegue. Finalmente, mostramos que ninguno de los esquemas satisface simultáneamente flexibilidad, eficiencia y seguridad post-cuántica, lo que evidencia la necesidad de explorar nuevas soluciones ZKP post-cuánticas, especialmente basadas en retículas.

II. Estado del arte

Las pruebas de conocimiento cero (ZKP) constituyen una de las primitivas criptográficas más activamente investigadas en los últimos años, impulsadas por aplicaciones que requieren privacidad fuerte, verificabilidad y eficiencia. Su evolución —desde los primeros protocolos interactivos hasta los modernos argumentos sucintos no interactivos— ha permitido que formen parte de sistemas reales de gran escala, incluyendo identidad digital, *blockchains* y credenciales verificables. En este contexto, destacan tres familias principales: zkSNARKs, zkSTARKs y protocolos *MPC-in-the-head*, junto con un ecosistema cada vez más maduro de herramientas para el diseño de circuitos y pruebas complejas.

A. Protocolos fundamentales de ZKP

Los **zkSNARKs** son ampliamente utilizados por su capacidad para generar pruebas compactas y verificables en tiempos reducidos. Entre ellos, Groth16 [12] sigue siendo un referente por su verificación eficiente y tamaño constante, aunque requiere un *trusted setup* por circuito. PLONK [13] introduce un diseño universal que permite reutilizar parámetros y construir circuitos complejos con soporte para recursión eficiente [9]. Ambos esquemas se basan en curvas emparejables y ofrecen un equilibrio sólido entre rendimiento, expresividad y coste computacional. Los **zkSTARKs**, en cambio, eliminan la necesidad de configuraciones de confianza, apoyándose únicamente en funciones hash. Además de transparencia total, proporcionan resistencia intrínseca frente a adversarios cuánticos, aunque a costa de pruebas más voluminosas que en los esquemas basados en *pairings*. Su transparencia y auditabilidad los hacen especialmente atractivos en infraestructuras reguladas. Los protocolos **MPC-in-the-head** generan pruebas mediante la simulación de una computación multipartita segura, lo que permite evitar estructuras algebraicas complejas. Se basan principalmente en funciones hash y compromisos, ofreciendo transparencia y flexibilidad, aunque con un rendimiento inferior al de los SNARKs más optimizados. De manera transversal, la investigación actual busca (i) *universalidad* en la configuración, (ii) *recursión eficiente* para grandes agregaciones de pruebas y (iii) mejoras sistemáticas en tiempos y tamaños, con vistas a habilitar despliegues masivos como los previstos en la identidad digital europea.

B. Toolchains y ecosistemas de implementación

La adopción industrial de ZKPs ha sido impulsada por la maduración de bibliotecas que facilitan el desarrollo práctico. Arkworks [22] ofrece un ecosistema modular y optimizado para curvas elípticas y compromisos criptográficos, mientras que Circom [26] ha democratizado la creación de pruebas mediante un DSL accesible y `snarkjs`, reduciendo los costes de entrada y acelerando la transición desde prototipos hacia sistemas interoperables. Estos entornos, junto con patrones de diseño auditados, han facilitado la experimentación con credenciales anónimas y el desarrollo de flujos avanzados que respetan la privacidad por diseño.

C. Expresividad mediante pruebas de predicado

Las **pruebas de predicado** permiten demostrar propiedades sobre atributos privados sin revelarlos, respetando los principios de minimización de datos y no correlación. Entre las más relevantes se encuentran las pruebas de rango (por ejemplo, para demostrar mayoría de edad), las de pertenencia o no pertenencia a conjuntos (usadas en listas de permisos o revocación) y las de igualdad/desigualdad entre valores comprometidos. Estos predicados se combinan con mecanismos como *nullifiers*, claves efímeras o compromisos re-aleatorizados para garantizar independencia criptográfica entre presentaciones y preservar la privacidad desde el origen.

D. Tendencias y mecanismos aplicados a credenciales anónimas

El interés creciente en credenciales anónimas ha impulsado la integración de ZKPs con otras primitivas avanzadas, como FHE, MPC y técnicas de privacidad diferencial. Paralelamente, se investigan optimizaciones de bajo nivel —incluyendo aceleración por GPU, ASICs especializados en FFT y técnicas algebraicas avanzadas como *lookup arguments* [8] y *folding schemes* [9]— que buscan reducir costes y facilitar la verificación en dispositivos móviles o *secure elements*. En el marco de eIDAS2, estas mejoras son especialmente relevantes debido a los requisitos de no correlación, revelación mínima y demostración selectiva exigidos por las EUDI Wallets.

E. Sistemas de credenciales basados en ZK: BBS, Longfellow y Crescent

Tres propuestas destacan actualmente por su madurez en credenciales anónimas: **BBS** [2], primer esquema para revelación selectiva considerado para estandarización por su compatibilidad con el modelo W3C y su integración industrial; **Longfellow** [27], diseñado por Google para minimizar la latencia de generación mediante preprocesamiento y técnicas derivadas de Ligero; y **Crescent Credentials** [6], propuesta de Microsoft que introduce flujos SNARK optimizados y compatibilidad con credenciales existentes sin requerir cambios en la infraestructura de emisión. Estas tres aproximaciones representan distintas estrategias para equilibrar privacidad, eficiencia y usabilidad en el marco regulatorio y técnico de eIDAS2.

III. Preliminares

En esta sección, se describen los 3 esquemas objeto de estudio de este trabajo.

BBS (Boneh-Boyen-Shacham). Las firmas BBS representan una tecnología criptográfica avanzada propuesta para la creación de credenciales anónimas. Se encuentra en proceso de estandarización, lo que la convierte en la solución más madura actual para su adopción en el contexto de la European Digital Identity Wallet (EUDI Wallet). Sin embargo, aunque algunos de los mejores criptógrafos europeos han recomendado este esquema para la EUDI Wallet [29], su falta de seguridad post-cuántica ha provocado que la Comisión no tenga en cuenta este esquema al menos para la primera iteración de las carteras (finales de 2026). Esta tecnología se basa en el esquema de firmas digitales desarrollado por Boneh, Boyen y Shacham, que permite capacidades únicas de revelación selectiva de datos.

Características técnicas principales:

- **Revelación selectiva:** Permite a los usuarios revelar únicamente los atributos específicos de una credencial que son necesarios para una verificación particular, manteniendo el resto de la información privada.
- **Privacidad mejorada:** Reducen significativamente la posibilidad de correlación y trazado entre diferentes presentaciones de credenciales, ya que cada presentación puede generar pruebas criptográficas únicas sin revelar la firma original.
- **Eficiencia computacional:** Ofrecen un equilibrio favorable entre seguridad criptográfica y rendimiento, siendo viables para implementación en dispositivos móviles.

- **Interoperabilidad:** Están diseñadas para ser compatibles con estándares W3C como Verifiable Credentials y DID (Decentralized Identifiers).

Requisitos de implementación: Las firmas BBS requieren certificados y credenciales específicamente diseñados para este esquema criptográfico, lo que implica adaptaciones en la infraestructura de emisión de credenciales existente. Los emisores deben implementar algoritmos de firma BBS para generar credenciales que soporten revelación selectiva [30].

Longfellow (Google). Es un protocolo innovador que combina técnicas avanzadas de verificación con Zero-Knowledge Proofs (ZKPs) basadas en los desarrollos del protocolo Ligero [5], [27], [7]. Esta tecnología está específicamente diseñada para abordar dos desafíos críticos en el ecosistema de credenciales digitales: la fricción en los procesos de renovación de credenciales y la preparación para la era postcuántica. Actualmente, se encuentra en un proceso de revisión de seguridad por parte de un equipo de expertos, y se ha posicionado como una de las soluciones más prometedores de cara a futuras iteraciones de la EUDI Wallet con soporte a credenciales anónimas. El hecho de que las pruebas generadas tengan seguridad post-cuántica, y que operen directamente sobre curva elíptica, lo convierten en la solución más atractiva a priori. Es importante mencionar que la curva elíptica va a seguir teniendo importancia en los primeros años de la migración a criptografía post-cuántica, puesto que ETSI está impulsando los enfoques de hibridación para que la migración se realice de la manera menos invasiva posible.

Características técnicas principales:

- **Protocolo híbrido:** Integra mecanismos de verificación tradicionales con pruebas de conocimiento cero, permitiendo validación eficiente sin comprometer la privacidad.
- **Optimización de Ligero:** Aprovecha las mejoras del protocolo Ligero para reducir el tamaño de las pruebas y acelerar los tiempos de verificación, especialmente relevante para aplicaciones móviles.
- **Resistencia postcuántica:** Utiliza primitivas criptográficas que se consideran seguras frente a ataques de computadoras cuánticas, preparando el sistema para futuras amenazas tecnológicas.

Crescent (Microsoft). Crescent es un sistema de credenciales anónimas que introduce un enfoque radicalmente pragmático: permite añadir propiedades de privacidad avanzadas —como *selective disclosure*, *unlinkability* y pruebas de predicados— a credenciales ya existentes, sin requerir ninguna modificación por parte del emisor original. A diferencia de otros sistemas que necesitan que las credenciales se emitan bajo un esquema criptográfico específico (como BBS), Crescent actúa como una capa intermedia que transforma credenciales tradicionales, tales como JSON Web Tokens (JWT) o Mobile Driver's Licenses (mDL), en credenciales anónimas mediante el uso de pruebas de conocimiento cero. Su diseño está pensado para acelerar la adopción industrial de credenciales privadas, evitando cambios costosos en la infraestructura de emisión.

El pilar técnico de Crescent consiste en separar el proceso en dos fases: una fase *offline* costosa, realizada solo una vez por credencial, y una fase *online* extremadamente eficiente para cada presentación. En la fase offline, Crescent genera una prueba Groth16 que demuestra que la credencial está correctamente firmada y que sus atributos se han extraído de forma íntegra. Posteriormente, este SNARK se reutiliza infinitas veces gracias a una re-aleatorización segura, combinándose con pruebas Sigma y compromisos de Pedersen para producir presentaciones ligeras, no vinculables y altamente flexibles.

Características técnicas principales:

- **Compatibilidad con credenciales existentes:** No requiere que los emisores adopten nuevos algoritmos de firma ni que modifiquen su infraestructura. Funciona directamente sobre JWT, mDL y credenciales firmadas mediante RSA, ECDSA o algoritmos similares.
- **Arquitectura en dos fases:** Realiza una única prueba SNARK costosa tras la emisión de la credencial, permitiendo que cada presentación posterior (*Show*) requiera únicamente unas pocas pruebas Sigma y operaciones sobre compromisos, con tiempos de ejecución de apenas decenas de milisegundos.
- **Revelación selectiva y pruebas sobre atributos:** Los atributos se comprometen mediante compromisos de Pedersen, permitiendo revelar solo un subconjunto o realizar pruebas como $age > 18$.
- **Desvinculación garantizada:** Cada presentación incluye una re-aleatorización del SNARK y de los compromisos, lo que impide correlacionar dos usos de la misma credencial incluso bajo supuestos adversarios fuertes.
- **Extensibilidad modular:** Permite añadir pruebas adicionales, como *device binding* mediante firmas ECDSA verificadas en ZK, limitación de tasas de uso (*rate limiting*) o firma pseudónima basada en conocimiento de una credencial.

Requisitos de implementación: Crescent exige una fase de generación de parámetros para Groth16 con ceremonia de confianza (Trusted Setup), así como circuitos específicos para cada tipo de credencial y algoritmo de firma utilizado por los emisores. Sin embargo, una vez generados estos parámetros, los verificadores no necesitan demasiados recursos para llevar a cabo su labor. Este diseño reduce significativamente las barreras de adopción, facilitando su implantación en ecosistemas donde ya se emplean JWT o mDL como formatos estándar [6].

IV. Resultados

En esta sección se presentan los resultados obtenidos en las diversas pruebas de rendimiento realizadas. Las implementaciones utilizadas corresponden a BBS [25], LongFellow-Zk [27] y Crescent [24], ejecutadas sobre un entorno homogéneo de hardware y software para garantizar la comparabilidad de los resultado: (i) CPU: Intel(R) Core(TM) Ultra 7 165H, (ii) RAM: 32GB y Ubuntu 24.

Con el objetivo de realizar correctamente el benchmarking de estos 3 esquemas, se han comparado todas las fases operativas de los mismos como parte de un proceso de divulgación selectiva de atributos. Esto comprende realizar una comparativa entre los tiempos de firma/verificación de ECDSA (sobre las que operan lonfellow-zk y crescent) frente

Métrica (2 atributos)	BBS	Longfellow-zk	Crescent
Firma (ms)	24	4	4
Generación de prueba (ms)	9.931	0.698	24
Verificación de prueba (ms)	11.758	0.316	22.1
Tamaño de la prueba (bytes)	528	325792	11447

TABLE I: Comparación de tiempos y tamaños para 2 atributos divulgados.

a BBS, los tiempos de generación y verificación de pruebas de los 3 esquemas, y los tamaños de las pruebas generadas. Adicionalmente, se ha visto la evolución de estos tiempos y tamaños en función del número de atributos a divulgar, para evaluar el comportamiento de estos esquemas en diversas situaciones.

En la siguiente tabla I se presentan los resultados obtenidos para el caso de dos atributos divulgados, lo que permite comparar de forma directa el comportamiento práctico de los tres esquemas evaluados en un escenario representativo de identidad digital con credenciales anónimas.

Como podemos observar, en el proceso de firma, BBS alcanza un tiempo de 24 ms, considerablemente superior al de Longfellow-zk y Crescent, ambos con 4 ms. Esta diferencia refleja la mayor complejidad del algoritmo de firma basado en emparejamientos de BBS frente a esquemas más ligeros como ECDSA utilizados en los otros dos sistemas. Aunque este coste puede ser aceptable en flujos de emisión poco frecuentes, resulta menos adecuado en escenarios que requieren firmas frecuentes o actualización dinámica de credenciales. Esto es un aspecto muy relevante, ya que aunque a priori los procesos de presentación de atributos ocurrirán más amenudo que las emisiones de credenciales, esto tiene matices. La revocación de credenciales es un aspecto crítico que por ahora no se está abordando en la EUDI Wallet, ya que introduce muchos problemas privacidad y trazabilidad del usuario a lo largo del sistema. Esto provoca que la recomendación actual sea reducir los tiempos de validez de la credencial, favoreciendo procesos de renovación de credenciales. Este hecho provoca que aumenten en gran medida la cantidad de crecenciales anónimas emitidas, y por tanto el impacto de las firmas BBS sería más notorio.

En la fase de generación de pruebas se observa la mayor divergencia entre tecnologías. BBS requiere 9.931 ms, mientras que Crescent presenta el tiempo más alto con 24 ms. Por el contrario, Longfellow-zk destaca con 0.698 ms, situándose más de un orden de magnitud por debajo de ambos y demostrando una enorme eficiencia en este proceso. Este resultado lo convierte en la opción más adecuada para escenarios con presentaciones frecuentes o en dispositivos con recursos limitados.

La verificación de pruebas reproduce el patrón anterior. BBS necesita 11.758 ms y Crescent 22.1 ms, lo que evidencia un coste significativo en ambos casos, especialmente en Crescent. En cambio, Longfellow-zk vuelve a mostrar el mejor rendimiento, con solo 0.316 ms, ofreciendo verificaciones extremadamente rápidas y adecuadas para despliegues masivos o validaciones repetidas en tiempo real.

El tamaño de la prueba revela una compensación clara. BBS produce artefactos muy compactos, con 528 bytes, lo que facilita su transmisión y almacenamiento. Crescent se mantiene en una posición intermedia con 11 447 bytes, aún razonable en la mayoría de contextos. Sin embargo, Longfellow-zk genera pruebas de 325 792 bytes, más de dos órdenes de magnitud superiores, lo que puede limitar su uso en redes de baja capacidad o en dispositivos con restricciones estrictas de almacenamiento. Esto es especialmente relevante en presentaciones offline basadas en NFC, tremendamente relevantes para la EUDI Wallet, en la que el tamaño de prueba es directamente prohibitivo para este tipo de autenticaciones por proximidad.

El análisis conjunto evidencia que no existe una solución universalmente superior, sino que cada esquema presenta fortalezas y limitaciones según las prioridades del sistema: rapidez extrema (Longfellow-zk), pruebas compactas (BBS), o un punto intermedio en tamaño y con firmas ligeras (Crescent).

Las tres figuras del documento muestran claramente cómo se comportan BBS, Longfellow-zk y Crescent cuando aumenta el número de atributos divulgados. Lo relevante es que el comportamiento observado es lineal en unos casos, sublineal en otros y claramente no monótono en Crescent, lo que permite extraer conclusiones importantes para el diseño de credenciales anónimas en EUDI Wallet.

A partir de las gráficas presentadas en la Figura 1, puede observarse que el tiempo de generación de pruebas evoluciona de forma muy distinta para cada uno de los tres esquemas evaluados. En el caso de BBS, la curva muestra un comportamiento descendente conforme aumenta el número de atributos divulgados. Este patrón, que a primera vista podría parecer contraintuitivo, se explica por la estructura interna del protocolo: al revelar más atributos, disminuyen los elementos que deben permanecer ocultos y, con ello, la complejidad algebraica asociada a las ecuaciones de verificación basadas en emparejamientos. En consecuencia, aunque BBS mantiene un coste inicial elevado, su escalado con respecto al número de atributos es relativamente favorable. Por el contrario, Longfellow-zk experimenta un incremento progresivo del tiempo de generación, aproximadamente lineal, que refleja cómo cada atributo adicional introduce compromisos, aperturas y operaciones hash propias de los protocolos tipo Ligero y MPC-in-the-head. Aun así, incluso en los escenarios más exigentes, Longfellow-zk continúa siendo el esquema con menor latencia absoluta, situándose varios órdenes de magnitud por debajo de BBS y Crescent. Crescent, por su parte, apenas presenta una evolución marcada. Aumentar el número de atributos a relevar provoca variaciones sutiles en el tiempo de generación de prueba, pero despreciables frente al tiempo total de generación. Estos resultados son muy interesantes, puesto que en escenarios en los que se lidie con credenciales con una cantidad muy elevada de atributos, BBS y Crescent son una opción que claramente escala mejor que longfellow-zk. Puede que esto no tenga una relevancia crítica en la EUDI Wallet actual para personas, pero puede ser un aspecto clave con la futura EU Business Wallet (EUB Wallet) en la que se gestionarán atributos de entidades.

La Figura 2 revela un patrón similar en el tiempo de verificación de las pruebas, aunque con matices relevantes. En BBS, la curva vuelve a descender de forma pronunciada

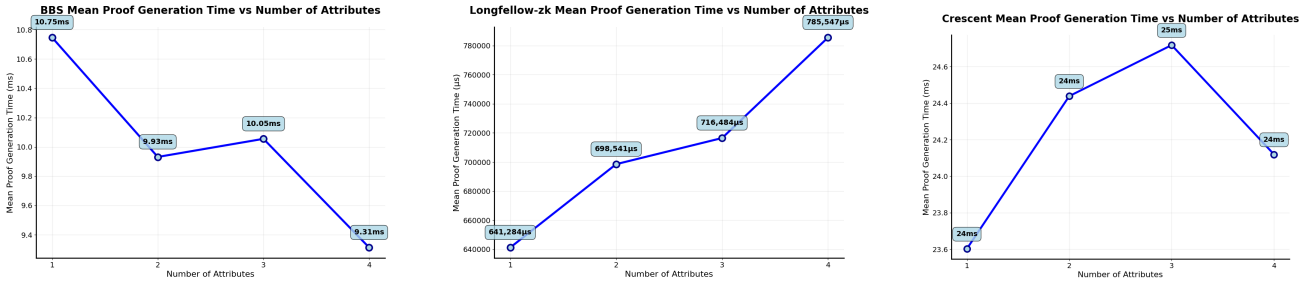


Fig. 1: Medición del tiempo de generación de pruebas en función del número de atributos divulgados para los tres esquemas evaluados.

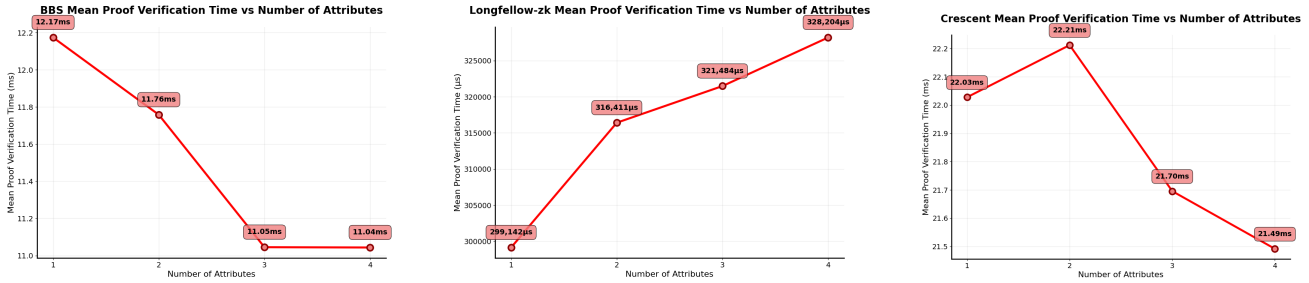


Fig. 2: Medición de tiempo de verificación de pruebas en función del número de atributos divulgados para los tres esquemas evaluados.

a medida que aumenta el número de atributos revelados. La explicación sigue siendo la simplificación estructural derivada de la reducción de elementos que deben mantenerse privados, lo que disminuye el número de verificaciones de emparejamiento necesarias. En el caso de Longfellow-zk, se observa nuevamente un incremento suave pero sostenido, que reproduce la lógica de su esquema de compromisos y verificaciones hash. A pesar de este crecimiento, Longfellow-zk permanece en todo momento como el sistema más eficiente, con tiempos consistentemente inferiores al milisegundo. Crescent presenta de nuevo un comportamiento no monótono pero con una variabilidad reducida. La Figura 3 permite analizar la evolución del tamaño de las pruebas generadas y presenta diferencias aún más marcadas entre los tres esquemas. Las pruebas producidas por BBS muestran una ligera disminución conforme aumenta el número de atributos, pero se mantienen siempre extremadamente compactas, rondando medio kilobyte. Este resultado sitúa a BBS como el esquema más adecuado para escenarios donde el tamaño es crítico, como NFC o bluetooth. En contraste, Longfellow-zk genera pruebas de tamaño muy elevado desde el inicio —en torno a 300 KB— y solo muestra una reducción marginal con más atributos divulgados. Este tamaño, dos órdenes de magnitud superior al de Crescent y casi tres órdenes por encima de BBS, compromete gravemente su aplicabilidad en entornos móviles. Crescent se sitúa en una posición intermedia, con tamaños en el rango de 10–20 KB, suficientemente compactos para la mayor parte de los flujos de credenciales verificables y sensiblemente reducidos cuando se revelan más atributos, debido al menor número de compromisos necesarios.

V. Conclusiones

Este trabajo ha comparado experimentalmente tres tecnologías de Pruebas de Conocimiento Cero aplicadas a credenciales anónimas —BBS, Longfellow-zk y Crescent—, evaluando su rendimiento en escenarios representativos de la EUDI Wallet. Los resultados muestran que cada esquema responde a prioridades distintas y que no existe una solución única óptima, sino un conjunto de compromisos entre eficiencia temporal, tamaño de las pruebas, interoperabilidad y sostenibilidad criptográfica.

Longfellow-zk se revela como la opción más eficiente en términos de latencia, con tiempos de generación y verificación muy inferiores al milisegundo, incluso cuando aumenta el número de atributos divulgados. Además, constituye la única de las soluciones evaluadas que ofrece seguridad post-cuántica, un requisito clave para la evolución del ecosistema eIDAS a medio plazo. Sin embargo, su principal limitación —el tamaño extremadamente elevado de las pruebas, del orden de cientos de kilobytes— restringe severamente su aplicabilidad en contextos móviles, flujos offline y entornos con canales de comunicación limitados.

BBS, por el contrario, muestra el mejor comportamiento en términos de compacidad, con pruebas de tan solo unos cientos de bytes. Su madurez y su avance hacia la estandarización internacional lo convierten en un candidato sólido a corto plazo, especialmente en escenarios donde el tamaño de los artefactos resulta crítico. No obstante, sus tiempos de generación y verificación son notablemente superiores a los de Longfellow-zk, y su falta de seguridad post-cuántica limita su proyección como solución duradera en el marco regulatorio europeo.

Crescent ocupa una posición intermedia que combina tamaños moderados, compatibilidad con credenciales

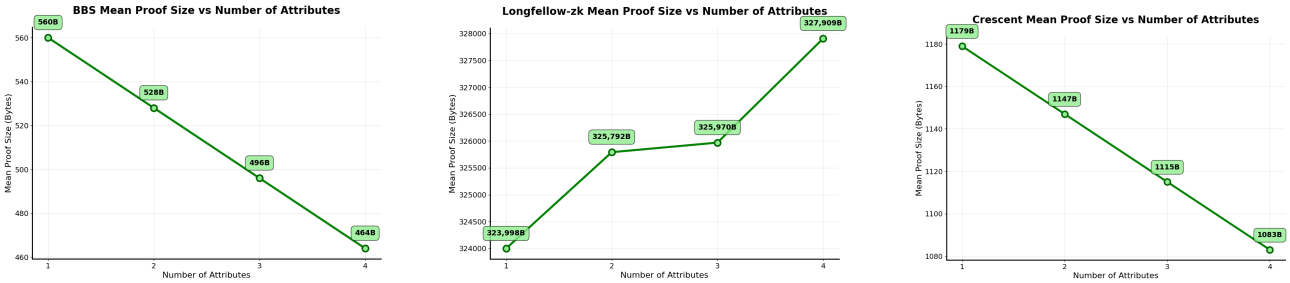


Fig. 3: Medición de tamaño de las pruebas en función del número de atributos divulgados para los tres esquemas evaluados.

existentes y un rendimiento estable, aunque con tiempos de prueba más elevados que Longfellow-zk. Su arquitectura híbrida —una fase offline costosa seguida de presentaciones ligeras— lo convierte en una alternativa pragmática para despliegues reales de gran volumen que requieren no correlación y mínima fricción para el usuario.

En conjunto, los resultados confirman que las tecnologías actuales cubren parcialmente las necesidades de la identidad digital europea, pero dejan patente que ningún esquema satisface simultáneamente eficiencia, flexibilidad, y seguridad post-cuántica. Por ello, este estudio revela la necesidad de seguir explorando nuevas familias de ZKPs post-cuánticas, especialmente aquellos basadas en lattices, que prometen combinar pruebas más ligeras y tamaños razonables con una resistencia criptográfica adecuada para el horizonte regulatorio que plantea eIDAS2 y las futuras EU Business Wallet y EUB Wallet.

Agradecimientos

Este trabajo ha sido financiado parcialmente por el Programa Marco Horizonte Europa de la Unión Europea para Acción de Innovación en el marco del proyecto TrustED (proyecto n.º 10.3030/101168467), por el proyecto CRITIC (teCnologías disRuptivas para la protección, evaluación y operación segura de dIsposiTivos Industriales Conectados) financiado por el CDTI en la convocatoria TransMisiones 2024 (MIG-20242112), y por el proyecto de I+D+i PID2023-148716OB-C31 (DISCOVERY), financiado por MICIU/AEI/10.13039/501100011033/ y "FEDER/UE".

This work was supported by the grant PID2020-113795RB-C33 funded by MICIU/AEI/10.13039/501100011033 (COMPROMISE project); the grant PID2023-148716OB-C31 funded by MCIU/AEI/10.13039/501100011033 (DISCOVERY project); and "TRUFFLES: TRUsted Framework for Federated LEarning Systems, within the strategic cybersecurity projects (INCIBE, Spain), funded by the Recovery, Transformation and Resilience Plan (European Union, Next Generation)". Additionally, it also has been funded by the Galician Regional Government under project ED431B 2024/41 (GPC).

References

[1] Goldwasser, Shafi and Micali, Silvio and Rackoff, Charles. *The knowledge complexity of interactive proof systems*. (1985). https://doi.org/10.1145/22145.22178

[2] Boneh, Dan and Boyen, Xavier and Shacham, Hovav. *Short group signatures*. (2004).

[3] Camenisch, Jan and Lysyanskaya, Anna. *Signature schemes and anonymous credentials from bilinear maps*. (2004).

[4] Goldreich, Oded. *Foundations of Cryptography: Volume 1*. (2006).

[5] Frigo, Matteo and Shelat, Abhi. *Anonymous credentials from ECDSA*. (2024). https://eprint.iacr.org/2024/2010

[6] Paquin, Christian and Policharla, Guru-Vamsi and Zaverucha, Greg. *Crescent: Stronger privacy for existing credentials*. (2024). https://eprint.iacr.org/2024/2013

[7] Ames, Scott and Hazay, Carmit and Ishai, Yuval and Venkitasubramaniam, Muthuramakrishnan. *Ligero: Lightweight sublinear arguments without a trusted setup*. (2022). https://eprint.iacr.org/2022/1608

[8] Campanelli, Matteo and Faonio, Antonio and Fiore, Dario and Li, Tianyu and Lipmaa, Helger. *Lookup arguments: Improvements, extensions and applications to zero-knowledge decision trees*. (2023). https://eprint.iacr.org/2023/1518

[9] Kothapalli, Abhiram and Setty, Srinath. *SuperNova: Proving universal machine executions without universal circuits*. (2022). https://eprint.iacr.org/2022/1758

[10] Kothapalli, Abhiram and Setty, Srinath and Tzialla, Ioanna. *Nova: Recursive zero-knowledge arguments from folding schemes*. (2021). https://eprint.iacr.org/2021/370

[11] Ernstberger, Jens and Chaliasos, Stefanos and Kadianakis, George and Steinhorst, Sebastian and Jovanovic, Philipp and Gervais, Arthur and Livshits, Benjamin and Orru, Michele. *zk-Bench: A toolset for performance benchmarking of SNARKs*. (2023). https://eprint.iacr.org/2023/1503

[12] Groth, Jens. *On the size of pairing-based non-interactive arguments*. (2016). https://eprint.iacr.org/2016/260

[13] Gabizon, Ariel and Williamson, Zachary J. and Ciobotaru, Oana. *PLONK: Permutations over Lagrange bases for universal SNARKs*. (2019). https://eprint.iacr.org/2019/953

[14] Ben-Sasson, Eli and Bentov, Iddo and Horesh, Yinon and Riabzev, Michael. *Scalable, transparent, and post-quantum secure computational integrity*. (2018). https://eprint.iacr.org/2018/046

[15] Bünz, Benedikt and Bootle, Jonathan and Boneh, Dan and Poelstra, Andrew and Wuille, Pieter and Maxwell, Greg. *Bulletproofs: Short proofs for confidential transactions and more*. (2017). https://eprint.iacr.org/2017/1066

[16] Kales, Daniel and Zaverucha, Greg. *Improving the performance of the Picnic signature scheme*. (2020). https://eprint.iacr.org/2020/427

[17] Setty, Srinath. *Spartan: Efficient and general-purpose zkSNARKs without trusted setup*. (2019). https://eprint.iacr.org/2019/550

[18] Fett, Daniel and Yasuda, Kristina and Campbell, Brian. *Selective disclosure for JWTs (SD-JWT)*. (2025). https://datatracker.ietf.org/doc/draft-ietf-oauth-selective-disclosure-jwt/22/

[19] *Regulation (EU) 2024/1183 – Establishing the European Digital Identity Framework (eIDAS 2)*. (2024). https://eur-lex.europa.eu/eli/reg/2024/1183/oj

[20] W3C. *Verifiable credentials data model*. (2025). https://www.w3.org/TR/vc-data-model-2.0/

[21] European Commission / European Digital Identity Cooperation Group. *The European Digital Identity Wallet: Architecture and Reference Framework*. (2023). https://digital-strategy.ec.europa.eu/en/library/european-digital-identity-wallet-architecture-and-reference-framework

[22] arkworks contributors. *arkworks zkSNARK ecosystem*. (2022). https://arkworks.rs

[23] iden3. *snarkjs*. https://github.com/iden3/snarkjs

[24] Microsoft. *Crescent*. https://github.com/microsoft/crescent-credentials

[25] Fraunhofer-AISEC. *libbbs*. https://github.com/Fraunhofer-AISEC/libbbs

[26] Circom Developers. *Circom 2 documentation*. https://docs.circom.io/

[27] Google. *longfellow-zk*. (2025). https://github.com/google/longfellow-zk

[28] European Commission. *EU Digital Identity Wallet Home*. (2025). https://ec.europa.eu/digital-building-blocks/sites/spaces/EUDIGITALIDENTITYWALLET/pages/694487738/EU+Digital+Identity+Wallet+Home

[29] alysyans and the EUDI Wallet Team. *Cryptographers' feedback on the EU Digital Identity's ARF (Issue #200)*. (2024). https://github.com/eu-digital-identity-wallet/eudi-doc-architecture-and-reference-framework/issues/200

[30] Decentralized Identity Foundation. *The BBS signature scheme*. (2025). https://identity.foundation/bbs-signature/draft-irtf-cfrg-bbs-signatures.html

Verificaciones de perfiles digitales: construcción de perfiles en entornos online mediante OSINT

Antonio Calvo Picón
Universidad de Extremadura
Centro Universitario de Mérida
Mérida, España
ORCID: 0009-0008-1815-7686
acalvopic@unex.es

José Carlos Sancho Núñez
Universidad de Extremadura
Centro Universitario de Mérida
Mérida, España
ORCID: 0000-0002-4584-6945
jcsanchon@unex.es

Luis Jesús Arévalo Rosado
Universidad de Extremadura
Centro Universitario de Mérida
Mérida, España
ORCID: 0000-0001-7360-2692
ljarevalo@unex.es

***Resumen*—La verificación de perfiles digitales en plataformas online es un componente esencial para mitigar el fraude, fomentar la confianza entre usuarios y reforzar la ciberseguridad en entornos abiertos. Este aporte presenta un sistema modular basado en fuentes Open Source Intelligence (OSINT) que permite construir y evaluar perfiles digitales mediante un modelo gráfico de evidencias.**

Se propone una arquitectura extensible que parte de atributos mínimos (como alias, teléfono o dominio web) y los expande mediante herramientas OSINT especializadas. A partir de estas evidencias, se genera un grafo que refleja la coherencia y trazabilidad del perfil, asignando una puntuación interpretativa que estima su consistencia.

El enfoque se valida mediante un prototipo funcional aplicado a casos reales de plataformas de alquiler vacacional, demostrando su utilidad como sistema de apoyo en procesos de verificación digital. Este sistema busca complementar mecanismos tradicionales de reputación y control, aportando transparencia, explicabilidad y trazabilidad al análisis de identidades digitales.

***Index Terms*—OSINT, verificación de perfiles, identidad digital, confianza, trazabilidad, RECSI**

I. Introducción

El crecimiento del fraude digital en plataformas de compraventa, alquiler vacacional y redes sociales ha evidenciado la necesidad de validar no solo el contenido de los anuncios, sino también la identidad y legitimidad de los usuarios que los publican. En entornos donde la interacción entre desconocidos es frecuente, contar con mecanismos verificables que permitan estimar la credibilidad de un perfil resulta clave para reducir riesgos y reforzar la confianza digital.

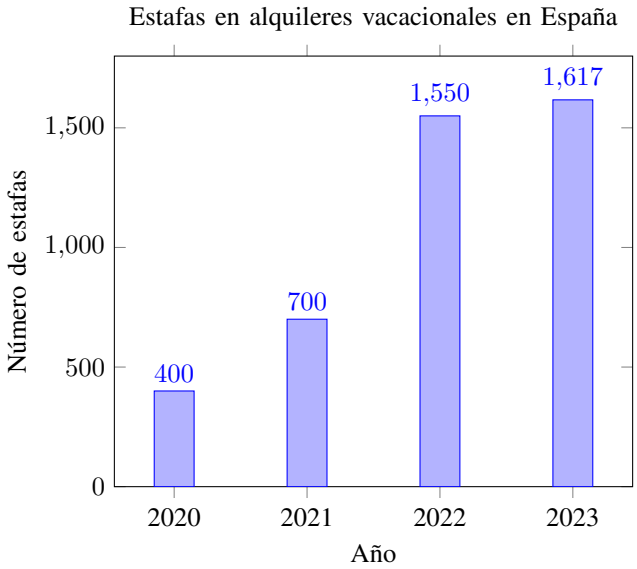


Figura 1: Crecimiento de estafas en alquiler vacacional (2020–2023), según datos de ABC España [1].

Este trabajo aborda este problema mediante la construcción estructurada de perfiles digitales a partir de evidencias públicas, integrando diversas fuentes OSINT para analizar la coherencia observable de cada identidad. La propuesta se valida sobre perfiles reales obtenidos de plataformas abiertas, mostrando su utilidad como apoyo a procesos de verificación digital.

Los principales objetivos que aborda este trabajo son:

- Diseñar una arquitectura modular para la construcción de perfiles digitales basada en evidencias OSINT.
- Implementar un sistema funcional que permita explorar y correlacionar atributos a partir de fuentes abiertas.
- Definir un modelo gráfico que represente relaciones y evidencias de forma explicable.
- Proponer una métrica de consistencia que permita interpretar el nivel de credibilidad de un perfil digital.

II. Estado del arte

II-A. Identidad digital y trazabilidad OSINT

La identidad digital puede entenderse como el conjunto de atributos, huellas y relaciones que una entidad genera en entornos online [2]. La trazabilidad digital, clave en procesos

de atribución o verificación, se apoya en la correlación de evidencias procedentes de múltiples fuentes abiertas [3]. Informes recientes de INCIBE y ENISE [4], [5] destacan el papel creciente de OSINT para validar identidades y reforzar la confianza en plataformas digitales. sistemáticas y verificables.

II-B. Técnicas de verificación automática de perfiles

Las investigaciones recientes muestran tres enfoques principales:

- **Modelos heurísticos.** Utilizan reglas manuales y patrones explícitos, como coincidencias entre correo, alias o teléfono. Son transparentes, pero poco tolerantes a variabilidad. En línea con los análisis de Christou [6], su limitación principal es la falta de adaptabilidad.
- **Modelos de aprendizaje automático.** Empleados en múltiples investigaciones de perfilado digital, como las de De et al. [7], permiten identificar anomalías en comportamiento, relaciones o contenidos generados por el usuario. Incluyen clasificadores tradicionales (SVM, Random Forest) y técnicas de clustering o embeddings.
- **Representación basada en grafos.** Las identidades se modelan como redes de nodos conectados mediante relaciones verificables. Este enfoque se fundamenta en trabajos como los de Zhang et al. sobre grafos aplicados a análisis de relaciones complejas [8], mostrando especial eficacia para correlacionar identidades dispersas.

II-C. Modelos híbridos y sistemas de scoring

La tendencia reciente se dirige hacia modelos híbridos que combinan reglas deterministas, análisis estadístico y técnicas de Machine Learning (ML), proporcionando un scoring interpretable que refleje consistencia, redundancia y coherencia entre evidencias. Según López-Rojas (2020) en su análisis de fraude en plataformas digitales, los modelos híbridos permiten equilibrar precisión y explicabilidad [9].

Estos sistemas de scoring se están viendo reforzados por técnicas de explicabilidad algorítmica (XAI), destacadas en informes de Gartner sobre tecnologías de confianza digital [10]. La XAI resulta esencial en contextos donde las decisiones afectan a usuarios reales, exigiendo mecanismos claros para justificar por qué un perfil se considera fiable o sospechoso.

II-D. Limitaciones actuales y líneas de mejora

Se identifica:

- Falta de normalización de fuentes OSINT (Ferreira, 2024 [2]).
- Trazabilidad insuficiente de decisiones automáticas.
- Dependencia de APIs o plataformas externas con restricciones cambiantes.
- Tensiones entre privacidad y detección precisa (AEPD, 2020 [11]).
- Ausencia de métricas universalmente aceptadas para "fiabilidad digital".

Las principales líneas de avance incluyen:

- Modelos explicables basados en grafos (XAI + Graph Learning).
- Uso de blockchain para registrar evidencias, siguiendo propuestas como la de Zyskind y Pentland [12].
- Ontologías semánticas e interoperabilidad entre fuentes heterogéneas.

II-E. Principios FAIR aplicados a OSINT

El marco FAIR (Wilkinson et al., 2016) [13] se está adoptando progresivamente en OSINT debido a su capacidad para mejorar la reproducibilidad y la trazabilidad de datos. En el contexto de verificación digital:

- **Findable:** metadatos y evidencias bien indexadas.
- **Accessible:** acceso regulado y conforme a RGPD.
- **Interoperable:** uso de estándares como JSON-LD o RDF.
- **Reusable:** evidencias documentadas para auditoría.

III. Arquitectura del sistema: construcción estructurada de perfiles digitales

El sistema propuesto adopta una arquitectura modular orientada a la construcción explicable de perfiles digitales, partiendo de evidencias OSINT mínimas y extendiéndose mediante un modelo gráfico ponderado. La arquitectura está diseñada para facilitar la integración de herramientas existentes, permitir la trazabilidad de cada paso y garantizar una evaluación progresiva de la consistencia del perfil generado. Esta aproximación se alinea con las prácticas descritas por Bazzell (2021) en *Open Source Intelligence Techniques* [14], donde se enfatiza la automatización y correlación sistemática de evidencias públicas.

La construcción del perfil se inicia a partir de uno o más atributos básicos disponibles en plataformas online (por ejemplo, `teléfono`, `URL`, `username`). A partir de estos elementos semilla, se ejecuta un proceso de expansión dirigido que explora múltiples fuentes públicas (sitios corporativos, redes sociales, buscadores inversos, etc.) mediante módulos especializados (scrapers, APIs o herramientas OSINT externas). Este enfoque coincide con los principios metodológicos expuestos por Christou (2019), quien define OSINT como "el arte de reunir datos dispersos en fuentes públicas para generar inteligencia aplicable" [6].

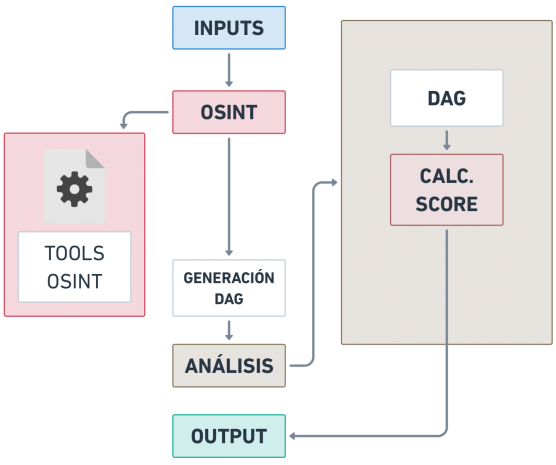


Figura 2: Pipeline modular del sistema propuesto para la verificación de perfiles digitales mediante fuentes OSINT.

Cada evidencia descubierta se modela como un nodo en un grafo dirigido, cuyas aristas representan relaciones verificables o asociaciones funcionales. Este modelo gráfico ha sido empleado previamente en estudios centrados en estructuras semánticas complejas [8]. En el presente caso, permite

representar tanto la densidad de nodos como la coherencia estructural del perfil, identificando redundancias, profundidad de verificación y consistencia entre fuentes heterogéneas.

III-A. Entrada flexible y expansión basada en heurística

El sistema admite múltiples tipos de entrada:

- `Username` o alias digital
- `Teléfono` asociado a un perfil
- `Enlace web` o dominio detectado

A partir de estos atributos iniciales, se ejecuta un proceso de enriquecimiento mediante módulos adaptables que exploran conexiones relevantes. Este proceso puede incluir:

- Scraping dirigido sobre sitios web corporativos o dominios conocidos.
- Consultas a herramientas OSINT como WhatsMyName, urlscan.io, HaveIBeenPwned, entre otras.
- Resolución de conexiones redundantes (por ejemplo, una misma web detectada desde varias fuentes).

Cada conexión descubierta se evalúa en función de tres ejes:

- Profundidad relativa dentro del grafo (distancia al nodo raíz)
- Fiabilidad de la fuente (clasificada como oficial, social o no verificada)
- Coherencia semántica respecto a nodos previamente identificados

III-B. Modelo gráfico explicable y puntuación interpretativa

El grafo construido permite aplicar una puntuación acumulativa basada en su estructura observable. Esta puntuación no constituye una medida absoluta de veracidad, sino un indicador trazable de consistencia digital, útil para diferenciar perfiles con huella sólida de aquellos con señales de incoherencia. Estudios previos han empleado métricas similares basadas en la coherencia observacional como aproximación a la confianza [7].

Cada arista del grafo incorpora un peso asociado a la solidez de la relación. Asimismo, la presencia de evidencias redundantes (por ejemplo, una dirección coincidente en web y red social) se considera un refuerzo estructural de la coherencia observada.

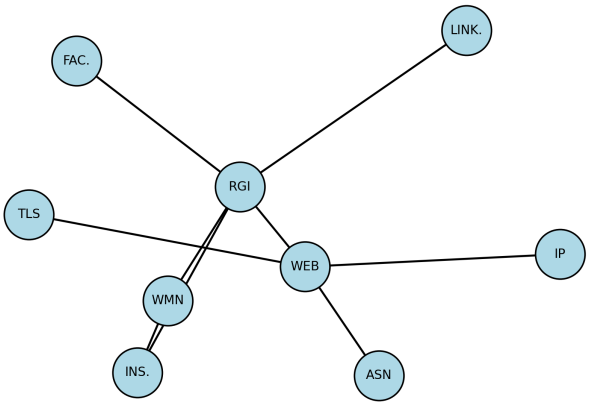


Figura 3: Ejemplo de construcción de grafo y asignación ponderada de relaciones entre nodos.

El sistema está diseñado para ampliarse mediante nuevos módulos de entrada (como reconocimiento de imágenes, perfiles visuales o firmas digitales) y para integrarse con fuentes OSINT adicionales, configurando un ecosistema verificable y adaptable a escenarios reales de riesgo.

III-C. Persistencia verificable mediante blockchain

Como línea futura, se plantea la integración de mecanismos basados en blockchain para aportar una capa de persistencia verificable al grafo. Esta posibilidad ha sido explorada en investigaciones que proponen blockchain como infraestructura descentralizada para la protección de identidades digitales [12] o para sistemas de confianza en entornos urbanos inteligentes [15].

Una implementación potencial consistiría en modelar las conexiones entre nodos como transacciones registradas en una red blockchain, donde cada arista represente una relación verificada, inmutable y trazable. Alternativamente, podría serializarse la estructura completa del grafo en un formato estándar (por ejemplo, JSON-LD o RDF) y registrar su hash en una red blockchain pública o privada.

Este mecanismo permitiría:

- Inmutabilidad: el grafo no puede modificarse sin invalidar su huella.
- Unicidad: perfiles con estructuras idénticas comparten el mismo hash, facilitando la detección de duplicidades.
- Verificabilidad descentralizada: cualquier tercero puede comprobar la autenticidad del grafo sin depender de un servicio centralizado.

La incorporación de una capa blockchain reforzaría la auditabilidad del sistema y abriría la puerta a futuras integraciones con plataformas antifraude, mecanismos de reputación digital o servicios de verificación en entornos de alquiler y compraventa online.

IV. Implementación del sistema y herramientas OSINT

La implementación funcional del sistema se apoya en herramientas OSINT consolidadas, seleccionadas en función de su capacidad para extraer evidencia relevante a partir de atributos mínimos (alias, número de teléfono, correo electrónico o dominios web). A continuación, se describen los tipos de datos empleados como entrada y las herramientas utilizadas para su enriquecimiento dentro del pipeline.

IV-A. Tipos de datos y herramientas principales

En entornos como plataformas de alquiler o compraventa, los atributos más comunes empleados como punto de partida son los siguientes:

- Teléfono
- Alias o *username*
- Dominio o URL web
- Correo electrónico (opcional)

Para cada tipo de entrada se selecciona una herramienta OSINT principal, evaluando su cobertura, fiabilidad y facilidad de automatización.

1. Teléfono: `Sync.me`
Justificación:

- Permite obtener una ubicación estimada (ciudad/país) asociada al número.
- Detecta posibles vínculos con empresas o reportes de spam.
- Es accesible a través de web sin necesidad de registro.

Alternativas consideradas: `PhoneInfoga`, `TrueCaller`.

Uso en el sistema:

- Se consulta el número mediante *scraping* controlado.
- Se generan nodos de localización, operador o empresa cuando la información está disponible.

2. Alias / Username: `WhatsMyName`

Justificación:

- Permite buscar un alias en más de 200 plataformas de redes sociales.
- Presenta mayor cobertura y actualización comunitaria que alternativas similares.
- Es ligero y fácilmente integrable en entornos automatizados.

Alternativas consideradas: `Maigret`, `Sherlock`.

Uso en el sistema:

- Se ejecuta una consulta sobre el alias detectado.
- Se generan nodos correspondientes a los perfiles sociales identificados (por ejemplo, Facebook o LinkedIn).

3. Dominio o URL: `urlscan.io`

Justificación:

- Proporciona información estructurada sobre IPs, certificados TLS, proveedores y tecnologías utilizadas.
- Permite analizar un dominio sin visitarlo directamente.
- Los resultados están optimizados para análisis OSINT.

Alternativas consideradas: `BuiltWith`, `Whois`.

Uso en el sistema:

- Se lanza un escaneo del dominio especificado.
- Se generan nodos asociados, incluyendo IP, ASN, tecnologías detectadas y autoridad del certificado.

4. Correo electrónico: `HaveIBeenPwned`

Justificación:

- Indica si un correo ha sido expuesto en filtraciones previas.
- No requiere autenticación y mantiene alta reputación como fuente.
- Su salida permite generar nodos de riesgo asociados al perfil digital.

Alternativas consideradas: `EmailRep`, `Hunter.io`.

Uso en el sistema:

- Se consulta si el correo ha sido comprometido.
- En caso afirmativo, se añade un nodo indicando exposición previa.

IV-B. Modelo de puntuación basada en profundidad

El modelo de verificación se fundamenta en la construcción de un grafo dirigido a partir de evidencias OSINT. Cada nodo representa una evidencia digital descubierta (por ejemplo, un perfil social, un dominio web o una dirección), mientras que las aristas reflejan relaciones verificables entre dichas evidencias. Estas conexiones pueden derivarse de coincidencias explícitas (como un número de teléfono presente en varias plataformas) o de asociaciones inferidas (como un dominio vinculado a una red social mediante metadatos).

Grafo dirigido acíclico (DAG). El sistema modela el grafo como un *Directed Acyclic Graph* (DAG), dado que no se permiten ciclos entre evidencias. Esta estructura posibilita un recorrido jerárquico desde el nodo raíz (dato inicial) hasta los nodos hoja (evidencias finales).

Puntuación basada en el camino más largo. La *consistencia observable* del perfil se estima mediante el cálculo del **camino más largo** del DAG, definido como la secuencia de nodos y aristas que maximiza la longitud acumulada. Esta métrica refleja la profundidad y robustez de la cadena de evidencias.

El cálculo de la puntuación se normaliza en una escala de 0 a 10:

$$\text{Score}_{\text{raw}} = \sum_{i=1}^{n} w_i \tag{1}$$

$$\text{Score}_{\text{final}} = \frac{\text{Score}_{\text{raw}}}{\text{Score}_{\text{máx}}} \cdot 10 \tag{2}$$

Donde:

- w_i es el peso asignado a cada arista del camino más largo (según la fiabilidad de la fuente, redundancia o tipo de evidencia).
- $\text{Score}_{\text{raw}}$ es la suma acumulada de los pesos del camino.
- $\text{Score}_{\text{máx}}$ representa el valor máximo teórico definido a priori.

Este modelo favorece perfiles con mayor profundidad (más saltos entre evidencias), coherencia estructural y múltiples relaciones independientes, y penaliza grafos planos, poco conectados o con evidencias débiles.

Ejemplo. Si el camino más largo contiene cinco evidencias con pesos $w = [1{,}0, 1{,}0, 0{,}8, 0{,}7, 1{,}0]$, entonces:

$$\text{Score}_{\text{raw}} = 4{,}5 \quad \text{y, si} \quad \text{Score}_{\text{máx}} = 7{,}0, \quad \Rightarrow \quad \text{Score}_{\text{final}} =$$

Este resultado se interpreta como un perfil con consistencia media-alta según la trazabilidad de las evidencias y su encadenamiento estructural.

Ventajas del enfoque:

- Proporciona una métrica cuantitativa derivada directamente de la estructura del grafo.
- Permite comparar perfiles de forma consistente y reproducible.
- Reforza la explicabilidad, al visualizar explícitamente la cadena de evidencias y su profundidad.

V. Resultados y discusión

Para evaluar la aplicabilidad del sistema propuesto, se implementa un prototipo funcional que construye automáticamente perfiles digitales a partir de atributos semilla utilizando fuentes OSINT. A continuación, se presentan dos de los casos de estudio realizados.

V-A. Caso 1: RGI Inmuebles

Se parte del alias `rgiinmuebles`, correspondiente a una entidad presente en plataformas de alquiler. A partir de dicho alias se aplican diversas herramientas OSINT:

- **WhatsMyName (WMN):** se confirma la existencia del alias en Instagram.
- **urlscan.io:** se extraen la IP asociada, el ASN del proveedor y los datos del certificado TLS de la web oficial.

Toda la evidencia se modela como un grafo dirigido, tal como se muestra en la Figura 4. Se observan nodos redundantes (por ejemplo, el perfil de Instagram aparece en múltiples fuentes), lo que refuerza la consistencia del perfil analizado.

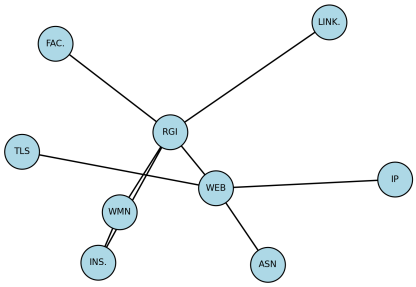


Figura 4: Grafo de verificación digital generado para el perfil `rgiinmuebles`.

Los nodos del grafo incluyen evidencias heterogéneas tales como perfiles en redes sociales (INS, FAC, LNK), registros técnicos (IP, ASN, TLS), datos de contacto (TEL, MAIL) y dominios asociados. Las abreviaturas empleadas representan categorías funcionales de evidencia más que códigos específicos, por lo que su interpretación puntual no es necesaria para comprender la estructura global del grafo.

La puntuación de consistencia se estima mediante una métrica basada en la longitud del camino más largo del grafo (DAG) y la presencia de evidencias redundantes, proporcionando así una evaluación interpretativa del grado de coherencia digital.

V-B. Caso 2: Quinta Avenida Inmobiliaria

En este caso, se introducen tres atributos de entrada: nombre de la entidad (`RK Quinta Avenida`), teléfono (`960 37 68 64`) y correo electrónico (`info@quintaavenidainmobiliaria.com`).

- **WhatsMyName:** se identifican perfiles en Idealista, Instagram, Facebook y LinkedIn.
- **Sync.me:** se confirma la localización del número de teléfono en Valencia.
- **urlscan.io:** se obtiene la IP del servidor web, ASN, proveedor TLS y metadatos del dominio.
- **HaveIBeenPwned:** no se detectan filtraciones asociadas al correo electrónico.
- **LinkedIn y Facebook:** se extraen evidencias adicionales como dirección física, dominios secundarios y contactos vinculados.

Toda la información recopilada se representa en el grafo de la Figura 5, donde se destaca una mayor densidad de nodos y relaciones frente al caso anterior.

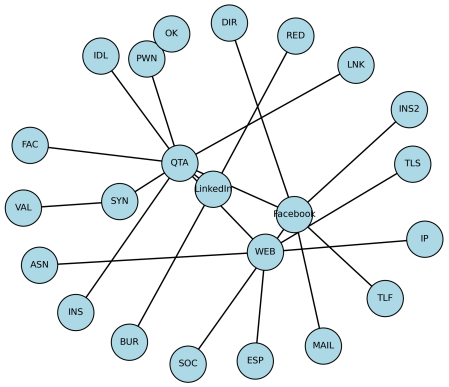


Figura 5: Grafo de evidencias digitales para el perfil `Quinta Avenida Inmobiliaria`.

Este perfil presenta una mayor riqueza de evidencias, algunas cruzadas entre distintas fuentes. La vinculación del dominio principal con otros dominios asociados (como `sociosrk.com`), así como la presencia activa en redes, permite asignar una puntuación de consistencia algo más elevada al caso anterior.

V-C. Indicadores preliminares de viabilidad

Además de los dos casos de estudio descritos, se ha realizado una evaluación preliminar sobre una muestra de 31 perfiles digitales, compuesta por 27 perfiles reales obtenidos desde plataformas abiertas y 4 perfiles artificiales o sospechosos generados con fines de prueba. Este análisis busca identificar patrones estructurales que distingan la presencia digital legítima de aquellas cuentas con una huella intencionalmente reducida o inconsistente. Entre los perfiles simulados, se incluye uno adicional que pertenece a un perfil real cuyo comportamiento es sospechoso (por la recepción de mensajes tipo spam sin coherencia temática), aunque no puede confirmarse su falsedad. Por cautela, no se proporcionan detalles específicos, pero se contabiliza en la categoría de simulados.

La Tabla I resume métricas estimadas obtenidas en pruebas realizadas:

Tabla I: Resumen de características en perfiles reales y simulados.

Código	Perfiles reales (n=27)	Perfiles simulados (n=4)
PM	7.4	3.5
DFO	Alta	Baja
CG	Alta	Baja o discontinua
REN (densidad)	Estable	Baja o fragmentada
PE	Profesional / comercial	Spam / stalking / incierto

Tabla II: Significado de las abreviaturas utilizadas en la Tabla I.

Código	Descripción
PM	Presencia media (número promedio de nodos por perfil)
DFO	Diversidad de fuentes OSINT
CG	Coherencia gráfica en el grafo resultante
REN	Relación entre nodos (densidad estructural)
PE	Propósito estimado del perfil digital

Las métricas muestran que los perfiles reales presentan, en promedio, 7.4 nodos relacionados, incluyendo redes sociales, web corporativa, registros técnicos o perfiles comerciales, con conexiones claras y coherentes entre evidencias (nodos). Por el contrario, los perfiles simulados muestran una estructura simplificada, con pocos nodos y escasa o nula interrelación entre ellos.

Estos resultados, si bien preliminares, refuerzan la hipótesis de que la topología del grafo puede actuar como un indicador interpretativo de legitimidad digital. Se propone extender este análisis en futuras etapas, con un conjunto más amplio y variado de perfiles para validar cuantitativamente esta relación.

V-D. Discusión

Los resultados muestran que el sistema es capaz de:

- Construir perfiles digitales explicables a partir de entradas mínimas.
- Integrar información heterogénea proveniente de múltiples fuentes OSINT.
- Detectar relaciones directas, indirectas y redundantes entre evidencias.
- Generar una métrica estructural basada en la topología del grafo que actúa como un indicador interpretable de consistencia.

Los casos mostrados y los indicadores de viabilidad demuestran la utilidad práctica del sistema en contextos reales como plataformas de compraventa, validación de identidades o análisis de fraude digital, permitiendo una verificación progresiva y trazable basada en evidencias públicas.

VI. Consideraciones éticas y privacidad

El uso de fuentes abiertas en verificación digital implica desafíos de privacidad y responsabilidad ética. Aunque los datos son públicos, su combinación y análisis puede considerarse tratamiento de datos personales según el RGPD [16] y la LOPDGDD.

En este proyecto se aplican principios de minimización, anonimización y uso limitado al contexto de investigación, evitando la identificación directa de personas y restringiendo el acceso a información sensible. Se emplea un modelo de filtrado previo, inspirado en el *Legacy Funnel System*, para controlar datos potencialmente protegidos antes de su procesamiento.

Asimismo, la reutilización de fuentes públicas debe respetar derechos de propiedad intelectual y condiciones de uso, reforzando la necesidad de un enfoque responsable y proporcional en la aplicación de técnicas OSINT.

VII. Conclusiones y trabajo futuro

Se presenta una infraestructura modular para la verificación de identidades digitales mediante grafos ponderados que representan y correlacionan evidencias OSINT de manera transparente y trazable. Los resultados muestran que, a partir de atributos mínimos, es posible construir perfiles digitales estructurados que permiten identificar niveles de coherencia y autenticidad, ofreciendo soporte explicable para procesos de verificación en entornos abiertos.

Como trabajo futuro se propone: (1) ampliar el sistema con más atributos iniciales para aumentar la cobertura y profundidad del grafo; (2) integrar técnicas de aprendizaje automático para analizar patrones y detectar comportamientos consistentes o anómalos; y (3) explorar mecanismos basados en blockchain para gestionar y auditar de forma inmutable las identidades y evidencias. Estas mejoras reforzarán la solidez y aplicabilidad del sistema en entornos de confianza digital.

Agradecimientos

Este trabajo ha sido desarrollado en el marco del Grupo de Investigación Ingeniería de Medios perteneciente a la Universidad de Extremadura. La actividad ha sido cofinanciada al 85 % por la Unión Europea, Fondo Europeo de Desarrollo Regional, y la Junta de Extremadura. Autoridad de Gestión: Ministerio de Hacienda. Número de expediente de la ayuda: GR24170.

El trabajo científico se desarrolla del Proyecto C108/23 "Detección de falsificación de documentos de identidad mediante técnicas de visión por computador e inteligencia artificial", del Instituto Nacional de Ciberseguridad (INCIBE), dentro del Plan de Recuperación, Transformación y Resiliencia financiado por la Unión Europea (Next Generation EU).

Referencias

[1] ABC España. (2023) Más de 1.600 estafas en alquileres vacacionales en 2023. Consultado: marzo 2025. [Online]. Available: https://www.abc.es/sociedad/estafas-alquileres-vacacionales.html

[2] J. Ferreira, *OSINT aplicado a redes sociales: Guía práctica para investigación digital*. Independently Published, 2024.

[3] L. Gómez and J. Pérez, "Aplicaciones de osint en ciberseguridad: métodos y retos recientes," in *Congreso Iberoamericano de Seguridad Digital*. Ciberseguridad Iberoamericana, 2022.

[4] INCIBE, "Informe anual de ciberseguridad 2023," Instituto Nacional de Ciberseguridad (INCIBE), Tech. Rep., 2023.

[5] ENISE, "Memoria del encuentro internacional de seguridad de la información (enise) 2024," Instituto Nacional de Ciberseguridad (INCIBE), Tech. Rep., 2024.

[6] G. Christou, *Open Source Intelligence: An Introduction*. Springer, 2019.

[7] S. De and A. Jindal, "User profiling using multi-source osint data," *International Journal of Information Security*, vol. 12, no. 3, pp. 115–129, 2018.

[8] W. Zhang, L. Chen, and Y. Zhou, "Graph-based models for complex relationship analysis: A systematic review," *ACM Computing Surveys*, vol. 53, no. 4, pp. 1–35, 2020.

[9] O. López-Rojas, *Fraude digital y modelos híbridos de detección*. Editorial Ciberseguridad, 2020.

[10] Gartner, "Top Strategic Technology Trends for 2023: AI Trust, Risk and Security Management," Oct 2022, iD G00774612. [Online]. Available: https://www.gartner.com/en/documents/4020030

[11] Agencia Española de Protección de Datos (AEPD), "Estudio sobre las tecnologías de lucha contra la COVID-19 y sus implicaciones para la privacidad," May 2020, comunicado de prensa y estudio asociado. [Online]. Available: https://www.aepd.es/prensa-y-comunicacion/notas-de-prensa/aepd-publica-estudio-tecnologias-covid

[12] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," in *2015 IEEE Security and Privacy Workshops*, 2015, pp. 180–184.

[13] M. D. Wilkinson *et al.*, "The fair guiding principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 160018, 2016.

[14] M. Bazzell, *Open Source Intelligence Techniques*, 10th ed. Independently Published, 2021.

[15] F. Xu, Z. Li, and K. Wu, "Blockchain-based trust management in smart cities: A survey," *IEEE Access*, vol. 7, pp. 97 452–97 466, 2019.

[16] "Reglamento (ue) 2016/679 del parlamento europeo y del consejo," Diario Oficial de la Unión Europea, 2016, reglamento General de Protección de Datos (RGPD).

Niños y adultos mayores ante la vulnerabilidad digital: la urgencia de programas de capacitación para su protección

José Agustín Portas Yáñez, Ana Lucila Sandoval Orozco, Luis Javier García Villalba*

Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {jportas, asandov, javierv}@ucm.es

***Resumen*—El crecimiento del uso de dispositivos móviles y del entorno digital ha incorporado a grupos con limitada experiencia tecnológica, especialmente niños y adultos mayores. Aunque pertenecen a etapas de vida opuestas, ambos comparten escasas competencias digitales y de ciberseguridad, lo que los vuelve especialmente vulnerables frente a las amenazas presentes en Internet. Este trabajo propone el diseño de programas formales de capacitación adaptados a las características cognitivas, sociales y tecnológicas de estas poblaciones, con el fin de desarrollar habilidades básicas para un uso seguro de las herramientas digitales, reconocer riesgos en línea y promover una cultura de responsabilidad digital. El fortalecimiento de estas capacidades no solo protege a los individuos, sino que también contribuye a construir una ciudadanía digital más informada, inclusiva y resiliente ante los desafíos de una sociedad cada vez más conectada.**

***Index Terms*—Ciberseguridad; alfabetización digital; ciudadanía digital; programas de capacitación; inclusión tecnológica; protección infantil; adultos mayores**

I. Introducción

Con frecuencia y en una tendencia de aumento sostenido en los últimos años la digitalización de la población trasciende barreras que antes eran limitantes, como lo era el estatus socioeconómico y la edad, tanto la edad de primer acercamiento a los dispositivos electrónicos, como el uso por parte de los ancianos que nunca habrían logrado (o necesitado) aprender sobre ellos. Conforme el umbral de edades de uso de las redes y dispositivos de conexión a las mismas se ensancha, aumenta la complejidad del análisis de problemáticas sobre su utilización, sin embargo, una constante es la creciente zozobra sobre los riesgos a los que se enfrentan dichos grupos poblacionales frente al desmedido aumento y profesionalización de las amenazas a las que podrían estar expuestos con el uso incipiente que las propias necesidades de digitalización les fuerzan a tener [1].

Como se detalla en los últimos estudios publicados de incidencias de cibercrímenes en España, existen tendencias al alza en la incidencia de los delitos informáticos, siendo este incremento más pronunciado desde la pandemia por COVID-19. Podemos afirmar que, desde el 2020, la migración hacia la digitalización total de muchos de los servicios habituales ha sufrido un incremento sumamente importante en aplicaciones como educación, compras, tarjetas de fidelización y beneficios de comercios, entre otros. Aunado a ello, el tiempo de implementación ha sido breve, por lo que los usuarios han tenido muy poco tiempo de aprendizaje, con prácticamente nula capacitación durante su inmersión en el mundo digital [2].

En el ámbito académico, para muchos de los menores se incluyen como requisitos la capacidad de acceso a recursos digitales, tanto dentro como fuera de las aulas. Esto requiere dispositivos con acceso a internet, como teléfonos móviles, tabletas y ordenadores. No se encuentran estudios especializados sobre su aprendizaje o utilización, y tampoco se puede comprobar que posean conocimientos sobre seguridad y prevención.

En el otro extremo etario, los adultos mayores se ven obligados al uso de dispositivos móviles para sumarse a las actividades cotidianas digitalizadas. Su capacitación suele ser mínima, enfocada únicamente en tareas específicas [3].

II. Objetivo

Analizar la separación etaria de los usuarios principiantes de las Tecnologías de la Información y Comunicaciones, verificando las dificultades de uso para cada segmento, así como los principales riesgos a los que se encuentran expuestos, sugiriendo alguna forma de mitigación de los mismos.

II-A. Objetivos Particulares

- Realizar una búsqueda documental sobre los cambios en las edades de primer contacto con tecnologías móviles e internet.
- Identificar los principales delitos informáticos basados en su ocurrencia al tiempo de elaboración del presente documento.
- Encontrar relaciones entre dichos delitos y los factores de riesgo que conducen a su afectación en los grupos etarios más vulnerables.
- Elaborar recomendaciones útiles sobre formas de mitigación.

III. Metodología

Se realizó un proceso estructurado de recopilación, análisis y síntesis de información relacionada con los riesgos de ciberseguridad que afectan a poblaciones en edades tempranas y tardías, fundamentado en literatura académica.

El estudio emplea un enfoque cualitativo y descriptivo permitiendo observar con más detalle cómo se relacionan estos grupos con los dispositivos digitales y qué tipos de riesgos suelen enfrentar cuando aún no tienen experiencia con el uso de la tecnología. La investigación se sustentan en el uso de fuentes secundarias, es decir, investigaciones, artículos y análisis realizados previamente por especialistas en el tema. La selección del material se centró únicamente en aquellas publicaciones que aportaran evidencia o reflexión sobre los siguientes ejes temáticos de: (i) primer contacto con dispositivos móviles y tecnologías digitales y (ii) amenazas y vulnerabilidades asociadas a la seguridad digital en usuarios con escasa experiencia tecnológica. Una vez detectados, con el apoyo de la literatura las necesidades particulares de los usuarios con mayor vulnerabilidad, se propondrán recomendaciones para fortalecer las debilidades hayadas.

IV. Análisis poblacional y segmentación de grupos con mayor vulnerabilidad

El ciberespacio se puede constituir como un ente vivo y en constante crecimiento y desarrollo, ya que día a día se incorporan a su universo de población miles de nuevos usuarios, tan solo en el 2024 experimentó un crecimiento de 136 millones de usuarios, lo cual le concede un acceso a internet a 67,9 % de la población mundial [4].

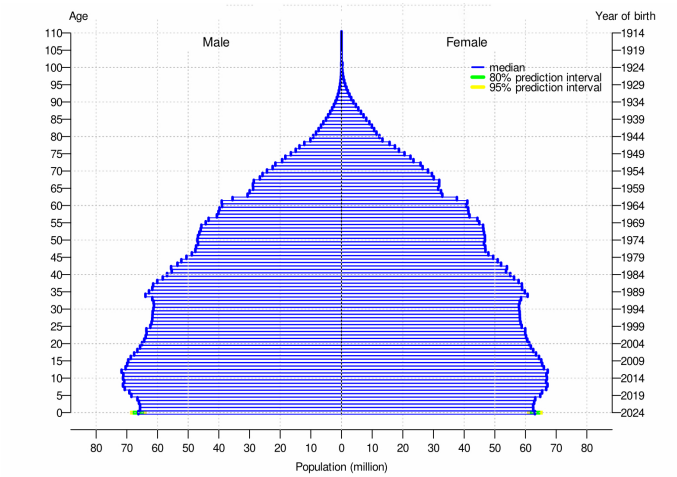


Figura 1: Pirámide de población mundial por edad y sexo en 2025 (datos de la ONU) [5].

Aunado a esto, los sectores productivo, educativo y gubernamental, añaden cada vez un mayor número de servicios, trámites y posibilidades de interacción a través del mundo digital, facilitando paulatinamente la búsqueda, oferta y puesta en marcha de servicios, esto con la finalidad de ahorrar costos y tiempos en atención presencial.

La digitalización de cada vez más actividades cotidianas resulta muy benéfica para el grueso de la población, ya que las personas, en edades medias (entre 18 a 60 años) se presumen mayoritariamente como usuarios de las tecnologías de información y comunicaciones, desde hace varios años pudiendo ser testigos de su proceso evolutivo, y que han tenido a su vez, algún tiempo considerable para familiarizarse con las interacciones que suceden en el mundo cibernético, así como con las mejoras que se realizan a los servicios disponibles.

Según datos del Departamento de Asuntos Económicos y Sociales de las Naciones Unidas, podemos analizar dos tendencias en el movimiento de la curva poblacional mundial. Por un lado, se encuentra el estrechamiento de dicha curva, que comienza a ser pronunciado a partir de 65 años de edad, y, a pesar de marcar una clara disminución, dicha curva aún mantiene una presencia sostenida, contribuyendo a la población hasta más de 80 y 90 años. Por la otra parte, rozando los 70 millones de habitantes, corresponde a los menores alrededor de los 10 años. Ambos extremos de ella, pueden relacionarse en una problemática particular para cada uno de los grupos, que consiste en su adhesión al mundo digital. Los menores de edad presentan una mente ávida de conocimiento e interacciones rápidas, por lo que su aprendizaje, en cuanto al manejo, de las nuevas tecnologías se realiza con una rapidez impresionante a ojos de los adultos a su cargo, aunque, al ser un aprendizaje autodidacta, se dejan de lado las consideraciones éticas o legales (que no se encuentran al alcance inmediato) como pueden ser, las que se refieren a seguridad, protección y autocuidado.

A su vez, los mayores presentan una problemática de origen distinto, ya que, en su mayoría, han tenido un uso nulo, o muy escaso de los dispositivos electrónicos y se manifiestan aún reticentes y temerosos ante la posibilidad de incorporarlos a su uso cotidiano, motivados, en su mayoría por desconocer el correcto manejo de los mismos. Situación que ha derivado en un importante atraso en cuanto a la adopción de las nuevas tecnologías por su parte, limitándose, a un manejo escaso, originado desde el extremo de “cuando ya no tienen más remedio”.

A pesar del origen diametralmente opuesto de ambas situaciones, podemos señalar un claro nexo de unión entre estas dos perspectivas, y es que ambas posturas carecen de información sobre seguridad informática.

Por una parte, los más pequeños debido a su desconocimiento de los peligros que yacen escondidos detrás de los beneficios y recompensas positivas que acostumbran a obtener; en el otro extremo, los adultos mayores, temerosos de dar cada paso, intuyendo que cualquier error de su parte puede, dejar inoperativo el dispositivo, además del creciente miedo a que les roben información. Confluyen estas poblaciones en el desconocimiento general sobre las amenazas a la ciberseguridad, y que genera, una vulnerabilidad mayor a dichos riesgos.

En cuanto a las amenazas existentes en la internet, debemos hacer la diferencia entre aquellas que “no eligen” a los usuarios que afectarán, como pueden ser virus, worms y malware, por ejemplo; ya que su ataque (infección) se realiza de forma transparente, y sin ninguna relevancia sobre el usuario al que pueda afectar, sino que se realiza entre dispositivos simplemente por el hecho de interactuar entre ellos.

El otro grupo de interés sobre los ataques cibernéticos se distingue por ser aquél cuyos ataques son dirigidos, como lo pueden constituir: fraude cibernético, suplantación de identidad, estafa, etcétera.

En el informe sobre tendencias del cibercrimen en España se categorizan los delitos informáticos en seis categorías

penales basados en los datos oficiales del Ministerio del Interior del 2023. La Tabla I muestra estas categorías junto con la frecuencia de su ocurrencia [2].

Tabla I: Frecuencia y tipología de delitos informáticos en España

Categoría Penal	Frecuencia
Fraude informático	90,5 %
Falsificación informática	3,2 %
Acceso e interceptación ilícita	1,6 %
Delitos Sexuales	0,4 %
Interferencia de datos y en sistema	0,4 %
Contra el honor	0,2 %
Contra la propiedad industrial/intelectual	0,01 %.

Pero, tras realizar un análisis más concienzudo, es correcto afirmar que dicha estadística está formada solamente por aquellos delitos que se encuentran tipificados, y que fueron detectados y correctamente denunciados ante la autoridad pertinente. Desestimando a todos aquellos que por motivos diversos como: desinformación, vergüenza, incapacidad probatoria, desconocimiento, o amenazas no fueron denunciados y siguen sucediendo, y sin duda, incrementando su frecuencia derivándose de la impunidad en la que aún se encuentran.

Adicional a esta reflexión, existen algunas otras amenazas a la seguridad informática que varían en cuanto a planteamiento con los delitos informáticos a los que se asocian, como lo son: Ciberbullying, Grooming, Sexting, Ransomware Sextorsión, Sharenting, Malware, Phishing, Spyware y Doxing [6].

El objetivo se centra únicamente en el análisis individual de cada una de ellas y en la correcta identificación de sus características, con el fin de que los usuarios de los diversos dispositivos conectados a Internet puedan desarrollar sus actividades de manera segura, evitando los distintos obstáculos que puedan presentarse en su entorno digital.

A continuación, se puntualizarán las condiciones particulares de las poblaciones que son el objeto de estudio del presente trabajo

V. Análisis situacional de los Menores de Edad por grupo de edad y acercamiento a las tecnologías

Debido a la diferenciación de condiciones que las distintas etapas del desarrollo emocional y cognitivo que presentan los menores de edad, se analizan a continuación segmentados por etapas educativas: (i) Lactantes y preescolares: 0 a 3 años, (ii) Infantil: 3 a 6 años, (iii) Educación Primaria: 6 a 12 años, (iv) Educación Secundaria: 12 a 16 años, (iv) Bachillerato: 17 y 18 años.

Una vez mencionados dichos niveles, se pueden vislumbrar diferencias claras entre los contextos y necesidades de los menores, pasando desde el desarrollo sensorial (preferentemente alejado de pantallas y dispositivos electrónicos) durante la lactancia y edad preescolar, hasta los requerimientos académicos y socioculturales que los bachilleres pueden constituir [7], [8].

i **Lactantes y preescolares**: Durante la etapa de lactancia y antes de la incorporación al colegio infantil, el desarrollo, tanto físico como intelectual de los individuos resulta crucial para desenvolver en personas sanamente constituidas, en los ámbitos biopsicosociales. Dada la trascendencia de la exploración sensorial y sano desarrollo intelectual y visual de las personas, se desaconseja completamente el uso de dispositivos móviles durante toda esta etapa; aunado al factor de que aún se carece de una comunicación fluida debido a que aún se encuentra en fase de aprendizaje del lenguaje y comunicación oral [9].

ii **Infantil**: Desde los 3 hasta los 6 años se produce una etapa de aprendizaje del entorno, de normas de convivencia, ampliación de vocabulario, incremento de motricidad gruesa y fina, aunado con el desarrollo físico y crecimiento. Puede resultar conveniente propiciar dicho desarrollo con material multimedia, o juegos lúdicos dosificados por los padres o cuidadores de los menores, siempre y cuando los mismos no superen una hora al día, asimismo se recomienda enérgicamente mantener una supervisión constante del uso de los dispositivos [9].

iii **Educación Primaria**: A lo largo de la educación primaria, el aprendizaje en los primeros años se centra en el lenguaje escrito, ampliar conocimientos y la diversificación de los mismos, por lo que, obtener acceso a fuentes de consulta sobre nuevos conocimientos, y la gran variedad de los que son ofrecidos a través de internet, de la mano con su disponibilidad en todo momento, vuelve al acceso a internet una valiosa herramienta para los escolares de educación primaria. De igual forma sucede con los docentes de dicho nivel, quienes a discreción pueden administrar su uso, o solicitar la consulta de alguno de los mismos como labor extramuros para el apoyo académico [9]. Asimismo, dependiendo del entorno social de los niños, puede que requieran en algún punto entre el umbral de edades mencionado, el uso de un teléfono móvil ya sea por requerimientos de comunicación con sus padres o tutores, o para poder cumplimentar los deberes que (principalmente en los grados superiores de la educación primaria) son solicitadas por parte de los docentes. Dicha adición de los menores como usuarios principales de un dispositivo conectado a internet, es que se sugiere que el acceso a los mismos sea paulatino, y supervisado por los padres o tutores de los menores, ya que un uso discriminado de ellos podría colocarlos en un sitio vulnerable a cualquiera de las amenazas existentes en la internet, y que se discutirán más adelante [9].

iv **Educación Secundaria**: Los jóvenes que cursan la Educación Secundaria, prácticamente todos ellos poseen uno o varios dispositivos capaces de conectarse a internet, y son usuarios asiduos a los mismos, de donde la precaución en dichas etapas se debe principalmente a la racionalización de su uso, así como en el cuidado de los contenidos que consumen, y las redes a las que se encuentran expuestos. El contexto etario de los, adolescentes, complica la supervisión cercana por parte de sus padres o tutores guardianes, ya que, con la aparición de la adolescencia, y todos los cambios que esta conlleva, comienza el establecimiento franco de su independencia como individuos y la necesidad de conservar límites entre lo que comunican a sus padres. La recomendación principal para este grupo etario se fundamenta en la concienciación sobre los

riesgos asociados tanto al uso de Internet como a las dinámicas sociales contemporáneas, así como en la regulación del tiempo de exposición a los dispositivos. Se busca, en la medida de lo posible, fomentar que los menores desarrollen sus relaciones interpersonales en entornos ajenos al dominio de los medios electrónicos [10].

- **Bachillerato**: En el tiempo que lleva a los bachilleres esta etapa formativa, se conoce que se encuentran completamente habilitados para el uso de dispositivos móviles y las distintas formas de comunicación a través de la internet, se sabe también que son usuarios frecuentes de muchos de los recursos que se encuentran a disposición para ser consultados. La habilitación tan acelerada para el uso fluido de los dispositivos y destinos a los que son capaces de acceder acelera también la posibilidad de obtener (mediante mecanismos poco legítimos) acceso a contenidos para mayores de edad, así como deriva en un incremento del riesgo que ostentan las amenazas existentes en internet. Durante la mencionada etapa, la recomendación gira en torno a la sensibilización de los posibles riesgos y la cultura de la prevención de estos, así como la capacitación sobre el procedimiento a seguir en caso de detectar alguna amenaza que pueda incurrir en alguno de los delitos en perjuicio de los jóvenes de este grupo etario [11].

VI. Usuarios nuevos sin capacitación tecnológica, en edades post-Laborales

El contexto del adulto mayor se analiza en forma simétrica que los menores, ya que se conoce que conforme los mayores estén más cerca de las etapas laborales (recién jubilados) tendrán más competencias digitales para el uso de dispositivos, y mientras más alejados de la edad productiva (edades más grandes) habrán estado más lejos del uso de los dispositivos electrónicos, así como del acceso a internet, durante algún punto de su etapa laboral, así que es esperable que, al no haberse visto forzados laboralmente a acercarse al uso de las Tecnologías de Información y Comunicaciones, les tengan cierto temor y reticencia a su uso; principalmente generado por el desconocimiento, inexperiencia, así como el gran cambio de paradigma que representan con respecto a las costumbres de los mayores [12].

Del desconocimiento de la tecnología, su uso correcto y la percepción de fragilidad de los dispositivos a ser "inutilizados o rotos" es que mucha de la población adulta mayor, intenta evitar su uso, para evitar la llamada "ansiedad tecnológica" y sus consecuencias [13].

La vida cotidiana y la digitalización paulatina de cada vez más aspectos de esta, comienzan a forzar a los mayores al uso de los dispositivos, tanto para fines de comunicación con familiares y amigos, como para la realización de procedimientos, trámites, compras, etcétera que comienzan a desistir de la realización física/presencial de los mismos. Dicha situación hace que los mayores (principalmente los de edades más próximas a las productivas) estén en búsqueda de apoyo para aprender el correcto manejo de dichas tecnologías.

Las familias suelen ser los elegidos, principalmente para solicitarles apoyo con la comprensión de alguna herramienta o trámite específico, pero el aprendizaje suele darse de manera puntual acerca del procedimiento que se debe seguir únicamente para la función buscada en ese momento, pero sin ir más allá y generar un aprendizaje que pueda ser extrapolado hacia una nueva tarea. No se pueden conocer más datos, ni generalizar ninguna otra métrica sobre los aprendizajes empíricos obtenidos mediante este método.

Por su parte, algunas universidades e instituciones de educación de varios países del mundo han implementado cursos de capacitación para adultos mayores, que abordan, entre otras muchas temáticas, capacitaciones informáticas. Si bien, algunas de ellas van enfocadas al uso de un ordenador, y paquetería de uso general como navegadores web y suites de ofimática, otras de ellas (las menos) van enfocadas a alguna paquetería más avanzada de uso particular.

De acuerdo con datos publicados por el Consejo Superior de Investigaciones Científicas, se encuentra que un total de 71 universidades e instituciones de formación profesional ofrecen programas de capacitación para mayores, distribuidos en el territorio español como se muestra en la Tabla II, que refleja aquellas comunidades autónomas que tienen al menos una institución que ofrece dichos programas, sin embargo, en la mayoría de estas, se ofrecen programas universitarios de varios años, y pocos cursos de capacitación específicos. En la Tabla III se muestran, en contraste, los programas de estudio de cursos auxiliares ofrecidos por Institución educativa, y que tienen que ver con las temáticas que aquí nos atañen, inicialmente conocimientos de informática, y posteriormente algún curso dirigido específicamente a ciberseguridad [14].

Tabla II: Número de Instituciones que ofrecen el programa Üniversidad de Mayores"por comunidad autónoma [14]

Comunidad Autónoma	Instituciones
Andalucía	9
Aragón	1
Asturias	1
Baleares	1
Canarias	2
Cantabria	1
Castilla - La Mancha	1
Castilla y León	10
Cataluña	11
Madrid	9
Comunidad Valenciana	6
Extremadura	1
Galicia	3
La Rioja	1
Murcia	2
Navarra	1
País Vasco	2
Programas de estudio específicos para mayores (a distancia)	9

De la información mostrada en las tablas anteriores, podemos notar que ya se han iniciado varios programas de estudio, como apoyo y aprendizaje para los mayores, sin embargo, existen comparativamente muy pocos cursos ofertados que tienen que ver con competencias informáticas, y constituyendo una minoría mucho más pequeña, aquellos específicamente dirigidos a campos que tienen que ver con ciberseguridad. Por tanto, en adelante se propone un marco general de intereses y temáticas necesarios para la adición de dicha temática tan interesante y vanguardista a las ofertas académicas.

VII. Recomendaciones particulares de programas de capacitación

Con base en las necesidades previamente identificadas, se plantean algunas directrices generales para el diseño de cursos

Tabla III: Instituciones educativas que ofrecen cursos para mayores orientados a Informática y/o Ciberseguridad [14].

Universidad	Informática / Tecnología	Ciberseguridad
U. de Córdoba	5	1
U. de Granada	4	0
U. de Huelva	1	0
U. de Málaga	1	0
U. de Sevilla	1	0
U. Pablo de Olavide	2	0
U. de León	1	0
U. de Segovia	1	0
U. de Valladolid	1	0
U. Politécnica de Cataluña	2	0
U. Complutense de Madrid	3	1
U. Rey Juan Carlos	8	0
Vniversitas Senioribvs	3	0
Comillas Universidad Pontificia	1	0
U. de Alicante	3	1
U. Jaume I	2	0
U. Politécnica de Valencia	1	0
U. de A Coruña	4	0

de capacitación que, al no requerir la emisión de certificados con validez oficial o titulación, pueden considerarse de carácter informal. No obstante, se recomienda que su impartición se realice dentro de una institución educativa, a fin de garantizar su adecuada orientación y aprovechamiento.

1. **Iniciación a la informática (menores de edad)**: Partiendo del punto en el que se cuenta con los conocimientos, no solo de informática básica, sino del software lúdico, educativo, y su utilización correcta, es deseable poder ofrecer cursos iniciales, para pequeños en edades de 6 a 9 años, que atacarán dos propósitos de gran relevancia: tanto la enseñanza de herramientas confiables y seguras para los menores y sus padres, como la sensibilización, aterrizada al nivel de comprensión de los menores sobre los pasos que siguen evolutivamente en su aprendizaje informático, y los riesgos que puede conllevar cruzar las fronteras de la corrección.
2. **Ciberseguridad para pequeños**: Equipar a los menores con conocimientos para conducirse por el ciberespacio con seguridad y confianza, es una meta realmente deseable con efectos de prevención y detección temprana ante los posibles riesgos y delitos cibernéticos en contra de los menores, que podría darse en seguida de Ïniciación a la informática"partiendo de edades de los 8 a los 12 años, tratando temáticas de relevancia como: (i) Detección de correos electrónicos y sitios maliciosos, (ii) Fortalecimiento de contraseñas y cuentas, (iii) Resguardo y protección de datos personales, (iv) Uso responsable de redes sociales, (v) ¿Qué es grooming? y ¿Cómo lo identifico?, (vi) Mejores prácticas durante la navegación en internet, (vii) Identificación de conductas de riesgo y amenazas potenciales, (viii) ¿Qué hacer en caso de sentirse inseguros o amenazados?. Dichas competencias tendrán un impacto muy positivo, derivado de la prevención y detección previa de situaciones que amenacen la integridad, privacidad o seguridad de alguno de los menores.
3. **Introducción a la informática (adultos mayores)**: Dirigir correctamente los esfuerzos de aprendizaje de los mayores, no solamente a las herramientas informáticas de oficina (procesador de textos, hojas de cálculo, presentación de diapositivas, etcétera), como si se tratase de una capacitación para algún puesto administrativo, sino realizar el desglose de un temario dirigido a las necesidades que se tienen según el estilo de vida de cada uno, como: Internet y redes sociales, Compra y venta por internet, Suscripciones y servicios on-line, Correo electrónico y mensajería instantánea, Páginas web y servicios gubernamentales, Servicios bancarios on-line, Datos personales, Riesgos y amenazas de internet, y ¿Qué hacer en caso de sentirse inseguros o amenazados? Sin dejar de lado, por supuesto, las buenas prácticas de seguridad en cada uno de los temas, haciendo hincapié en la protección de la información y cuentas que los mantenga seguros y con la guardia arriba ante las amenazas de la red.

VIII. Conclusiones

Es, mediante la capacitación, idealmente formalmente estructurada y dirigida correcta y profesionalmente, como se puede abatir la problemática de incertidumbre en la seguridad. Mayoritariamente sufren la comisión de delitos informáticos el grupo que se encuentra en etapa laboralmente activa (26 a 65 años). Lo anterior, excluye aquellos cometidos en contra de menores o de adultos mayores, que no hayan sido propiamente detectados y denunciados. La correcta detección de amenazas a la seguridad informática permitirá, no solo tipificar y realizar correctamente la denuncia correspondiente, sino a su vez prevendrá caer en la trampa que la amenaza plantea. Una actitud de prevención y la habilidad de detección de amenazas puede darse, únicamente mediante el conocimiento sobre ciberseguridad. La mejor manera de garantizar un aprendizaje correcto y profesional es mediante la instrucción formal y capacitada. Los organismos mejor capacitados (y que ya cuentan con algo de infraestructura) para realizar cursos formales son las instituciones educativas. Existen ya algunas propuestas como la Universidad para Mayores por parte de algunas universidades, tanto de España como del mundo. Sería altamente benéfico promocionar y publicitar los programas ya existentes de las Universidades de Mayores, para que pudieran ampliar la población en la que impactan. Resulta de gran relevancia proponer la inclusión de curos de ciberseguridad como los anteriormente descritos en el currículo de los que se ofrecen en las Universidades para Mayores. Una vez realizados las planeaciones de cursos para mayores, es una labor más fácil la realización de estos para los cursos anteriormente descritos para niños. Las capacitaciones para adolescentes y jóvenes pueden incluso plantease en esquemas híbridos y no-presenciales. Una vez implementados algunos de estos programas de capacitación sería de alta relevancia poder analizar su impacto en los participantes.

Agradecimientos

Este trabajo se ha realizado con los fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra "Ciberseguridad para la Innovación y la Protección Digital" INCIBE-UCM. Además, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, proyecto CIRMA-CM (TEC-2024/COM-404). El contenido de este artículo no refleja la opinión oficial de la Unión Europea. Los autores son

los únicos responsables de la información y las opiniones expresadas en él.

REFERENCIAS

[1] A. I. Cerezo Domínguez y R. García Cornejo: ”La ciberdelincuencia en España”, en *Revista Peruana de Ciencias Penales*, vol. 1, n. 34, pp. 91-106, 2020.

[2] M. N. Solari Merlo: ”Tendencias del Cibercrimen en España”, en *Instituto Andaluz Interuniversitario de Criminología*, 2025.

[3] L. Tarditi: Çomparaciones intergeneracionales en los usos de TIC”, en *Anales en Gerontología*, vol. 1, n. 14, pp. 1-26, 2022.

[4] N. Fraguela: ”Número de usuarios de internet en el mundo”, en *Marketing 4 Commerce*, 2025.

[5] United Nations, “World Population Prospects 2025 — Population pyramid,” *UN Population Division*, [Online]. Available: https://population.un.org/wpp/graphs?loc=900&type=Demographic%20Profiles&category=Population%20Pyramids&year=2025. Accessed: Nov. 18, 2025.

[6] P. F. Troya Aldaz et al.: “Criminología relacionada con delitos cibernéticos y la falta de punibilidad de conductas”, en *Ciencia Latina*, vol. 8, n. 6, pp. 241-258, 2024.

[7] C. A. Trejos Gil y Y. Peláez Vélez: Çiberdelitos en menores de edad en la red social Facebook: revisión sistemática de literatura”, en *Nuevo Derecho*, vol. 19, n. 32, pp. 1-18, 2023.

[8] G. Orozco López: Çiberseguridad y derechos de niñas, niños y adolescentes”, en *Compendio internacional de discusiones jurídicas contemponráneas II*, pp. 77-94, 2022.

[9] L. A. López Vera et al.: ”Ciberseguridad y los derechos de niñas, niños y adolescentes”, en *Compendio internacional de discusiones jurídicas contemponráneas II, Ciudad de México*, pp. 77-94, 2022.

[10] D. C. Toledo Verdugo y F. E. Ochoa Rodríguez: ”Derechos fundamentales y criminalidad cibernética en niños, niñas y adolescentes: análisis para la no indefensión de la víctima”, en *Revista Científica Ciencas Económicas y Empresariales*, vol. 6, n. 30, pp. 336-363, 2021.

[11] R. Guerrero: ”Dispositivos tecnológicos y sobreprotección de los padres: dos enemigos para la salud mental de los adolescentes”, en *El País, Madrid,*, 2025.

[12] J. Cabero Almenara y C. Llorente Cejudo: ”La adopción de las tecnologías por las personas mayores: aportaciones desde el modelo TAM, en *Publicaciones. Facultad de Educación y Humanidades del Campus de Melilla*, n. 1 pp. 141-157, 2020.

[13] Peral-Peral, B., Villarejo-Ramos, Á.F. y Arenas-Gaitán, J. ”Self-efficacy and anxiety as determinants of older adults’ use of Internet Banking Services.[en] *Univ Access Inf Soc* vol. 19, pp. 825–840, 2020. Disponible en: https://doi.org/10.1007/s10209-019-00691-w

[14] Consejo Superior de Investigaciones Científicas: ”Programas Universitarios para Mayores”, en *Consejo Superior de Investigaciones Cientſficas, Madrid*, 2025, [Online]. Disponible en: http://envejecimiento.csic.es/documentacion/varia/programas-universidad-mayores.html. Consultado: Nov. 18, 2025.

Interpretable Neuro-Symbolic Graph Neural Network for Anomaly Detection

Fernando Ares-Robledo*, Helena Rifà-Pous*†, Robert Clarisó*
*UOC-TECH Research Center, Estudios de Informática, Multimedia y Telecomunicación
Universitat Oberta de Catalunya (UOC)
{faresro,hrifa,rclariso}@uoc.edu
‡ CYBERCAT - Center for Cybersecurity Research of Catalonia

***Abstract*—Graph-based intrusion detection often relies on black box Graph Neural Network (GNN) with strong accuracy, limited interpretability, and fragile transfer across networks. Existing explainers mostly give post-hoc attributions rather than executable, verifiable rules. We present a pipeline for anomaly detection that couples a single round GNN with Symbolic Regression (SR). The GNN gives predictions at edge and graph-level, while SR distills its computations into compact closed-form equations for edge messages, node updates, and graph decisions, preserving the model's computation path. The resulting formulas make decisions auditable and reusable, and they support transfer by calibrating constants instead of retraining full models. We document the relationship between parsimony and fit, the presence of families of near equivalent expressions, and the effect of domain shift, showing near GNN performance at graph-level and moderate drops at edge-level with clear gains in interpretability.**

***Index Terms*—Graph Neural Network (GNN), Explainable Artificial Intelligence (XAI), Symbolic Regression (SR), Anomaly Detection**

I. Introduction

Anomaly detection remains central to cybersecurity because intrusions are rare, stealthy, and can trigger disproportionate damage. Systems must therefore surface deviations that standard tooling often misses [1]. Enterprise activity is inherently relational and unfolds over time. Hosts, IP addresses, users, and processes interact through connections that form evolving structures, so a graph representation is a natural fit to capture such dependencies [2]. Modeling events as graphs exposes multi-hop relationships and collective patterns at the subgraph-level, which helps reveal distributed behaviors such as coordinated scanning or low-rate attacks that would be invisible in isolated records [3].

Real environments are non-stationary. Traffic and attacker behavior shift, concept drift erodes the accuracy of models trained on yesterday's distributions and pushes toward online adaptation and continual updates [4]. At the same time, benign traffic vastly outnumbers malicious activity and ground truth is scarce or noisy, which complicates the learning and raises the risk of bias toward the majority class [5]. These pressures make the relational view even more valuable, because structural context can counteract sparsity in labels while highlighting structure that simple tabular pipelines discard.

Graph Neural Network (GNN) aggregate information from neighbors via a message-passing framework to iteratively update and encode node embeddings. This process captures structural context such as multi-hop relationships and collective patterns that tabular models, by construction, ignore. This mechanism captures higher order dependencies and temporal cues when used on dynamic graphs [6]. Despite their performance, such models are often perceived as black boxes that are hard to audit, debug, or justify to Security Operations Center (SOC) analysts and compliance stakeholders, which hinders trust and actionability in operational settings [7]. In security operations, explanations are not ornamental: analysts need them to triage alerts, cut false positives, and decide concrete responses, including forensics and regulatory reporting [8].

A large share of Explainable Artificial Intelligence (XAI) for graphs produces instance-specific rationales. These explanations can be unstable under small perturbations and expensive to compute after the fact, limiting their practicality for real-time workloads [9], [10]. The shortage of global explanations, rules, or equations that hold across broad regions of the data space also curbs knowledge transfer and leaves systems exposed to evasion when they rely on brittle, non-robust features [11]. In practice, many graph-based intrusion detection systems (IDS) pipelines still optimize for accuracy first, leaving interpretability as a secondary concern and widening the gap between strong metrics and actionable decisions [12].

Only a few frameworks attempt to unify graph learning with symbolic reasoning to provide compact and verifiable explanations [13]. A previous work on symbolic distillation for graph-based intrusion detection also falls into this category [14], and this remains an open research area. Evaluation also tends to overlook operational utility, such as analyst effort, time to containment, and cost per investigated alert. However, these are the quantities that matter in a SOC [15].

Symbolic regression searches over a space of mathematical expressions to identify equations that fit observed data. It can distill closed-form relationships that link features to anomaly scores, producing concise, human-readable models that support scrutiny and reuse [16]. Integrating symbolic components, rules, equations or expert knowledge with GNN combines predictive strength with traceability and verifiability [17]. Equations that capture invariants ease auditing, aid in transfer to new domains, and can strengthen robustness under Out-of-Distribution (OOD) shifts [18]. This neuro-symbolic direction also creates a direct path to defense policy, since learned formulas or rules can be translated into Security Information and Event Management (SIEM) or Network Intrusion Detection System (NIDS) specifications that drive automated responses [7].

This paper presents a novel model for graph-based anomaly detection that combines a single round of graph neural com-

putation with symbolic regression to obtain compact and verifiable equations for edge, node, and graph decisions. Unlike standard black box GNNs, this neuro-symbolic approach achieves predictive performance at the graph-level close to that of the GNN while providing executable and auditable rules that improve model interpretability. Explicit formulas enable practical model transfer through lightweight calibration of constants across datasets, avoiding full retraining and mitigating domain shift while preserving the functional structure.

The remainder of this paper is organized as follows. Section II introduces the basic concepts. Section III describes the model pipeline, the single round GNN, the training procedure, and the symbolic surrogates derived from the model traces. Section IV details the experimental setup and reports graph and edge-level findings, including the learned equations and their behavior cross-domain. Section V summarizes the results and describes directions for future work.

II. Background and definitions

This section outlines the core notions and terminology used throughout.

a) Graphs: A graph is an ordered pair $G = (V, E)$, where V is a finite set of nodes and E is a set of edges. In a directed graph, edges are ordered pairs (u, v), so $(u, v) \neq (v, u)$ in general. Features may be attached to nodes, edges, or both, $x_v \in \mathbb{R}^{d_v}$ for $v \in V$ and $x_{u\to v} \in \mathbb{R}^{d_e}$ for $(u, v) \in E$. We say that a method operates at the graph-level when it outputs a single label or score per G, and at the edge-level when it outputs one per $(u, v) \in E$.

b) GNN: A GNN is a neural architecture that operates on graphs. It propagates information along edges E (message passing) so that each node V updates its state from its neighbors, then aggregates node states into a fixed-size representation (readout) for downstream prediction. The computation is permutation-invariant with respect to node ordering and naturally supports supervision at node, edge, or graph-level.

c) Symbolic Regression (SR): SR is a type of genetic algorithm that aims to discover human-readable formulas that explain a target from inputs. It searches a space of expression trees built from a small set of primitives (for example, addition, multiplication, division) and balances predictive fit with simplicity so that the final expression is compact and interpretable.

d) Denial-of-Service (DoS): A DoS attack attempts to exhaust a service or link by overwhelming it with traffic or requests. In NetFlow-style records this often manifests as bursts with large packet counts, unbalanced directions, or atypical durations.

e) Metrics: Let True positive (TP), True negative (TN), False positive (FP), and False negative (FN) be the counts from the confusion matrix for a binary decision task. We define *accuracy* in Eq. 1 and *precision* in Eq. 2:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{2}$$

III. Architecture

This section lays out the end-to-end pipeline we employ, tying a single round GNN to a set of symbolic surrogates learned from model traces. The diagram in Fig. 1 offers a compact overview, while Algorithm 1 summarizes training, selection, and the SR stages.

The starting point is building a directed graph $G = (V, E)$, where each node v_i represents an IP, with edge features $x_{u\to v} \in \mathbb{R}^{d_{\text{in}}}$. The edge encoder ϕ_e converts raw features, such as the total number of bytes and packets or the flow duration, into messages, as formalized in Eq. 3:

$$e_{u\to v} = \phi_e(x_{u\to v}) \in \mathbb{R}^{d_{\text{msg}}}, \tag{3}$$

applied independently to each edge during the forward sweep depicted in Fig. 1. Messages are then aggregated per node by a directed reduction $\bigoplus \in \{\text{sum}, \text{mean}\}$, that is, according to Eq. 4:

$$m_{\text{in}}(v) = \bigoplus_{(u\to v)\in E} e_{u\to v}, \qquad m_{\text{out}}(v) = \bigoplus_{(v\to w)\in E} e_{v\to w}, \tag{4}$$

and concatenated as $m(v) = [m_{\text{in}}(v) \parallel m_{\text{out}}(v)]$. The node encoder ψ_v maps this summary to a latent state as in Eq. 5:

$$h_v = \psi_v\big(m(v)\big) \in \mathbb{R}^{d_{\text{node}}}, \tag{5}$$

where h_v^{lat} denotes the output of the last linear layer of ψ_v prior to the nonlinearity.

Graph-level prediction relies on pooling and a shallow head. Node states are aggregated as specified in Eq. 6:

$$h_G = \bigoplus_{v\in V} h_v, \tag{6}$$

and a linear head Θ produces a graph logit, the unnormalised score, as shown in Eq. 7:

$$z_G = \Theta(h_G) \in \mathbb{R}. \tag{7}$$

Edge-level prediction uses a local context composed of the message and the endpoint states, as formalized in Eq. 8:

$$z_{u\to v} = \Xi(\,[e_{u\to v} \parallel h_u \parallel h_v]\,) \in \mathbb{R}, \tag{8}$$

with a shared edge head Ξ.

Training optimizes a multitask objective with binary cross entropy and sparse activations (Algorithm 1, lines 2–4), as shown in Eq. 9:

$$\mathcal{L} = \text{BCE}(z_G, y_G) + \alpha\,\text{BCE}(z_E, y_E) + \lambda_e\,\overline{\|e\|_1} + \lambda_h\,\overline{\|h\|_1}, \tag{9}$$

where $\overline{\|e\|_1}$ and $\overline{\|h\|_1}$ denote average ℓ_1 penalties over messages and node states, respectively. The checkpoint that maximizes validation F1 at the graph-level is retained.

Once the GNN has been trained and frozen, three symbolic surrogates are distilled from the saved pairs, as summarized in Algorithm 1, lines 5-8 and shown in Fig. 1. A first surrogate S_e explains a scalar summary of the edge message using raw features, via the mapping in Eq. 10, and is fitted on the edge-level pairs built in Algorithm 1 (lines 9–10):

$$S_e :\; x_{u\to v} \mapsto \hat{s}_e \approx s_e\big(e_{u\to v}\big), \tag{10}$$

where s_e is chosen from $\{e_{u\to v}^{[j^\star]},\ \|e_{u\to v}\|_2,\ \text{mean}(e_{u\to v})\}$. The max-variance rule selects either a specific channel $e_{u\to v}^{[j^\star]}$

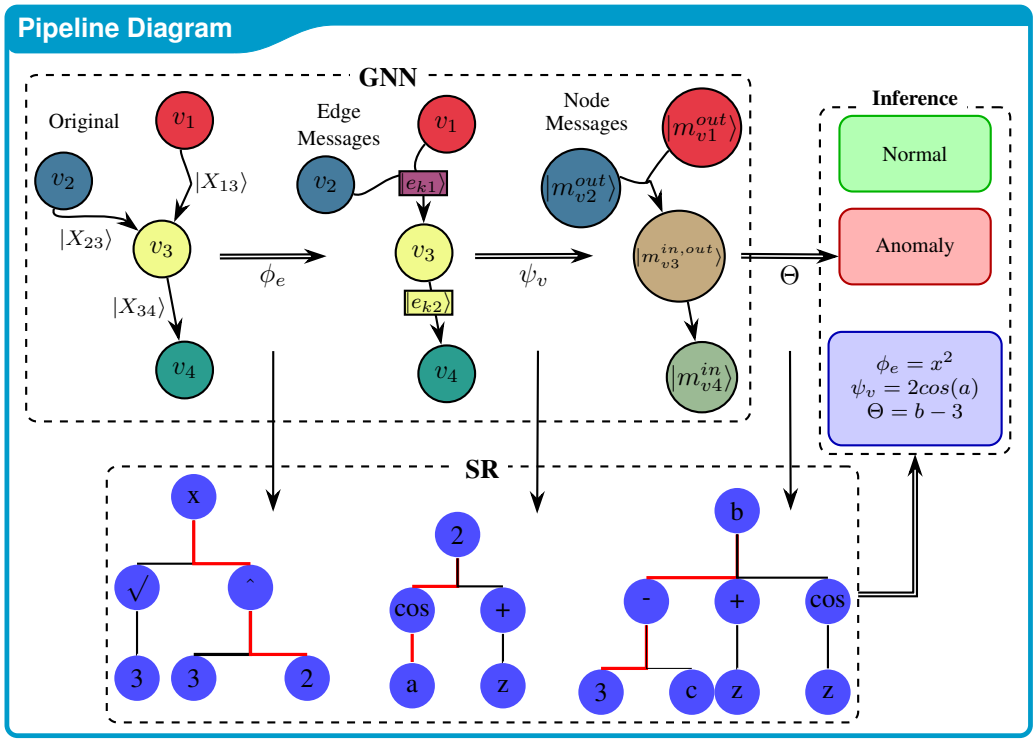


Fig. 1. End-to-end pipeline: a single round GNN (top) yields edge and node messages together with a graph logit. Tapped signals (arrows) feed SR components (bottom), which learn symbolic surrogates used at inference time.

Algorithm 1 Symbolic surrogate pipeline

procedure TRAINANDDISTILL(D)
  split D into $D_{\text{train}}, D_{\text{val}}, D_{\text{test}}$
  train GNN on D_{train} with loss
  select checkpoint $\theta^\star$ by graph-level F1 on D_{val}
  freeze $\theta^\star$ and run forward passes on all splits
  dump $(x_{u\to v}, e_{u\to v})$ for all edges
  dump $(m_{\text{in}}(v), m_{\text{out}}(v), h_v, h_v^{\text{lat}})$ for all nodes
  dump (h_G, z_G) for all graphs
  build (X_e, y_e) from edge pairs $(x_{u\to v}, e_{u\to v})$
  fit S_e on (X_e, y_e) by SR
  build (X_v, y_v) from node pairs $(m_{\text{in}}, m_{\text{out}}, h_v^{\text{lat}})$
  fit S_v on (X_v, y_v) by SR
  pool S_e and S_v per graph to obtain (X_G, y_G)
  fit S_G on (X_G, y_G) by SR
  return $\theta^\star, S_e, S_v, S_G$
end procedure

or a norm based summary. A second surrogate S_v emulates the node update on that channel, using the two directed aggregates as input (Algorithm 1, lines 11–12), as formalized in Eq. 11:

$$S_v : \left[m_{\text{in}}^{[j^\star]}(v),\, m_{\text{out}}^{[j^\star]}(v)\right] \mapsto \hat{s}_v \approx s_v\big(h_v^{\text{lat}}\big), \tag{11}$$

where s_v is a scalar summary of the latent node state, such as a fixed coordinate or an ℓ_2 norm.

Graph-level behavior is finally approximated by composing pooled summaries (Algorithm 1, final lines). For each graph G, edge and node contributions are aggregated as in Eq. 12

$$e_{\text{pool}}(G) = \sum_{(u\to v)\in E} S_e(x_{u\to v}), \qquad sv_{\text{pool}}(G) = \sum_{v\in V} S_v(\cdot), \tag{12}$$

and a two-input graph surrogate is learned (Eq. 13):

$$S_G : \left[sv_{\text{pool}}(G),\, e_{\text{pool}}(G)\right] \mapsto \hat{z}_G \approx z_G, \tag{13}$$

which matches the lower-right composition in Fig. 1. As an ablation, we also approximate the edge head from the message alone by $S_E : e_{u\to v} \mapsto \hat{z}_{u\to v} \approx z_{u\to v}$, reusing the first d_{msg} coordinates of the edge context.

IV. RESULTS

A. Experimental Setup

We used two NetFlow datasets [19]: NF-UNSW-NB15-v2 and NF-ToN-IoT-V2. They cover a conventional enterprise network and an IoT oriented environment, so they expose DoS traffic in two different deployment contexts while sharing the same flow semantics. From each, we select the Normal and DoS records and construct graphs with the same preprocessing and message-passing configuration described in the previous section.

After this filtering step, UNSW-NB15 contains $2,295,222$ benign instances (99.3%) and $16,353$ DoS instances (0.7%), while ToN-IoT contains $6,099,469$ benign instances (89.5%) and $712,609$ DoS instances (10.5%). We draw an 80/20 split between training and test data, stratified by class.

For the feature space, we keep only numeric traffic attributes, remove identifier fields, and drop all records that contain missing values. Categorical attributes are converted to numeric form by one-hot encoding, and every feature is normalized to zero mean and unit variance using statistics computed on the training split only; the same transformation is then applied to the test split.

The hardware setup consists of an AMD Ryzen 7 7435HS CPU, an NVIDIA GeForce RTX 4050 6GB GPU, and 24GB

of DDR5 RAM. The software versions being used are: Python 3.12.7, gplearn 0.4.2, matplotlib 3.9.3, numpy 1.26.4, pandas 2.2.3, polars 1.31.0, torch 2.5.1, and scikit-learn 1.5.2.

B. Results

We evaluate classification at two granularities, graph-level (deciding whether an entire graph has Normal or DoS traffic) and edge-level (deciding whether each edge instance belongs to the DoS class). In NF-UNSW-NB15-v2, the trained GNN reaches graph-level accuracy 1.000 and precision 1.000, and edge-level accuracy 0.993 and precision 0.982 on the held-out split.

The symbolic equations learned on NF-UNSW-NB15-v2 are:

$$S_e(x) = \log\left(\frac{x_7}{x_2 - x_4}\right), \tag{14}$$

$$S_v([m_{\text{in}}, m_{\text{out}}]) = \log\big(-1.260 - m_{in,2} - m_{out,2}\big), \tag{15}$$

$$S'_G(sv_pool, e_pool) = 2.794\,\frac{sv_pool - 1.666}{\sqrt{sv_pool}}. \tag{16}$$

Here x_2=IN_PKTS, x_4=OUT_PKTS, x_7=DURATION_OUT. These features describe, respectively, the number of packets received by a host, the number of packets sent, and the duration of the outbound connection within a NetFlow record.

The form of S_e in Eq. 14 increases with longer outgoing duration and decreases when inbound packets dominate outbound packets; DoS traffic tends to push $x_2 \gg x_4$, shrinking S_e.

The node surrogate in Eq. 15 uses the same message channel selected by S_e: $m_{in,2}$ is the incoming aggregate for index 2 and $m_{out,2}$ is the outgoing aggregate for the same index with offset d_{msg}. The learned logarithmic form acts as a smooth, saturating nonlinearity that compresses large variations in node activity. Its argument must remain positive, which implies $m_{in,2} + m_{out,2} < -1.260$. This constraint delineates the operational region of the model: nodes where the combined incoming and outgoing contributions on that channel are strongly negative, typically those exhibiting unbalanced or dominating outbound flows, fall within the active domain of S_v. In practice, such nodes correspond to DoS-like behavior, where traffic patterns are asymmetric and the outgoing rate overwhelms the inbound direction.

Finally, S'_G in Eq. 16 depends only on the pooled node surrogate sv_pool. In a Normal–DoS setting, the discriminative signal is largely volumetric and already captured by the node aggregates; the summed edge contribution e_pool is approximately colinear with these totals (and with the size of the graph), it adds little marginal information. The symbolic regressor, penalizing complexity, therefore drops e_pool without loss of predictive power.

The emergence of logarithmic terms in the symbolic surrogates is not coincidental. These functions mirror the nonlinear compression performed by the activation layers of the underlying GNN, dampening large variations while preserving relative differences. Fig. 2 compares the learned graph-level surrogate S'_G with a standard logarithmic curve, showing that both exhibit the same concave, saturating profile, they grow rapidly for small inputs, and flatten for larger values.

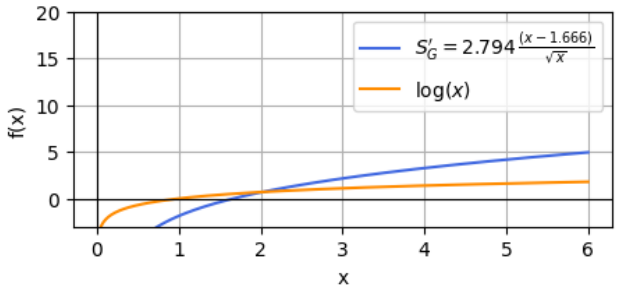


Fig. 2. Comparison between the learned graph-level surrogate S'_G and a logarithmic function.

Applying these equations back to NF-UNSW-NB15-v2 yields graph-level accuracy 0.9992 and precision 1.0000, and edge-level accuracy 0.8929 and precision 0.9833. Compared to the original GNN, which achieved almost perfect graph metrics and very high edge accuracy, this shows a moderate drop at edge-level while graph-level performance is almost unchanged. In return, the model exposes explicit equations that analysts can inspect, audit, and reuse, rather than relying on opaque latent representations.

To probe how the learned equations behave under cross dataset transfer, we reuse them on NF-ToN-IoT-V2 without retraining or recalibration. Using the same equations on NF-ToN-IoT-V2 shows a domain shift, graph-level accuracy/precision 0.8057, and edge-level accuracy 0.6710 with precision 0.6796. The gap points to different feature ranges and packet dynamics across datasets, though the closed-forms still carry informative signal at both levels.

C. Discussion

The above results expose four practical issues. First, keeping formulas interpretable requires a hard cap on expression size. We measure length as the count of primitive operations plus the count of variables. Under this metric, S_e in Eq. 14 has length 6 (three operations, three variables). Allowing a length of up to 12 improves the metrics by about 2% on NF-UNSW-NB15-v2 and about 7% when reusing the same equations in NF-ToN-IoT-V2, but the added structure makes the resulting expressions harder to read and justify.

Second, even with short programs, different expressions often achieve the same scores. The setting is not uniquely identifiable, many algebraically related forms can approximate the same target and collapse to one another under simplification. In practice, we observe small families of operationally equivalent formulas that are interchangeable from a predictive standpoint.

Third, the usefulness of a given surrogate depends on the type of attack. With only Normal and DoS graphs, S_e captures the most salient signal, while S_v offers limited additional insight for detection. For attacks that hinge on node-centric manipulation (for example, man-in-the-middle), the two-input node surrogate may become more informative. The goal is not to replace domain expertise, but to offer a compact starting point for interpreting the observed behavior.

Fourth, cross-domain transfer reveals a clear domain shift, equations learned in NF-UNSW-NB15-v2 lose accuracy in

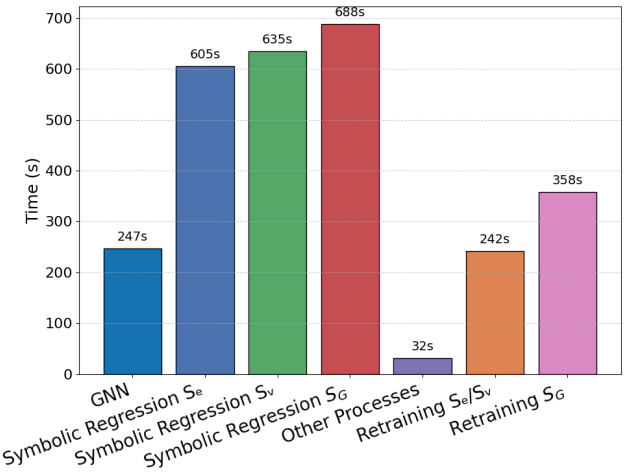


Fig. 3. Execution time of the main stages.

NF-ToN-IoT-V2. Rather than retraining, a lightweight calibration helps freeze the structure and refit a few constants on a small slice of the target data. Starting from S_e in Eq. 14, this yields $\tilde{S}_e(x) = \log\left(\frac{x_7 - 0.4}{x_2 - x_4}\right)$, which, after recomputing the pooled signals and decisions, raises the edge-level accuracy to 0.8013 with precision 0.7971 in NF-ToN-IoT-V2. The improvement is modest, but it shows that the functional form carries over while constants absorb scale and offset differences between domains. Extending the same calibration to S_v and S'_G and pairing it with simple feature normalization is a natural next step to stabilize transfer without sacrificing interpretability.

From an operational standpoint, the explicit symbolic form of S_e, S_v, and S'_G also supports the analysis of false positives. Each alert can be traced back to concrete terms in the equations, making it possible to identify which traffic attributes or aggregation patterns drive the anomaly score. For example, alerts dominated by S_e in Eq. 14 can be examined in terms of packet asymmetry and flow duration, allowing analysts to verify whether such conditions are consistent with the expected role of a host or service. This enables the distinction between benign deviations and genuinely suspicious behavior, and allows lightweight adjustments of constants or thresholds to reflect local traffic policies, yielding explicit and auditable rules without retraining the full model.

D. Performance

The initial training stage requires approximately 30–40 minutes to complete, reflecting the combined cost of GNN optimization and the first round of symbolic regression. During this phase, the symbolic regressors S_e, S_v, and S'_G dominate the runtime because the evolutionary search must explore a large space of possible expressions before converging to compact, interpretable formulas. In contrast, retraining symbolic surrogates, illustrated in Fig. 3, is considerably faster since it starts from an existing equation or family of related forms, adjusting only a few constants instead of discovering an entirely new structure. The baseline GNN remains comparatively efficient, as it is optimized within a fixed parameterized architecture. The column labeled Other Processes represents auxiliary operations such as data loading, variable elimination, model instantiation, and result storage, which together account for only a minor portion of the total execution time.

V. Conclusions

We presented a pipeline that couples a single round GNN with a set of symbolic surrogates learned from the model traces. The approach keeps the predictive path intact while exposing compact equations for edge messages, node updates, and graph decisions, a view that is executable and readable. The results on NetFlow data with Normal and DoS classes show that short expressions can recover most of the behavior of the trained model, while also revealing the conditions under which the log domain or pooling steps drive the decision boundary.

Compared with current work on neuro-symbolic graph learning and intrusion detection, which often focuses on feature attributions, local surrogates, or rule sets at graph-level, this study delivers explicit equations for both edges and graphs. Graph-level performance stays almost identical to the original GNN, while edge-level metrics drop but remain strong. This supports the view that symbolic regression can approach black box performance in security settings, and extends it to flow graphs where edge decisions and graph decisions are modeled in a single pipeline.

The study also highlights compromises and limits. Expression length must be controlled to preserve interpretability, yet stricter caps can leave accuracy on the table. Multiple near equivalent formulas often tie, reflecting the fact that the mapping is not uniquely identifiable. Transfer across datasets exposes domain shift; still, the learned structure offers a starting point to calibrate constants or adjust terms rather than retraining from scratch.

Future work includes broadening the attack taxonomy, exploring multiround message passing, adding structural priors or constraints during symbolic search, and developing lightweight calibration for new domains.

Acknowledgments

This work was supported by the Spanish Ministry of Science and Innovation through the projects PID2021-125962OB-C31 “SECURING/CYBER”, PID2023-147592OB-I00 “SE4GenAI” and PID2024-156914OB-C41 “SAFE/CYBER”. Additional funding was provided by the ARTEMISA International Chair of Cybersecurity (C057/23) and the DANGER Strategic Project of Cybersecurity (C062/23), both funded by the Spanish National Institute of Cybersecurity through the European Union NextGenerationEU and the Recovery, Transformation, and Resilience Plan.

References

[1] L.-H. Lim, L.-Y. Ong, and M.-C. Leow, “Federated learning for anomaly detection: A systematic review on scalability, adaptability, and benchmarking framework,” *Future Internet*, vol. 17, no. 8, 2025. [Online]. Available: https://www.mdpi.com/1999-5903/17/8/375

[2] A. S. Ahanger, S. M. Khan, F. Masoodi, and A. O. Salau, “Advanced intrusion detection in internet of things using graph attention networks,” *Scientific Reports*, vol. 15, no. 1, p. 9831, 2025. [Online]. Available: https://doi.org/10.1038/s41598-025-94624-8

[3] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: a survey," *Data Min. Knowl. Discov.*, vol. 29, no. 3, p. 626–688, May 2015. [Online]. Available: https://doi.org/10.1007/s10618-014-0365-y

[4] R. Gupta, S. Liu, R. Zhang, X. Hu, X. Wang, H. Benkraouda, P. Kommaraju, P. Cao, N. Feamster, and K. Nahrstedt, "Generative active adaptation for drifting and imbalanced network intrusion detection," 2025. [Online]. Available: https://arxiv.org/abs/2503.03022

[5] G. Agrawal, A. Kaur, and S. Myneni, "A review of generative models in generating synthetic attack data for cybersecurity," *Electronics*, vol. 13, no. 2, 2024. [Online]. Available: https://www.mdpi.com/2079-9292/13/2/322

[6] A. Zoubir and B. Missaoui, "Integrating graph neural networks with scattering transform for anomaly detection," 2024. [Online]. Available: https://arxiv.org/abs/2404.10800

[7] F. Uccello, M. Pawlicki, S. D'Antonio, and R. Kozik, "A new cybersecurity approach enhanced by xai-derived rules to improve network intrusion detection and siem," *Computers, Materials & Continua*, vol. 83, no. 2, pp. 1607–1621, 2025. [Online]. Available: http://www.techscience.com/cmc/v83n2/60589

[8] R. Kalakoti, R. Vaarandi, H. Bahşi, and S. Nõmm, "Evaluating explainable ai for deep learning-based network intrusion detection system alert classification," in *Proceedings of the 11th International Conference on Information Systems Security and Privacy*. SCITEPRESS - Science and Technology Publications, 2025, p. 47–58. [Online]. Available: http://dx.doi.org/10.5220/0013180700003899

[9] J. Ribeiro, L. Cardoso, V. Santos, E. Carvalho, N. Carneiro, and R. Alves, "How reliable and stable are explanations of xai methods?" 2024.

[10] S. Akkas and A. Azad, "Gnnshap: Scalable and accurate gnn explanation using shapley values," in *Proceedings of the ACM Web Conference 2024*, ser. WWW '24. ACM, May 2024, p. 827–838. [Online]. Available: http://dx.doi.org/10.1145/3589334.3645599

[11] H. Zhang, D. Han, Y. Liu, Z. Wang, J. Sun, S. Zhuang, J. Liu, and J. Dong, "Explainable and transferable adversarial attack for ml-based network intrusion detectors," 2024. [Online]. Available: https://arxiv.org/abs/2401.10691

[12] W.-C. Chung, S.-T. Huang, and H.-T. Pai, "Multi-granular discretization for interpretable generalization in precise cyberattack identification," 2025. [Online]. Available: https://arxiv.org/abs/2507.14223

[13] V. B. Kumar, B. Ganesan, M. Ameen, D. Sharma, and A. Agarwal, "Automated evaluation of gnn explanations with neuro symbolic reasoning," in *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*, ser. Proceedings of Machine Learning Research, D. Kiela, M. Ciccone, and B. Caputo, Eds., vol. 176. PMLR, 06–14 Dec 2022, pp. 314–318. [Online]. Available: https://proceedings.mlr.press/v176/kumar22a.html

[14] F. Ares-Robledo, H. Rifà-Pous, and R. Clarisó, "Hyperparameter optimization in neuro-symbolic unsupervised graph learning," in *Availability, Reliability and Security*, B. Coppens, B. Volckaert, V. Naessens, and B. D. Sutter, Eds. Cham: Springer Nature Switzerland, 2025, pp. 275–292.

[15] N. Rastogi, S. Pant, D. Dhanuka, A. Saxena, and P. Mairal, "Too much to trust? measuring the security and cognitive impacts of explainability in ai-driven socs," 2025. [Online]. Available: https://arxiv.org/abs/2503.02065

[16] A. B. Javadi and P. Pong, "A review on symbolic regression in power systems: Methods, applications, and future directions," *Renewable and Sustainable Energy Reviews*, vol. 224, p. 116075, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032125007488

[17] L. C. Lamb, A. Garcez, M. Gori, M. Prates, P. Avelar, and M. Vardi, "Graph neural networks meet neural-symbolic computing: A survey and perspective," 2021. [Online]. Available: https://arxiv.org/abs/2003.00330

[18] A. Almadhor, S. Alsubai, N. Kryvinska, A. A. Hejaili, B. Bouallegue, M. Ayari, and S. Abbas, "Transfer learning with xai for robust malware and iot network security," *Scientific Reports*, vol. 15, no. 1, p. 26971, 2025. [Online]. Available: https://doi.org/10.1038/s41598-025-12404-w

[19] M. Sarhan, S. Layeghy, and M. Portmann, "Towards a standard feature set for network intrusion detection system datasets," *Mobile networks and applications*, pp. 1–14, 2022. [Online]. Available: https://doi.org/10.1007/s11036-021-01843-0

A Complete Description of the Hermitian Dual of Dihedral Codes and Derived Quantum Codes

Miguel Sales-Cabrera
Dpt. de Matemàtiques, Universitat d'Alacant
Sant Vicent del Raspeig, Ap. Correus 99, 03080 Alacant (Spain)
miguel.sales@ua.es

Xaro Soler-Escrivà
Dpt. de Matemàtiques, Universitat d'Alacant
Sant Vicent del Raspeig, Ap. Correus 99, 03080 Alacant (Spain)
xaro.soler@ua.es

Víctor Sotomayor
Departamento de Álgebra, Facultad de Ciencias, Universidad de Granada
Av. Fuente Nueva s/n, 18071 Granada (Spain)
vsotomayor@ugr.es

***Resumen*—Let D_n be the dihedral group of order $2n$ and q a prime power. Given the finite field $\mathbb{F}_{q^2}$ of q^2 elements, in this note we present a suitable Wedderburn-Artin's decomposition of the group algebra $\mathbb{F}_{q^2}[D_n]$, whenever q and n are coprime numbers. Then we obtain a full description of the hermitian dual of a given dihedral code, based on its expression as a direct sum of ideals within the Wedderburn-Artin's decomposition of $\mathbb{F}_{q^2}[D_n]$. Finally, we give a method to build quantum error-correcting codes, using the CSS construction for hermitian self-orthogonal dihedral codes.**

***Index Terms*—Linear codes, Group algebras, CSS quantum codes**

I. Introduction

In this note $\mathbb{F}_q$ will be the finite field of q elements, for some prime power q and n a positive integer such that $\mathrm{char}(\mathbb{F}_q) \nmid n$. Given a group G of order n, the $\mathbb{F}_q$-vector space $\mathbb{F}_q[G] = \left\{ \sum_{g\in G} a_g g \mid a_g \in \mathbb{F}_q \right\}$, with basis the elements of G, and endowed with multiplication $\left(\sum_{g\in G} a_g g\right) \cdot \left(\sum_{g\in G} b_g g\right) := \sum_{g\in G} \left(\sum_{h\in G} a_h b_{h^{-1}g}\right) g$, is an algebra, called the *group algebra* of G over $\mathbb{F}_q$. A linear code $\mathcal{C} \subseteq \mathbb{F}_q^n$ is called a *(left) group code* (or simply a *G-code*) if there exists an isomorphism $\phi : \mathbb{F}_q^n \longrightarrow \mathbb{F}_q[G]$ such that $\phi(\mathcal{C})$ is closed under (left) multiplication of every element of G. In other words, a G-code is any (left) ideal of $\mathbb{F}_q[G]$. Hereafter, when we talk about ideals of the algebra $\mathbb{F}_q[G]$, we always assume that they are left ideals.

The starting point of this algebraic approach to study linear codes may be placed in the papers [1] and [21], where S. Berman and F. MacWilliams independently introduced the notion of a group code, respectively. In fact, it was discovered in [1] that Reed-Muller codes over a finite field of characteristic 2 may be seen as ideals in the group algebra of an elementary abelian 2-group, and this was later extended for Generalised Reed-Muller codes (*i.e.* for odd characteristic) in [10]. Several families of linear codes have been proved to be G-codes for certain finite groups G, and we refer the interested reader to the survey [17].

The main interest in the study of group codes is that their algebraic structure can be deeply analysed using powerful tools from representation theory, some of which will be described in this note. An instance of this feature is given in [20], which takes advantage of the underlying algebra structure to propose a new decoding algorithm for Reed-Muller codes. This illustrates once again that the more mathematical structure an object possesses, the better are the descriptions we can obtain.

Abelian group codes have been extensively studied. However, the research on non-abelian group codes is gaining an increasing interest in recent years (*cf.* [13], [14], [26], to name a few). One of the main motivations for this is that quantum computers seem not able to break, so far, the security of public-key cryptosystems based on non-abelian groups codes (see [25]). We emphasise that a code can be seen as a group code for two different groups, and indeed one of them may be abelian and the other may not. In any case, as it is mentioned in the expository paper [15], non-abelian group codes that cannot be realised as G-codes for some abelian group G have length and dimension at least 24 and 4, respectively.

Brochero-Martínez described in [6] the Wedderburn-Artin's decomposition of the group algebra $\mathbb{F}_q[D_n]$ of any dihedral group of order $2n$ in the semisimple case, *i.e.* whenever the characteristic of the field $\mathbb{F}_q$ is coprime with $2n$. Later on, the decomposition of $\mathbb{F}_q[D_n]$ was utilised by Vedenev and Deundyak to obtain in [26] a detailed algebraic description of all D_n-codes and their euclidean dual codes. Cao *et al.* also determined algebraic expressions of D_n-codes, their euclidean dual codes and their hermitian dual codes but with a different approach: through the concatenated structure of dihedral codes (see [9], [8]).

This note is part of a paper in progress whose aim is to contribute to the aforementioned research line. More concretely, we refine the Wedderburn-Artin's decomposition of $\mathbb{F}_{q^2}[D_n]$

to provide a full description of the hermitian dual code of any D_n-code over $\mathbb{F}_{q^2}$ whenever the characteristic of the field is coprime with n. Our alternative approach via the group algebra structure reduces considerably the laborious computations given in [8]. After that and based on the hermitian CSS construction, we provide numerical examples of optimal quantum error-correcting codes that can be constructed from hermitian self-orthogonal dihedral codes.

II. Preliminars

All groups considered in this paper are supposed to be finite and $\mathcal{M}_n(R)$ will be the ring of $(n \times n)$-matrices over a ring R. Moreover, $\langle X \rangle_R$ denotes the ideal of R generated by a non-empty subset X of R (we will simply write $\langle X \rangle$ when the ambient ring R is clear enough). The rest of the notation not explained here is standard in the context of group theory or coding theory.

II-A. On Group Codes

The group algebra $\mathbb{F}_q[G]$ is said to be *semisimple* if it can be realised as a direct sum of simple $\mathbb{F}_q[G]$-modules, and this occurs, by Maschke's theorem, if and only if the characteristic of the field does not divide the order of G (cf. [11], [5]). In this case, every ideal of $\mathbb{F}_q[G]$ is always a direct summand and thus it must be principal. Moreover, the well-known Wedderburn-Artin's theorem ([11, Theorem 4.4, p. 112]) ensures that when $\mathbb{F}_q[G]$ is semisimple it can be realised as a direct sum of matrix algebras.

Theorem II.1 (Wedderburn-Artin's decomposition for finite group algebras)**.** *Let G be a finite group such that $\mathbb{F}_q[G]$ is a semisimple group algebra. Then $\mathbb{F}_q[G]$ is isomorphic, as $\mathbb{F}_q$-algebra, to the direct sum of some matrix rings over suitable extensions of $\mathbb{F}_q$. Specifically, one has:*

$$\mathbb{F}_q[G] \cong \bigoplus_{i=1}^{s} \mathcal{M}_{n_i}(\mathbb{F}_{q^{r_i}})$$

satisfying $|G| = \sum_{i=1}^{s} n_i^2 r_i$.

A *linear code* $\mathcal{C} \subseteq \mathbb{F}_q^n$ is just an $\mathbb{F}_q$-vector subspace of $\mathbb{F}_q^n$. We assume that the reader is familiar with the fundamentals of linear codes, which can be found in any basic book on coding theory. Under certain conditions, some linear codes can be viewed as ideals of a group algebra, thus taking advantage of this algebraic structure for their study. More specifically, given a group $G = \{g_1, \ldots, g_n\}$, we say that a linear code $\mathcal{C} \subseteq \mathbb{F}_q^n$ is a (left) *G-code* if there exists a bijection $\theta : \{1, \ldots, n\} \longrightarrow G$ such that the set

$$I_{\mathcal{C}} = \left\{ \sum_{i=1}^{n} a_i \theta(i) \mid (a_1, \ldots, a_n) \in \mathcal{C} \right\} \tag{1}$$

is a (left) ideal of the group algebra $\mathbb{F}_q[G]$.

II-B. On Quantum Codes

A $[[n, k, d]]_q$ *quantum error-correcting code* Q is a subspace of the n-fold tensor product $(\mathbb{C}^q)^{\otimes n}$ of the complex vector space $\mathbb{C}^q$. The code Q has length n, dimension $\dim_{\mathbb{C}^q}(\mathcal{Q}) = q^k$ and minimum distance d, i. e., any error acting on at most $d-1$ of the tensor factors can be detected or has no effect on the code. This kind of codes are useful to reduce the effects of environmental and operational noise in quantum information processing. For more on quantum codes, see, for example, [19]. Here we focus on the so-called stabilizer codes based on linear codes over finite fields and, within them, we deal with *quantum CSS codes*, whose name is due to Calderbank, Shor and Steane. The basic theory of this kind of codes can be found on [18]. In particular, our goal will be to construct quantum codes by using $\mathbb{F}_{q^2}$-linear codes, since in this case more efficient algorithms to compute their minimum distance can be used (see [16]).

Given two elements $x = (x_1, x_2, ..., x_n)$ and $y = (y_1, y_2, ..., y_n)$ in $\mathbb{F}_{q^2}^n$, their *hermitian inner-product* is defined as $x \cdot_H y = \sum_{i=1}^{n} x_i y_i^q$. Thus, the *hermitian dual code* of a linear code $\mathcal{C}$ of $\mathbb{F}_{q^2}^n$ is

$$\mathcal{C}^{\perp_H} = \{ y \in \mathbb{F}_{q^2}^n \ : \ x \cdot_H y = 0, \ \forall\, x \in \mathcal{C} \}.$$

The code $\mathcal{C}$ is called *hermitian self-orthogonal* if $\mathcal{C} \subseteq \mathcal{C}^{\perp_H}$, that is, $\mathcal{C}$ is contained in its hermitian dual. The CSS-construction of quantum codes using this metric is the following:

Theorem II.2 ([16])**.** *Let $\mathcal{C} \subseteq \mathbb{F}_{q^2}^n$ be an hermitian self-orthogonal linear code, that is, $\mathcal{C} \subseteq \mathcal{C}^{\perp_H}$. If $\mathcal{C}$ is an $[n, k]_{q^2}$-code then there exists an $[[n, n-2k, d_{\mathcal{Q}}]]_q$ quantum stabiliser code $\mathcal{Q}$ with*

$$d_{\mathcal{Q}} = \begin{cases} d(\mathcal{C}^{\perp_H}) & \text{when } \mathcal{C} = \mathcal{C}^{\perp_H} \\ d(\mathcal{C}^{\perp_H} \setminus \mathcal{C}) \geq d(\mathcal{C}^{\perp_H}) & \text{when } \mathcal{C} \neq \mathcal{C}^{\perp_H} \end{cases}.$$

We can transfer the hermitian metric to the group algebra, in order to work better with group codes. To do so, we use the antialgebra automorphism of $\mathbb{F}_q[G]$ given by $\hat{a} = \sum_{g \in G} a_g g^{-1}$ for all $a = \sum_{g \in G} a_g g$. Given $a, b \in \mathbb{F}_{q^2}[G]$, their hermitian inner-product is

$$a \cdot_H b = \Big(\sum_{g \in G} a_g g\Big) \cdot_H \Big(\sum_{g \in G} b_g g\Big) = \sum_{g \in G} a_g b_g^q.$$

Moreover, given a linear code $\mathcal{C}$ of $\mathbb{F}_{q^2}^n$, the ideal $I_{\mathcal{C}^{\perp_H}}$ of $\mathbb{F}_{q^2}[G]$ associated to $\mathcal{C}^{\perp_H}$ can be computed as

$$I_{\mathcal{C}^{\perp_H}} = \{ \hat{a}^q \in \mathbb{F}_{q^2}[G] \mid ua = 0, \forall u \in I_{\mathcal{C}} \}.$$

III. A suitable decomposition of $\mathbb{F}_{q^2}[D_n]$

When the group algebra $\mathbb{F}_q[G]$ is semisimple, the decomposition provided by the Wedderburn-Artin's theorem (Thm. II.1) has been concretised in several cases. With regard to dihedral groups, such a decomposition is given in [6, Theorem 3.1] and we are going to rewrite it appropiately to address the study of the hermitian dual of any dihedral code over $\mathbb{F}_{q^2}$.

Given a polynomial $f = \sum_{i=0}^{m} a_i \mathrm{x}^i \in \mathbb{F}_{q^2}[\mathrm{x}]$, we denote by $\overline{f} = \sum_{i=0}^{m} a_i^q \mathrm{x}^i$ its *conjugate polynomial*. In particular, f is called *self-conjugate* whenever $f = \overline{f}$. We denote by f^* the *reciprocal* polynomial of f, i.e. $f^*(\mathrm{x}) = f(0)^{-1} \mathrm{x}^{deg(f)} f(\mathrm{x}^{-1})$. The polynomial f is said to be *self-reciprocal* if $f = f^*$. We write $f^\dagger$ for the conjugate of the reciprocal polynomial of f, that is, $f^\dagger = \overline{f^*}$.

Notation III.1. *We decompose the polynomial $\mathrm{x}^n - 1$ into irreducible factors in $\mathbb{F}_{q^2}[\mathrm{x}]$ as follows:*

$$\mathrm{x}^n - 1 = \prod_{j \in J_0} f_j \prod_{j \in J_1} f_j \overline{f_j} \prod_{j \in J_2 \cup J_3} f_j f_j^* \prod_{j \in J_4} f_j f_j^* \overline{f_j} f_j^\dagger, \tag{2}$$

where:

- $j \in J_0$ *if and only if* $f_j = \overline{f_j} = f_j^*$. *In this case,* $f_j \in \{\mathrm{x}-1, \mathrm{x}+1\}$.
- $j \in J_1$ *if and only if* $f_j = \underline{f_j^*}$ *but* $f_j \neq \overline{f_j}$.
- $j \in J_2$ *if and only if* $\underline{f_j} = \overline{f_j}$ *but* $f_j \neq f_j^*$
- $j \in J_3$ *if and only if* $\overline{f_j} = f_j^*$ *but* $f_j \neq \underline{f_j^*}$
- $j \in J_4$ *if and only if* $f_j, \overline{f_j}$ *and* f_j^* *are all different.*

Set $r_j = \deg(f_j)$, and $J = \bigcup_{i=0}^{4} J_i$. Given the group algebra $\mathbb{F}_{q^2}[D_n]$, we restate below the isomorphism given in [6, Theorem 3.1] by considering the decomposition (2) and by grouping together those blocks associated to pairs of conjugate polynomials.

Below α_j denotes an arbitrary root of f_j, and C_2 is the cyclic group of order 2 generated by the matrix $\left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$.

Theorem III.2. *If* $\mathrm{char}(\mathbb{F}_q) \nmid n$ *then there exists an isomorphism of* $\mathbb{F}_{q^2}$*-algebras*

$$\mathbb{F}_{q^2}[D_n] \overset{\rho}{\cong} \bigoplus_{j \in J} A_j,$$

where A_j *is equal to*

- $\mathbb{F}_{q^2} \oplus \mathbb{F}_{q^2}$, *if* $j \in J_0$ *and* $2 \nmid q$.
- $\mathbb{F}_{q^2}[C_2]$, *if* $j \in J_0$ *and* $2 \mid q$.
- $\mathcal{M}_2\left(\mathbb{F}_{q^2}(\alpha_j + \alpha_j^{-1})\right) \oplus \mathcal{M}_2\left(\mathbb{F}_{q^2}(\alpha_j^q + \alpha_j^{-q})\right)$, *if* $j \in J_1$.
- $\mathcal{M}_2\left(\mathbb{F}_{q^2}(\alpha_j)\right)$, *if* $j \in J_2 \cup J_3$.
- $\mathcal{M}_2\left(\mathbb{F}_{q^2}(\alpha_j)\right) \oplus \mathcal{M}_2\left(\mathbb{F}_{q^2}(\alpha_j^q)\right)$, *if* $j \in J_4$.

IV. On the hermitian dual of a $\mathbb{F}_{q^2}[D_n]$-code

The aim of this subsection is to obtain a full description of the hermitian dual of a dihedral code based on its expression as a direct sum of ideals within the Wedderburn-Artin's decomposition of $\mathbb{F}_{q^2}[D_n]$ provided in Theorem III.2.

Consider the automorphism $\sigma : x \mapsto x^q$ of $\mathbb{F}_{q^2}$. It is easy to see that, if $\mathcal{C} \subseteq \mathbb{F}_{q^2}$ is a linear code, then $\mathcal{C}^{\perp_H}$ can be computed from $\mathcal{C}^{\perp_E}$ by simply applying σ to every component of each codeword of $\mathcal{C}^{\perp_E}$. Taking into account this observation, and the description of the euclidean dual of any dihedral code already given in [26], our goal is then to find φ such that the following diagram is commutative. By conj_q we mean the natural linear extension of σ to $\mathbb{F}_{q^2}[D_n]$.

$$\begin{array}{ccc} \mathbb{F}_{q^2}[D_n] & \xrightarrow{\mathrm{conj}_q} & \mathbb{F}_{q^2}[D_n] \\ \downarrow{\scriptstyle \rho} & & \downarrow{\scriptstyle \rho} \\ \bigoplus_{j \in J} A_j & \xrightarrow{\varphi} & \bigoplus_{j \in J} A_j \end{array} \tag{3}$$

Theorem IV.1. *Following Notation III.1, the diagram (3) is commutative if*

$$\varphi = \bigoplus_{j \in J} \varphi_j,$$

with

- $\varphi_j(x \oplus y) = x^q \oplus y^q$, *if* $j \in J_0$ *and* $2 \nmid q$;
- $\varphi_j(X) = \mathcal{H}^1(X)$, *if* $j \in J_0$ *and* $2 \mid q$;
- $\varphi_j(X \oplus Y) = \left(\sigma_j \circ \mathcal{H}^{2r_j-1} \circ \overline{\sigma_j}^{-1}\right)(Y) \oplus \left(\overline{\sigma_j} \circ \mathcal{H}^1 \circ \sigma_j^{-1}\right)(X)$, *if* $j \in J_1$;
- $\varphi_j(X) = \mathcal{H}^{r_j}(X)$, *if* $j \in J_2$;
- $\varphi_j(X) = \mathcal{T}^{r_j}(X)$, *if* $j \in J_3$;
- $\varphi_j(X \oplus Y) = \mathcal{H}^{2r_j-1}(Y) \oplus \mathcal{H}^1(X)$, *if* $j \in J_4$,

where, for any $m \in \mathbb{Z}$, $\mathcal{H}^m$ *and* $\mathcal{T}^m$ *are the following maps of* $\mathcal{M}_2\left(\mathbb{F}_{q^{2s_j}}\right)$, *with* $s_j = r_j/2$ *for* $j \in J_1$ *and* $s_j = r_j$ *for* $j \in J \setminus J_1$, *:*

$$\mathcal{H}^m : \begin{bmatrix} a & b \\ c & d \end{bmatrix} \mapsto \begin{bmatrix} a^{q^m} & b^{q^m} \\ c^{q^m} & d^{q^m} \end{bmatrix}, \quad \mathcal{T}^m : \begin{bmatrix} a & b \\ c & d \end{bmatrix} \mapsto \begin{bmatrix} d^{q^m} & c^{q^m} \\ b^{q^m} & a^{q^m} \end{bmatrix}.$$

We are now in a position to state the result that will give us the complete description of the hermitian dual of a dihedral code.

Theorem IV.2. *Let* $(q, n) = 1$. *Let* $I_{\mathcal{C}} \subseteq \mathbb{F}_{q^2}[D_n]$ *be the ideal corresponding to a dihedral code* $\mathcal{C}$ *over* $\mathbb{F}_{q^2}$. *Following Notation III.1, one has*

$$I_{\mathcal{C}} \cong \bigoplus_{j \in J} \mathcal{I}_j,$$

where $\mathcal{I}_j$ *is an ideal of* A_j. *Set* $\mathcal{I}_j = \langle X \rangle \oplus \langle Y \rangle$ *for* $j \in J_1 \cup J_4$, *and* $\mathcal{I}_j = \langle X \rangle$ *for* $j \in J_2 \cup J_3$, *with* X, Y *matrices in row reduced echelon form. Then the ideal corresponding to hermitian dual code* $\mathcal{C}^{\perp_H}$ *is*

$$I_{\mathcal{C}^{\perp_H}} \cong \bigoplus_{j \in J} \widehat{\mathcal{I}}_j,$$

where:

$j \in J_0$ and $2 \nmid q$:

$$\widehat{\mathcal{I}}_j = \begin{cases} \mathbb{F}_{q^2} \oplus \mathbb{F}_{q^2} & \text{if } \mathcal{I}_j = \mathbf{0} \oplus \mathbf{0} \\ \mathbf{0} \oplus \mathbb{F}_{q^2} & \text{if } \mathcal{I}_j = \mathbb{F}_{q^2} \oplus \mathbf{0} \\ \mathbb{F}_{q^2} \oplus \mathbf{0} & \text{if } \mathcal{I}_j = \mathbf{0} \oplus \mathbb{F}_{q^2} \\ \mathbf{0} \oplus \mathbf{0} & \text{if } \mathcal{I}_j = \mathbb{F}_{q^2} \oplus \mathbb{F}_{q^2} \end{cases}$$

$j \in J_0$ and $2 \mid q$:

$$\widehat{\mathcal{I}}_j = \begin{cases} \mathbb{F}_{q^2}[C_2] & \text{if } \mathcal{I}_j = \mathbf{0} \\ \left\langle \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \right\rangle & \text{if } \mathcal{I}_j = \left\langle \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \right\rangle \\ \mathbf{0} & \text{if } \mathcal{I}_j = \mathbb{F}_{q^2}[C_2] \end{cases}$$

$j \in J_1$: $\widehat{\mathcal{I}}_j = \langle Y_1 \rangle \oplus \langle X_1 \rangle$, and one of the following holds for X_1 and Y_1:

- $X_1 = I - X$, *if* $X = \mathbf{0}$ *or* $X = I$;
- $X_1 = \begin{bmatrix} 2 & -(\alpha_j + \alpha_j^{-1})^q \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$;
- $X_1 = \begin{bmatrix} (2\lambda_j + \alpha_j + \alpha_j^{-1})^q & (-2 - (\alpha_j + \alpha_j^{-1})\lambda_j)^q \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix}$, $\lambda_j \in \mathbb{F}_{q^{r_j}}$,
- $Y_1 = I - Y$, *if* $Y = \mathbf{0}$ *or* $Y = I$;
- $Y_1 = \begin{bmatrix} 2 & -(\alpha_j + \alpha_j^{-1}) \\ 0 & 0 \end{bmatrix}$, *if* $Y = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$;
- $Y_1 = \begin{bmatrix} 2\lambda_j^{q^{2r_j-1}} + \alpha_j + \alpha_j^{-1} & -2 - (\alpha_j + \alpha_j^{-1})\lambda_j^{q^{2r_j-1}} \\ 0 & 0 \end{bmatrix}$, *if* $Y = \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix}$, $\lambda_j \in \mathbb{F}_{q^{r_j}}$.

$j \in J_2$: $\widehat{\mathcal{I}}_j = \langle X_1 \rangle$, and one of the following holds:

- $X_1 = I - X$, *if* $X = \mathbf{0}$ *or* $X = I$;
- $X_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$;

- $X_1 = \begin{bmatrix} 1 & -\lambda_j^{q^{r_j}} \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix}$, $\lambda_j \in \mathbb{F}_{q^{2r_j}}$;

$j \in J_3$: $\widehat{\mathcal{I}}_j = \langle X_1 \rangle$, *and one of the following holds:*

- $X_1 = I - X$, *if* $X = \mathbf{0}$ *or* $X = I$;
- $X_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$;
- $X_1 = \begin{bmatrix} -\lambda_j^{q^{r_j}} & 1 \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix}$, $\lambda_j \in \mathbb{F}_{q^{2r_j}}$.

$j \in J_4$: $\widehat{\mathcal{I}}_j = \langle Y_1 \rangle \oplus \langle X_1 \rangle$, *and one of the following holds for X_1 and Y_1:*

- $X_1 = I - X$, *if* $X = \mathbf{0}$ *or X=I;*
- $X_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$;
- $X_1 = \begin{bmatrix} 1 & -\lambda_j^{q} \\ 0 & 0 \end{bmatrix}$, *if* $X = \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix}$, $\lambda_j \in \mathbb{F}_{q^{r_j}}$;
- $Y_1 = I - Y$, *if* $Y = \mathbf{0}$ *or* $Y = I$;
- $Y_1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, *if* $Y = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$;
- $Y_1 = \begin{bmatrix} 1 & -\lambda_j^{q^{2r_j-1}} \\ 0 & 0 \end{bmatrix}$, *if* $Y = \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix}$, $\lambda_j \in \mathbb{F}_{q^{r_j}}$.

V. Hermitian self-orthogonal $\mathbb{F}_{q^2}[D_n]$-codes and related quantum codes

In this section we obtain a full algebraic description of any hermitian self-orthogonal dihedral code. As a result, we will be able to compute the total number of dihedral codes $\mathcal{C}$ over $\mathbb{F}_{q^2}$ that are hermitian self-orthogonal and also to construct quantum error-correcting codes via the CSS-method (see Theorem II.2).

Theorem V.1. *Let $(q, n) = 1$. Following the notation in Theorem IV.2, consider the expression $\rho(\mathcal{C}) = \bigoplus_{j \in J} \mathcal{I}_j$ of a dihedral code $\mathcal{C}$ over $\mathbb{F}_{q^2}$. Then $\mathcal{C} \subseteq \mathcal{C}^{\perp_H}$ if and only if the following conditions are satisfied:*

- *If $j \in J_0$ and q is odd, then $\mathcal{I}_j = \mathbf{0} \oplus \mathbf{0}$.*
- *If $j \in J_0$ and q is even, then $\mathcal{I}_j = \mathbf{0} \oplus \mathbf{0}$ or $\mathcal{I}_j = \langle \left[\begin{smallmatrix} 1 & 1 \\ 0 & 0 \end{smallmatrix}\right] \rangle$.*
- *If $j \in J_1$, then either one of the summands of $\mathcal{I}_j$ is $\mathbf{0}$, or $\mathcal{I}_j$ has the form:*
  - $\left\langle \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 2 & -(\alpha_j^q + \alpha_j^{-q}) \\ 0 & 0 \end{bmatrix} \right\rangle$,
  - $\left\langle \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} (2\lambda_j + \alpha_j + \alpha_j^{-1})^q & [-2 - \lambda_j(\alpha_j + \alpha_j^{-1})]^q) \\ 0 & 0 \end{bmatrix} \right\rangle$.
- *If $j \in J_2$, then $\mathcal{I}_j$ is either $\mathbf{0}$, $\left\langle \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right\rangle$, or $\left\langle \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix} \right\rangle$ with $\lambda_j = -\lambda_j^{q^{r_j}}$.*
- *If $j \in J_3$, then $\mathcal{I}_j$ is either $\mathbf{0}$, or $\left\langle \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix} \right\rangle$ with $0 \neq \lambda_j = -\lambda_j^{-q^{r_j}}$.*
- *If $j \in J_4$, then either one of the summands of $\mathcal{I}_j$ is $\mathbf{0}$, or $\mathcal{I}_j$ has the form:*
  - $\left\langle \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \right\rangle$,
  - $\left\langle \begin{bmatrix} 1 & \lambda_j \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 1 & -\lambda_j^q \\ 0 & 0 \end{bmatrix} \right\rangle$.

As a consequence of this result, we can compute how many hermitian self-orthogonal dihedral codes there are in the group algebra $\mathbb{F}_{q^2}[D_n]$.

Corollary V.2. *Let $(q, n) = 1$. Then the number of dihedral codes $\mathcal{C}$ over $\mathbb{F}_{q^2}$ that are hermitian self-orthogonal, i.e. such that $\mathcal{C} \subseteq \mathcal{C}^{\perp_H}$, is*

- $\prod_{j \in J_1} (3q^{r_j} + 6) \prod_{j \in J_2 \cup J_3} (q^{r_j} + 2) \prod_{j \in J_4} \left(3q^{2r_j} + 6\right)$, *if* $2 \nmid q$.
- $2 \prod_{j \in J_1} (3q^{r_j} + 6) \prod_{j \in J_2 \cup J_3} (q^{r_j} + 2) \prod_{j \in J_4} \left(3q^{2r_j} + 6\right)$, *if* $2 \mid q$.

Finally, using Theorem V.1, the CCS-construction for hermitian self-orthogonal codes, allow us to build quantum error-correcting codes. Next we present an example computed with MAGMA wich is optimal.

Example V.3. *Let $q = 9$ and D_{16} the dihedral group of order 32. Let ω be a primitive element of $\mathbb{F}_9$ and ν a primitive element of $\mathbb{F}_{9^2}$. Following Notation 2, consider the factorisation of $\mathrm{x}^{16} - 1$ in $\mathbb{F}_9[\mathrm{x}]$:*

$$\mathrm{x}^{16} - 1 = (\mathrm{x}-1)(\mathrm{x}+1)(\mathrm{x}+\omega)(\mathrm{x}+\omega^2)(\mathrm{x}+\omega^3)(\mathrm{x}+\omega^5)$$
$$(\mathrm{x}+\omega^6)(\mathrm{x}+\omega^7)(\mathrm{x}^2+\omega)(\mathrm{x}^2+\omega^3)(\mathrm{x}^2+\omega^5)(\mathrm{x}^2+\omega^7).$$

Thus, $f_1 = \mathrm{x} - 1$, $f_2 = \mathrm{x} + 1$, $f_3 = \mathrm{x} + \omega$, $f_4 = \mathrm{x} + \omega^2$, $\bar{f}_3 = \mathrm{x} + \omega^3$, $\bar{f}_3{}^* = \mathrm{x} + \omega^5$, $f_4^* = \bar{f}_4 = \mathrm{x} + \omega^6$, $f_3^* = \mathrm{x} + \omega^7$, $f_3 = \mathrm{x}^2 + \omega$, $\bar{f}_5 = \mathrm{x}^2 + \omega^3$, $\bar{f}_5{}^* = \mathrm{x}^2 + \omega^5$ and $f_5^* = \mathrm{x}^2 + \omega^7$. Therefore, $J_0 = \{1, 2\}$, $J_1 = J_2 = \emptyset$, $J_3 = \{4\}$ and $J_4 = \{3, 5\}$ and the group algebra $\mathbb{F}_9[D_{16}]$ decomposes in the following way by Theorem III.2:

$$\mathbb{F}_9[D_{16}] \cong (\mathbb{F}_9 \oplus \mathbb{F}_9) \oplus (\mathbb{F}_9 \oplus \mathbb{F}_9) \oplus \mathcal{M}_2(\mathbb{F}_9)$$
$$\oplus (\mathcal{M}_2(\mathbb{F}_9) \oplus \mathcal{M}_2(\mathbb{F}_9)) \oplus (\mathcal{M}_2(\mathbb{F}_{9^2}) \oplus \mathcal{M}_2(\mathbb{F}_{9^2})) =$$
$$= 4\mathbb{F}_9 \oplus \mathcal{M}_2(\mathbb{F}_9) \oplus 2\mathcal{M}_2(\mathbb{F}_9) \oplus 2\mathcal{M}_2(\mathbb{F}_{9^2}).$$

Consider now the dihedral code $\mathcal{C}$ over $\mathbb{F}_9$ such that

$$I_{\mathcal{C}} \cong (\mathbf{0} \oplus \mathbf{0}) \oplus (\mathbf{0} \oplus \mathbf{0}) \oplus \left\langle \begin{bmatrix} 1 & \omega^7 \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \right\rangle$$
$$\oplus \left\langle \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 1 & \nu^{14} \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 1 & \nu^2 \\ 0 & 0 \end{bmatrix} \right\rangle.$$

Then by Theorem IV.2 we can stay that

$$I_{\mathcal{C}^{\perp_{\mathcal{H}}}} \cong (\mathbb{F}_9 \oplus \mathbb{F}_9) \oplus (\mathbb{F}_9 \oplus \mathbb{F}_9) \oplus \left\langle \begin{bmatrix} 1 & \omega^7 \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix} \right\rangle$$
$$\oplus \mathcal{M}_2(\mathbb{F}_9) \oplus \left\langle \begin{bmatrix} 1 & \nu^{14} \\ 0 & 0 \end{bmatrix} \right\rangle \oplus \left\langle \begin{bmatrix} 1 & \nu^2 \\ 0 & 0 \end{bmatrix} \right\rangle.$$

Thus $\mathcal{C} \subseteq \mathcal{C}^{\perp_H}$ and therefore $\mathcal{C}$ is an hermitian self-orthogonal code. Consequently, Theorem II.2 ensures the existence of a QECC with parameters $[[32, 8, 8]]_3$, which is optimal.

This method also yields, for instance, the existence QECC having parameters $[[32, 16, 6]]_3$, $[[34, 10, 7]]_2$ and $[[42, 12, 8]]_2$.

Referencias

[1] S.D. Berman: On the theory of group codes. *Cybernetics* **3**(1) (1967), 25–31.

[2] J.J. Bernal, A. del Río and J.J. Simón: An intrinsical description of group codes. *Des. Codes Cryptogr.* **51**(3) (2009), 289–300.

[3] M. Borello, J. De La Cruz and W. Willems: On checkable codes in group algebras. *J. Algebra Appl.* **21**(6) (2022), 2250125.

[4] M. Borello and W. Willems: Group codes over fields are asymptotically good. *Finite Fields Their Appl.* **68** (2020), 101738.

[5] N. Bourbaki: *Algebra: Chapter 8.* Springer Nature. Switzerland, 2023.

[6] F.E. Brochero-Martínez: *Structure of finite dihedral group algebra*. *Finite Fields Appl.* **35** (2015), 204–214.

[7] F.E. Brochero-Martínez, L. Batista de Oliveira and C. R. Giraldo Vergara: Wedderburn decomposition and idempotents of some finite metacyclic group algebras. *Preprint arXiv: 2210.07226* (2022).

[8] Y. Cao, Y. Cao and F.W. Fu: Hermitian duality of left dihedral codes over finite fields. *Discrete Math.*, **346**(1) (2023), 113179.

[9] Y. Cao, Y. Cao and F. Ma: Construction and enumeration of left dihedral codes satisfying certain duality properties. *Discrete Math.* **345**(11) (2022), 113059.

[10] P. Charpin: *idéaux de certaines algèbres modulaires*. Publication de l'Université de Paris, vol. VII. Thèse de 3eme cycle, 1982.

[11] K. Doerk and T. Hawkes: *Finite soluble groups.* De Gruyter Expositions in Mathematics 4. Berlin, 1992.

[12] F.S. Dutra, R.A. Ferraz and F.C. Polcino-Milies: Semisimple group codes and dihedral codes. *Algebra Discrete Math.* **8**(3) (2009), 2–48.

[13] Y. Gao, Q. Yue and Y. Wu: LCD codes and self-orthogonal codes in generalized dihedral group algebras. *Des. Codes Cryptogr.* **88**(11) (2020), 2275–2287.

[14] Y. Gao and Q. Yue: Idempotents of generalized quaternion group algebras and their applications. *Discrete Math.* **344**(5) (2021), 112342.

[15] S. González, V. Markov, O. Markova and C. Martínez: Group codes. *Algebra, codes and cryptology: First international conference, A2C 2019 in honor of Prof. Mamadou Sanghare* (2019), 83–96.

[16] M. F. Ezerman, M. Grass, S. Ling, F. Özbudak and B. Özkaya: Characterization of nearly self-orthogonal quasi-twisted codes and related quantum codes. *IEEE Trans. Inform. Theory* **71**(1) (2024), 499–517.

[17] A.V. Kelarev and P. Solé: Error-correcting codes as ideals in group rings. *Contemp. Math.* **273** (2001), 11–18.

[18] A. Ketkar, A. Klappenecker, S. Kumar, and P. K. Sarvepalli: Nonbinary stabilizer codes over finite fields. *IEEE Trans. Inform. Theory* **52**(11) (2006), 4892-4914.

[19] G.G. La Guardia,: *Quantum error correction*. Quantum science and technology. Springer, 2020.

[20] P. Landrock and O. Manz: Classical codes as ideals in group algebras. *Des. Codes Cryptogr.* **2** (1992), 273–285.

[21] M.F. MacWilliams: Binary codes which are ideals in the group algebra of an abelian group. *Bell Syst. Tech. J.* **49**(6) (1970), 987–1011.

[22] S. Perlis and G.L. Walker: Abelian group algebras of finite order. *Trans. Amer. Math. Soc.* **3** (68) (1950), 420–426.

[23] R.R. de Araujo, F.C. Polcino-Milies and R.A.Ferraz: Minimal group codes over alternating groups. *Preprint arXiv 2310.17375* (2023).

[24] M. Sales-Cabrera, X. Soler-Escrivà and V. Sotomayor: Codes in algebras of direct products of groups and associated quantum codes. *Preprint arXiv: 2412.09695v2* (2025).

[25] N. Sendrier: Code-based cryptography: State of the art and perspectives. *IEEE Secur. Priv.* **4** (2017), 44–50.

[26] K.V. Vedenev and V.M. Deundyak: Some properties of dihedral group codes. *Preprint arXiv: 2005.08283* (2021).

Rompiendo Barreras en el Entorno Médico: un Framework Seguro Basado en Identidad para el Control de Acceso

Antonio López Martínez*, Montassar Naghmouchiº, Maryline Laurentº,
Joaquin Garcia-Alfaroº, Manuel Gil Pérez*, Antonio Ruiz Martínez*
*Facultad de Informática, Universidad de Murcia, España
ºSAMOVAR, Télécom SudParis, Institut Polytechnique de Paris, 91120 Palaiseau, France
*{antonio.lopez41, mgilperez, arm}@um.es
º{montassar-bellah_naghmouchi, maryline.laurent, joaquin.garcia_alfaro}@telecom-sudparis.eu

***Resumen*—Este es un resumen extendido de la publicación [1].**

La digitalización de los datos sanitarios ha intensificado los desafíos de seguridad, privacidad e interoperabilidad. Este trabajo aborda estos retos proponiendo un framework integral basado en la Identidad Auto-Soberana (SSI) para garantizar un acceso seguro y controlado a los registros de salud. Introducimos un novedoso y detallado modelo de control de acceso basado en roles (RBAC) que identifica a todos los actores relevantes del ecosistema sanitario y mapea sus permisos de acceso a recursos específicos del estándar Fast Healthcare Interoperability Resources (FHIR). El modelo propuesto se implementa mediante un conjunto de cuatro contratos inteligentes (smart contracts) personalizados en la plataforma blockchain Hyperledger Fabric, asegurando un control de acceso auditable y automatizado. La viabilidad y escalabilidad del framework se demuestran a través de una implementación de prueba de concepto y rigurosos tests de rendimiento, que finalmente se validan frente a la escala real de un sistema nacional de salud.

***Index Terms*—SSI, Blockchain, Control de acceso basado en roles, Seguridad**

I. Introducción

La creciente digitalización de los Registros Electrónicos de Salud (EHR) ha expuesto al sector sanitario a riesgos significativos. Ataques como el sufrido por Change Healthcare en 2024 [2], que explotó la falta de autenticación multifactor, subrayan la vulnerabilidad de los sistemas centralizados. En este contexto, normativas como el General Data Protection Regulation (GDPR) y la ley Health Insurance Portability and Accountability Act (HIPAA) imponen requisitos estrictos para la protección de datos, exigiendo soluciones que garanticen la seguridad y la privacidad del paciente.

La Identidad Auto-Soberana (SSI) emerge como un paradigma prometedor, devolviendo al individuo el control sobre su identidad digital mediante el uso de Identificadores Descentralizados (DIDs) y Credenciales Verificables (VCs). Si bien trabajos previos han explorado el uso de blockchain para la gestión de EHRs, como MedChain de Daraghmi et al. [3] o la wallet de identidad de Harrel et al. [4], nuestro análisis de la literatura revela brechas significativas. Específicamente, estas propuestas a menudo presentan modelos de roles incompletos, carecen de una definición formal del EHR basada en estándares modernos como FHIR, o no validan su escalabilidad en escenarios realistas a nivel nacional.

Para abordar estas limitaciones, nuestro trabajo original [1] presenta un marco integral cuyas contribuciones clave, que lo diferencian del estado del arte, son:

- **Identificación exhaustiva de actores:** Se define un modelo de roles completo que va más allá de los actores tradicionales.
- **Definición formal del EHR:** Se utiliza el estándar FHIR para estructurar los datos de salud, garantizando la interoperabilidad.
- **Control de acceso robusto:** Se diseña una arquitectura de contratos inteligentes que automatiza y audita el acceso a los datos.
- **Validación a escala nacional:** Se valida la solución obteniendo los datos de un caso de uso realista y se compara su rendimiento con las necesidades de un sistema sanitario nacional.

II. Modelo de control de acceso basado en roles

La base de nuestro framework es un modelo de control de acceso que define con precisión quién puede acceder a qué información y por qué.

II-A. Roles para el entorno médico

Se realizó un análisis profundo para identificar a todos los participantes que interactúan con los datos del paciente. El modelo resultante, ilustrado en la Figura 1, considera al paciente como el centro del ecosistema. Se definen roles con responsabilidades y permisos específicos, aplicando el principio de minimización de datos. Por ejemplo:

- **Proveedor Especialista:** Accede a recursos como ‘Condition‘, ‘Observation‘ y ‘DiagnosticReport‘ para realizar un diagnóstico, pero no al ‘CarePlan‘ general por defecto.
- **Investigador Médico:** Accede a los mismos recursos que un especialista, pero de forma **anonimizada**, para no comprometer la identidad del paciente, salvo en ensayos clínicos con consentimiento explícito.
- **Aseguradora:** Su acceso se limita a recursos financieros como ‘Claim‘ y ‘ExplanationOfBenefit‘ para gestionar los pagos, sin necesidad de visualizar datos clínicos sensibles.

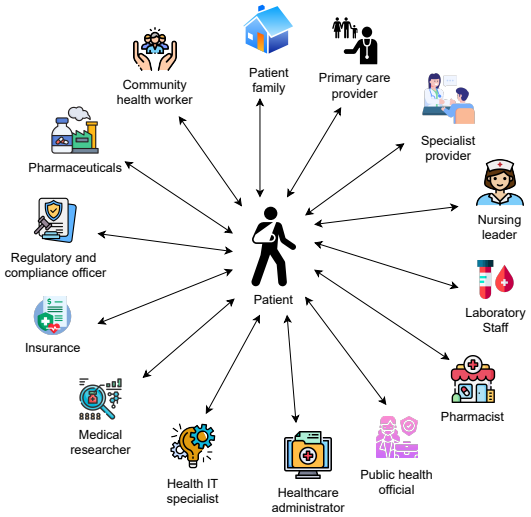


Figura 1. Actores del entorno médico

II-B. Definición del EHR en base al estándar FHIR

La elección del estándar **FHIR (Fast Healthcare Interoperability Resources)** fue estratégica. A diferencia de estándares legados como HL7v2 o CDA, que son más rígidos y complejos, FHIR se basa en tecnologías web modernas (APIs RESTful, JSON), lo que simplifica drásticamente la integración y el intercambio de datos en tiempo real. Su naturaleza modular, basada en "recursos", permite un control de acceso granular, esencial para nuestro modelo. La contribución aquí fue seleccionar de todo el conjunto de recursos que componen el estándar FHIR, el mínimo de ellos que permite definir el EHR de un paciente. Algunos de los recursos clave seleccionados en nuestro trabajo son:

- **Patient:** Contiene la información demográfica del paciente. Es el recurso central al que se vinculan casi todos los demás.
- **Encounter:** Representa una interacción entre un paciente y un proveedor de salud (ej. una consulta, una hospitalización). Actúa como un contenedor contextual para otros eventos clínicos.
- **Observation:** Almacena mediciones y resultados de laboratorio (signos vitales, análisis de sangre, etc.). Es uno de los recursos más consultados.
- **MedicationRequest:** Formaliza una orden o prescripción de un medicamento, esencial para la comunicación entre médicos y farmacias.
- **DiagnosticReport:** Agrupa los resultados e interpretaciones de un estudio diagnóstico (ej. un informe de radiología).
- **Consent:** Permite registrar el consentimiento del paciente para compartir sus datos en diferentes contextos (investigación, emergencias), una pieza clave en un marco SSI.

II-C. Validación del modelo con casos de uso

Para validar y complementar el modelo de acceso basado en roles y permisos definidos, en el artículo original se detallaron tres casos de uso realistas con la intención de estudiar si el diseño encajaba con estos casos de uso y estudiar posibles necesidades no contempladas:

1. **Ciclo de vida de una muestra de paciente :** Desde la cita con el médico de atención primaria hasta la recepción de la medicación en la farmacia, demostrando la interacción entre múltiples roles (médico, enfermera, personal de laboratorio, especialista, farmacéutico), y el conjunto de recursos FHIR accedido por cada uno de ellos. Una descripción más detallada de este caso de uso se puede encontrar en [5].
2. **Ensayo Clínico:** Se modela el flujo completo, desde el diseño del ensayo por una farmacéutica hasta el reclutamiento de pacientes y el análisis de resultados, involucrando a roles como investigadores, oficiales de cumplimiento y especialistas.
3. **Caso de Emergencia:** Se simula un accidente en el que un paciente inconsciente es atendido por servicios de emergencia, activando los protocolos de acceso de urgencia definidos en la arquitectura.

III. Arquitectura técnica y Smart Contracts

La implementación del modelo RBAC se realiza sobre una arquitectura descentralizada, cuya lógica reside en una serie de contratos inteligentes.

III-A. Arquitectura de la plataforma

La arquitectura (Figura 2) fue diseñada en un primer trabajo [6], creando el framework SSI con los siguientes dos componentes principales:

- **Wallet de Usuario:** Una wallet basada en agentes, edge y cloud. Gracias al agente edge, instalado en el dispositivo móvil del usuario, se gestionan Identificadores Descentralizados (DIDs), Credenciales Verificables (VCs) y claves criptográficas. Por el contrario, el agente cloud permite la comunicación entre wallets, guarda una copia de seguridad de los datos médicos, y permite la comunicación con la red Blockchain.
- **Plataforma Blockchain:** Una red permisionada **Hyperledger Fabric** que actúa como Registro de Datos Verificables (VDR) y aloja la lógica de control de acceso a través de contratos inteligentes (chaincodes). Esta elección se debe a sus características idóneas para entornos empresariales: canales privados para la segregación de datos, gestión de identidades mediante MSPs (Membership Service Providers) y alto rendimiento.

Además, se puede observar como en la plataforma existe un canal global entre organizaciones y canales locales en cada organización. Los canales locales se debe al modelado de identidades que cambian de rol entre organizaciones, por ejemplo, médicos, que pueden actuar como médicos especialistas en cierta organización, y medicos de cabecera en otros. En cuanto al canal global, aquí se guarda el rol de médico de urgencias, para situaciones de emergencias donde es necesario acceder a la información de los pacientes obteniendo un acceso especial para ello.

III-B. Diseño de los Smart Contracts

Se diseñaron cuatro contratos inteligentes (chaincodes) modulares para gestionar el acceso:

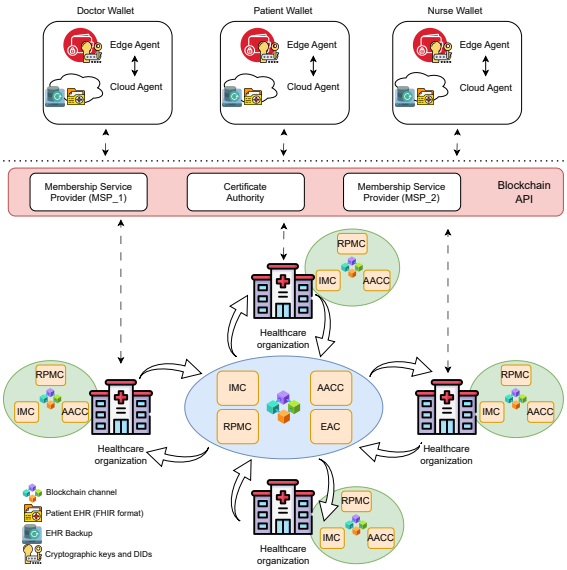


Figura 2. Arquitectura de la plataforma

- **Identity Management Contract (IMC):** Funciona como un VDR. Almacena las claves públicas de los usuarios, permitiendo que cualquier participante verifique criptográficamente la identidad de otro. La necesidad de este Smart Contract nace de poder verificar al emisor de un VC y registrar entidades validas en la plataforma.
- **Role and Permission Management Contract (RPMC):** Es el cerebro del control de acceso. Permite a las organizaciones (ej. un hospital) registrar su modelo de roles y asignar dichos roles a los DIDs de sus empleados. Cuando un usuario solicita acceso a cierto recurso del EHR del paciente, este contrato verifica si su rol le otorga los permisos necesarios sobre el recurso solicitado.
- **Audit and Compliance Contract (AACC):** Garantiza la trazabilidad total. Cada interacción relevante (creación de identidad, asignación de rol, solicitud de acceso) genera un evento que es registrado inmutablemente por el AACC, facilitando auditorías y el cumplimiento de normativas.
- **Emergency Access Contract (EAC):** Gestiona el acceso en situaciones críticas. Define un "canal de emergencia" compartido por hospitales y servicios de urgencias. Un médico autorizado puede invocar este contrato para generar un *token de emergencia (ET)* que le otorga acceso temporal al EHR de un paciente inconsciente. El proceso notifica al paciente (quien puede objetar si está consciente) y queda registrado en el AACC.

IV. Implementación y Análisis de Rendimiento

Se desplegó una red de pruebas con 5 organizaciones, 7 nodos y 2 orderers para procesar transacciones en Hyperledger Fabric v2.5.10. La implementación de los Smart Contracts fue a través de chaincodes escritos con el lenguaje TypeScript. Para las pruebas de rendimiento, se utilizó la herramienta **Hyperledger Caliper**, un benchmark proporcionado por Hyperledger para ejecutar pruebas masivas. Para el caso de este estudio, se realizaron dos tipos de pruebas, estrés y lineales, evaluando las operaciones más frecuentes de lectura (lectura de roles, de DIDs, etc.) y escritura (registrar claves, asignar roles, etc.).

IV-A. Resultados obtenidos

La Figura 3 presenta los resultados obtenidos demostrando que el modelo de Smart Contracts tenía un comportamiento robusto y eficiente. El **test de estrés**, diseñado para encontrar el punto de ruptura del sistema, demostró que la plataforma podía soportar picos extremos de 1000 TPS manteniendo un 100 % de éxito para un lote de 10,000 transacciones.

El **test lineal**, que simula condiciones de operación normales, arrojó resultados más matizados y reveladores:

- **Operaciones de Lectura:** Funciones como "GetRoles" o "GetPermissionsForRole" mostraron latencias extremadamente bajas y un 100 % de éxito incluso a tasas de envío de 500 TPS. Esto es crucial, ya que la mayoría de las interacciones diarias en un hospital (consultar un historial, verificar un permiso) son operaciones de lectura.
- **Operaciones de Escritura:** Funciones como "AssignRole" o "RegisterPublicKey" mostraron una latencia mayor y una tasa de éxito decreciente a partir de 300 TPS. Este comportamiento es inherente y esperado en una blockchain. Las escrituras requieren un proceso de consenso entre los nodos de la red para garantizar la integridad y la inmutabilidad del registro, lo cual es computacionalmente más costoso. La tasa de fallo a altas cargas no se debió a errores en el chaincode, sino al agotamiento del tiempo de espera de la transacción (60s por defecto en Fabric), lo cual es un mecanismo de protección de la red.

En conclusión, la plataforma está optimizada para el patrón de uso del mundo real: consultas rápidas y frecuentes, y escrituras menos frecuentes pero seguras y auditables.

V. Validación de los Resultados en un Caso de Estudio Nacional

Para validar si el rendimiento observado era suficiente, se estimó la carga de transacciones que generaría el sistema sanitario español. Se partió de los siguientes datos base extraídos de fuentes públicas:

- Población de España: $\approx$ 49 millones de personas.
- Hospitales: 845 (resultando en $\approx$ 58,333 personas por hospital).
- Consultas médicas anuales por persona: $\approx$ 14.
- Tasa de accidentes laborales (extrapolada a la población): 3.7 %.
- Ensayos clínicos en 2024: 843.

A partir de estos datos, se calculó la carga de TPS para los diferentes tipos de transacciones definidas en nuestros casos de uso:

1. **Transacciones Diarias (canal local):** Se calculó el número de citas por hospital y se multiplicó por los 6 actores que interactúan en el caso de uso y las 4 operaciones diarias por actor.
   (58,333 pers/hosp x 14 cons/año) x 6 actores x 4 op = 19.6M trans/año

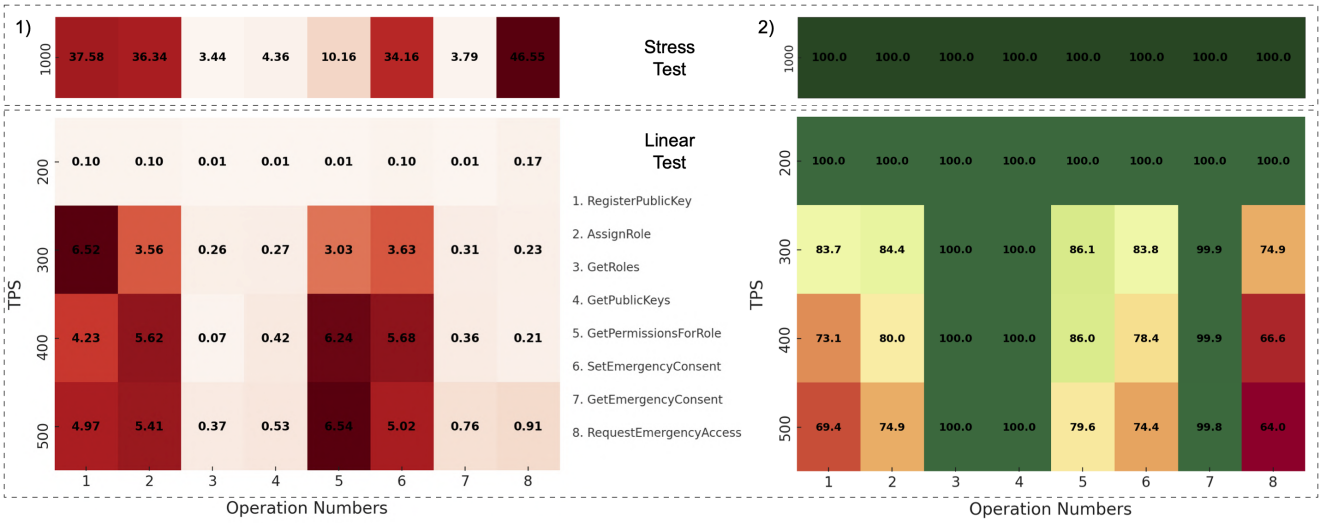


Figura 3. Resultados de los tests de rendimiento

Esto se traduce en **0.62 TPS** por canal de organización local.

2. **Transacciones de Emergencia (canal global):** Se calculó el número de personas accidentadas y se multiplicó por los actores y operaciones del caso de uso de emergencia.
   (49M pers x 3.7 %) x 4 actores x 5 op = 43.5M trans/año = **1.14 TPS**
3. **Otras transacciones (globales/locales):** Cálculos similares para registros (7.76 TPS), actualizaciones (9.32 TPS) y ensayos clínicos (0.64 TPS) completaron la estimación.

V-A. Estudio de los resultados obtenidos

La suma de las cargas de transacciones resultó en una necesidad estimada de $\approx$ **18.86 TPS para el canal multi-organización** y $\approx$ **17.72 TPS para los canales locales**. Comparando estos requisitos con el rendimiento demostrado de nuestra plataforma (operación estable a **200 TPS**), se concluye que la solución no solo es viable, sino que opera con un margen de eficiencia de más de un orden de magnitud, garantizando su capacidad para soportar la carga de todo un país.

VI. Conclusión y vías futuras

El trabajo original presenta un marco SSI robusto y completo para la gestión de la identidad y el acceso en el sector sanitario. La combinación de un modelo de roles detallado, el estándar FHIR y una arquitectura de smart contracts en Hyperledger Fabric proporciona una solución segura, auditable y centrada en el paciente. La validación experimental confirma su escalabilidad para un despliegue a nivel nacional.

Como trabajo futuro, se explorará la mejora de la integración entre las wallets SSI y la lógica en blockchain para permitir un intercambio de datos directo *wallet-to-wallet*, gobernado por las políticas de la red. Además, se investigará la incorporación de agentes de Inteligencia Artificial locales en las wallets para actuar como asesores de privacidad, ayudando a los usuarios a tomar decisiones informadas sobre con quién y qué datos compartir.

Referencias

[1] López Martínez, A., Naghmouchi, M., Laurent, M., García Alfaro, J., Gil Pérez, M., Ruiz Martínez, A.: Breaking barriers in healthcare: A secure identity framework for seamless access. Computer Standards & Interfaces, 95, 104020 (2026). doi:10.1016/j.csi.2025.104020

[2] Energy and Commerce: What We Learned: Change Healthcare Cyber Attack. (2024). https://energycommerce.house.gov/posts/what-we-learned-change-healthcare-cyber-attack.

[3] Daraghmi, E.-Y., Daraghmi, Y.-A., Yuan, S.-M.: MedChain: A Design of Blockchain-Based System for Medical Records Access and Permissions Management. IEEE Access, 7, 164595-164613 (2019). doi:10.1109/ACCESS.2019.2952942

[4] Harrell, D.T., Usman, M., Hanson, L., Abdul-Moheeth, M., et al.: Technical Design and Development of a Self-Sovereign Identity Management Platform for Patient-Centric Health Care Using Blockchain Technology. Blockchain in Healthcare Today, 5 (2022). doi:10.30953/bhty.v5.197

[5] López Martínez, A., Gil Pérez, M., Ruiz Martínez, A.: A comprehensive model for securing sensitive patient data in a clinical scenario. IEEE Access, 11, 137083-137098 (2023). doi:10.1109/ACCESS.2023.3338170

[6] López Martínez, A., Naghmouchi, M., Laurent, M., Garcia-Alfaro, J., Gil Pérez, M., Ruiz Martínez, A., Nespoli, P.: Empower healthcare through a Self-Sovereign Identity infrastructure for secure electronic health data access. arXiv preprint arXiv:2501.12229 (2025). https://arxiv.org/abs/2501.12229

Estudio de la ciclicidad de códigos grupo binarios sobre grupos no abelianos de orden menor que 24

Beatriz García García
Universidad de Oviedo,
Dpto. Matemáticas,
C/ Leopoldo Calvo Sotelo, 18,
33007 Oviedo (Asturias)
garciagbeatriz@uniovi.es

Consuelo Martínez López
Universidad de Oviedo,
Dpto. Matemáticas,
C/ Leopoldo Calvo Sotelo, 18,
33007 Oviedo (Asturias)
cmartinez@uniovi.es

Ignacio F. Rúa
Universidad de Oviedo,
Dpto. Matemáticas,
C/ Leopoldo Calvo Sotelo, 18,
33007 Oviedo (Asturias)
rua@uniovi.es

***Resumen*—El estudio de códigos correctores de errores para ser utilizados en criptografía postcuántica es una de las líneas de investigación actuales en la comunidad científica. En la RECSI 2024 se presentó un estudio realizado en esta línea en el que se incluye la clasificación, salvo equivalencia por permutación, de códigos grupo binarios de longitud menor que 24. Este trabajo sigue la estela del mencionado anteriormente, estudiando el comportamiento de los códigos grupo binarios construidos sobre grupos no abelianos de orden menor que 15 y sobre los grupos diédricos hasta orden 22, para comparar el comportamiento con respecto a los códigos obtenidos sobre grupos abelianos.**

***Index Terms*—códigos correctores de errores, códigos grupo, criptografía postcuántica, seguridad información.**

I. Introducción

La criptografía basada en códigos correctores de errores se encuentra actualmente entre las propuestas más esperanzadoras de métodos resistentes a la computación cuántica. Por tanto, parece natural plantearse la utilización de códigos grupo para posibles aplicaciones en criptografía.

Un código lineal $\mathcal{C}$ de longitud n es un código grupo sobre un cuerpo finito $\mathbb{F}$, si existen un grupo finito $G := \{g_1 = e, g_2, \ldots, g_n\}$ y una aplicación biyectiva $\nu : \{1, \ldots, n\} \to G$ que lleva $i \to g_i$ tal que $\{\sum_{i=1}^{n} a_i g_i : (a_1, \ldots, a_n) \in \mathcal{C}\}$ es un ideal bilátero del álgebra de grupo $\mathbb{F}G$. Nótese que, en el último capítulo de este trabajo, se consideran también para su comparación con los códigos grupo, los códigos grupo a derecha. En este caso, los ideales no son biláteros, sino únicamente ideales a derecha.

Para poder usar una familia de códigos grupo en criptografía se necesita disponer de buenos algoritmos de descodificación. En este área se encuentran los trabajos [1], [2], [3]. Es necesario, además, conocer bien la estructura y propiedades de esta familia de códigos ([4], [5], [6], [7], [8]). También interesa estudiar la indistinguibilidad de dichos códigos entre códigos lineales arbitrarios, propiedad importante para su uso en criptografía [9], si bien este no es el objetivo central de este trabajo.

Cabe destacar [8] que todos los códigos hasta orden 23 se pueden ver como códigos grupo abelianos. También se demuestra en [8] que existen códigos grupo que no se pueden ver como códigos sobre un grupo abeliano.

En las actas de la RECSI 2024 [10], se presentó un listado de códigos grupo binarios de longitudes menor que 24, que permitió extraer algunas propiedades de la familia de códigos grupo. Puesto que estos códigos son todos códigos grupo abelianos, solo se necesita considerar códigos sobre los grupos abelianos. Se aprecia que los códigos grupo no cíclicos solo aparecen en las longitudes 8, 9, 12, 16, 18 y 20. Todos los códigos lineales cíclicos obtenidos, salvo equivalencia por permutación, se pueden ver en un álgebra de grupo con grupo abeliano no cíclico. En contraste con el caso semisimple, sobre el cuerpo de dos elementos, existen códigos grupo de longitud 16 y dimensión 7 que no se pueden obtener a través de un ideal principal en el álgebra de cualquier grupo abeliano de orden 16. También se vio que existen códigos grupo no cíclicos de longitud 16 y dimensión 8 que no se pueden ver como G-códigos para algún G grupo abeliano no cíclico de orden 16.

En este trabajo se plantea el comportamiento de los grupos no abelianos. En concreto, se quiere ver si todos los códigos grupo que han aparecido, y, en particular, los códigos grupo cíclicos, se pueden ver como códigos grupo sobre un grupo no abeliano.

En la primera parte se estudian los códigos grupo sobre todos los grupos no abelianos de orden menor o igual que 15. Se quiere tener una visión completa que permita ver un patrón de los códigos considerados.

En una segunda parte, se consideran códigos grupo de longitud comprendida entre 16 y 23, pero centrando el estudio únicamente en los grupos diédricos de orden menor que 24.

Finalmente, se considera también códigos grupo a derecha (es decir, que aparecen ligados a un ideal a derecha del álgebra de grupo) y se compara el comportamiento con los casos estudiados anteriormente.

Nótese que, en todo el trabajo, se identifican dos códigos grupo si son permutacionalmente equivalentes.

II. Códigos grupo binarios sobre grupos no abelianos de orden menor o igual que 15

En adelante, n denotará siempre la longitud de un código, que, en caso de ser código grupo, coincidirá con el orden del grupo. Para cada código, k denotará siempre la dimensión del código y d su distancia mínima. Se denotará [n,k,d]-código a un código de longitud n, dimensión k y distancia mínima d.

Se dirá que un código grupo sobre un cuerpo finito $\mathbb{F}$ se realiza en un grupo G si existe un código grupo en $\mathbb{F}G$ permutacionalmente equivalente a él.

n	k	Grupos Abelianos		D_3	
		No cíclicos	cíclicos	No cíclicos	cíclicos
6	1	0	1	0	1
	2	0	2	0	1
	3	0	1	0	0

Figura 1. Códigos abelianos de longitud 6 sobre grupos no abelianos

n	k	Grupos Abelianos		D_4		Q_8	
		No cíclicos	cíclicos	No cíclicos	cíclicos	No cíclicos	cíclicos
8	1	0	1	0	1	0	1
	2	0	1	0	1	0	1
	3	0	1	0	1	0	1
	4	1	1	1	1	1	1

Figura 2. Códigos abelianos de longitud 8 sobre grupos no abelianos

n	k	Grupos Abelianos		D_5	
		No cíclicos	cíclicos	No cíclicos	cíclicos
10	1	0	1	0	1
	2	0	1	0	1
	3	0	0	0	0
	4	0	1	0	0
	5	0	1	0	0

Figura 3. Códigos abelianos de longitud 10 sobre grupos no abelianos

n	k	Grupos Abelianos		D_6		A_4		DC_3	
		No cíclicos	cíclicos	No cíclicos	cíclicos	No cíclicos	cíclicos	No cíclicos	cíclicos
12	1	0	1	0	1	0	1	0	1
	2	0	2	0	1	0	1	0	1
	3	0	2	0	1	0	1	0	1
	4	1	3	0	2	1	0	0	2
	5	0	2	0	1	0	1	0	1
	6	2	3	1	1	0	0	0	1

Figura 4. Códigos abelianos de longitud 12 sobre grupos no abelianos

Se sabe que hasta longitud 15 todos los códigos cíclicos se realizan en cualquier grupo abeliano de ese orden y los códigos no cíclicos se realizan en todos los grupos abelianos no cíclicos de ese orden, cabe pensar si pasa lo mismo con los grupos no abelianos.

Se estudia, a continuación, en detalle para qué ordenes los códigos abelianos se pueden realizar también en algún grupo no abeliano y cuándo los cíclicos pueden aparecer en un grupo no abeliano.

n	k	Grupos Abelianos		D_7	
		No cíclicos	cíclicos	No cíclicos	cíclicos
14	1	0	1	0	1
	2	0	1	0	1
	3	0	1	0	0
	4	0	1	0	0
	5	0	1	0	0
	6	0	2	0	0
	7	0	2	0	0

Figura 5. Códigos abelianos de longitud 14 sobre grupos no abelianos

II-A. Orden 6: D_3

En este caso ya se observa que hay códigos grupo cíclicos que no se realizan en el único grupo que no es abeliano (*Fig. 1*). Con la misma dimensión, dimensión 2, hay uno que se realiza y otro que no, y en dimensión 3 el único que hay no se realiza.

II-B. Orden 8: D_4 y Q_8

En este caso (*Fig. 2*) los códigos grupo cíclicos y los no cíclicos se realizan en todos los grupos no abelianos. Se podría pensar que esto pasa siempre que es un grupo de orden potencia de 2, pero este hecho no es cierto, como se verá al estudiar longitud 16.

II-C. Orden 10: D_5

Todos los códigos grupo de longitud 10 son cíclicos, pero hay dos de ellos en las dimensiones 4 y 5 que no se realizan en D_5 (*Fig. 3*).

II-D. Orden 12: D_6, A_4 y DC_3

Los códigos grupo de longitud 12 presentan una mayor variedad. Hay códigos grupo que se realizan en todos los grupos abelianos, otros que se realizan sólo en alguno de ellos y otros que no se realizan en ninguno.

En dimensiones 2, 3 y 5, se puede observar (*Fig. 4*) que, de los dos cíclicos que hay en cada dimensión, solo se realiza uno de ellos en cada grupo no abeliano. ¿Será siempre el mismo código cíclico? Se puede demostrar que no, en dimensiones 2 y 3, un cíclico se realiza en D_6 y DC_3 pero no en A_4, mientras que otro se realiza únicamente en el alternado.

Hay cíclicos que no se realizan en ningún grupo no abeliano. En dimensión 6 hay tres cíclicos, de distancias mínimas 2, 3 y 4, respectivamente. Los de distancia mínima 3 y 4 no se realizan en ningún grupo no abeliano de esta longitud. El de distancia mínima 2 se realiza en los grupos D_6 y DC_3 pero no en el grupo A_4.

Respecto a los códigos grupo no cíclicos, en dimensión 4 el no cíclico solo se conserva en el A_4, y en dimensión 6 de los dos no cíclicos solo se conserva uno en el D_6.

II-E. Orden 14: D_7

En longitud 14 todos los códigos grupo son cíclicos. De orden 14 hay el D_7 como no abeliano y solo hay códigos en dimensiones 1 y 2 (*Fig. 5*).

n	k	Grupos Abelianos		D_8	
		No cíclicos	cíclicos	No cíclicos	cíclicos
16	1	0	1	0	1
	2	0	1	0	1
	3	0	1	0	1
	4	1	1	1	1
	5	1	1	0	1
	6	2	1	0	1
	7	4	1	0	1
	8	6	1	1	1

Figura 6. Códigos abelianos de longitud 16 sobre grupos no abelianos

n	k	Grupos Abelianos		D_9	
		No cíclicos	cíclicos	No cíclicos	cíclicos
18	1	0	1	0	1
	2	0	2	0	1
	3	0	1	0	0
	4	1	2	0	1
	5	1	1	0	1
	6	2	2	0	1
	7	1	1	0	0
	8	3	2	0	0
	9	2	1	0	0

Figura 7. Códigos abelianos de longitud 18 sobre grupos no abelianos

III. Códigos grupo binarios sobre D_n, $8 \leq n \leq 11$

En esta sección ya no se consideran todos los grupos no abelianos y el estudio se centra únicamente en los grupos diédricos que aparecen entre los órdenes 16 y 23.

III-A. Diédrico de orden 16*:* D_8

En todas las dimensiones (*Fig. 6*), los códigos grupo cíclicos se recuperan en el diédrico, sin embargo, de los no cíclicos se pierden la mayoría, solo se conserva el de dimensión 4 y uno de dimensión 8, ambos códigos se realizan en todo grupo abeliano no cíclico de orden 16.

III-B. Diédrico de orden 18*:* D_9

Sobre el diédrico D_9 (*Fig. 7*), se observa que hay solo códigos grupo cíclicos y solo en ciertas dimensiones.

III-C. Diédrico de orden 20*:* D_{10}

Sobre el diédrico D_{10}, como se puede ver en *Fig. 8*, se pierden tanto códigos grupo cíclicos como no cíclicos.

III-D. Diédrico de orden 22*:* D_{11}

Todos los códigos grupo de longitud 22 son cíclicos (*Fig. 9*) y no todos ellos se pueden realizar en el grupo diédrico D_{11}.

n	k	Grupos Abelianos		D_{10}	
		No cíclicos	cíclicos	No cíclicos	cíclicos
20	1	0	1	0	1
	2	0	1	0	1
	3	0	1	0	1
	4	0	2	0	1
	5	0	1	0	0
	6	0	1	0	0
	7	0	1	0	0
	8	2	2	0	1
	9	2	1	0	1
	10	4	1	1	1

Figura 8. Códigos abelianos de longitud 20 sobre grupos no abelianos

n	k	Grupos Abelianos		D_{11}	
		No cíclicos	cíclicos	No cíclicos	cíclicos
22	1	0	1	0	1
	2	0	1	0	1
	3	0	0	0	0
	4	0	0	0	0
	5	0	0	0	0
	6	0	0	0	0
	7	0	0	0	0
	8	0	0	0	0
	9	0	0	0	0
	10	0	1	0	0
	11	0	1	0	0

Figura 9. Códigos abelianos de longitud 22 sobre grupos no abelianos

IV. Códigos grupo binarios a derecha sobre D_n, para $3 \leq n \leq 11$

El programa computacional con el que se extraen los códigos grupo en este estudio es Magma [11], el cuál trabaja con ideales a derecha.

En las *Fig. 10, 11 y 12*, los códigos grupo a derecha que aparecen en la última columna son aquellos que provienen de un ideal a derecha sobre el correspondiente D_n, pero no se pueden obtener como ideales biláteros de un álgebra de grupo del mismo orden.

Se analizan los códigos grupo a derecha sobre D_n, para $3 \leq n \leq 11$, salvo equivalencia por permutación, con el objetivo de visualizar, por una parte, si aparecen más códigos grupo a derecha que no se pueden obtener como códigos grupo; y por otra parte, si hay códigos que no se pueden realizar en el grupo diédrico pero sí aparecen como código grupo a derecha.

Excepto en longitudes 18 y 22, no aparecen códigos grupo a derecha en D_n que no hayan aparecido como códigos grupo.

Los códigos grupo cíclicos no se recuperan en todos los

n	k	Grupos Abelianos		D_7		
		No cíclicos	cíclicos	No cíclicos	cíclicos	A derecha nuevos
14	1	0	1	0	1	0
	2	0	1	0	1	0
	3	0	1	0	0	0
	4	0	1	0	0	0
	5	0	1	0	0	0
	6	0	2	0	2	0
	7	0	2	0	2	0

Figura 10. Códigos grupo a derecha sobre el D_7

n	k	Grupos Abelianos		D_9		
		No cíclicos	cíclicos	No cíclicos	cíclicos	A derecha nuevos
18	1	0	1	0	1	0
	2	0	2	0	2	0
	3	0	1	0	1	0
	4	1	2	0	2	0
	5	1	1	0	1	0
	6	2	2	0	2	0
	7	1	1	0	1	0
	8	3	2	0	2	1
	9	2	1	0	1	1

Figura 11. Códigos grupo a derecha sobre el D_9

diédricos, en longitud 14 (*Fig. 10*), por ejemplo, todos los códigos grupo son cíclicos, y no se recuperan cada uno de los tres cíclicos de dimensiones 3, 4 y 5.

Respecto a los códigos grupo abelianos no cíclicos, que solo aparecen en las longitudes 8, 12, 16, 18 y 20, solo se recuperan todos ellos en longitud 8.

Sobre el diédrico D_9 (*Fig. 11*), los códigos grupo a derecha nuevos aparecen en dimensiones 8, 9 y 10.

Sobre el diédrico D_{11} (*Fig. 12*), solo aparecen códigos grupo a derecha nuevos en dimensiones 10, 11 y 12.

V. Conclusiones

Analizando las figuras presentadas se observa que en general utilizando grupos no abelianos se obtienen menos códigos que utilizando grupos abelianos.

El comportamiento del grupo diédrico es desigual. En unas longitudes hay códigos grupo cíclicos que se pueden realizar como códigos sobre el diédrico y en otras hay códigos grupo cíclicos que no se pueden realizar sobre el grupo diédrico.

Se puede observar que en los grupos diédricos D_{2p} con p primo impar, aparecen códigos grupo solamente en dimensiones 1, 2, $2p-2$ y $2p-1$. Parece natural plantearse que existan únicamente esos cuatro ideales propios en cada una de las álgebras de grupo $\mathbb{F}_2 D_{2p}$.

n	k	Grupos Abelianos		D_{11}		
		No cíclicos	cíclicos	No cíclicos	cíclicos	A derecha nuevos
22	1	0	1	0	1	0
	2	0	1	0	1	0
	3	0	0	0	0	0
	4	0	0	0	0	0
	5	0	0	0	0	0
	6	0	0	0	0	0
	7	0	0	0	0	0
	8	0	0	0	0	0
	9	0	0	0	0	0
	10	0	1	0	1	1
	11	0	1	0	1	1

Figura 12. Códigos grupo a derecha sobre el D_{11}

Se aprecia que, salvo para D_9 y D_{11}, no se obtienen códigos nuevos utilizando ideales a derecha, y que todos ellos se pueden obtener con ideales biláteros.

Agradecimientos

Este trabajo fue financiado por la Unión Europea a través de INCIBE dentro del Plan de Recuperación, Transformación y Resiliencia – NextGenerationEU en el marco de la Agencia España Digital 2026 (proyecto MRR-MAETD:24-INCIBE-001).

Referencias

[1] C. Martínez, F. Molina (2023). The syndromes decoding algorithm in group codes. Finite Fields and Their Applications, 89, 102206.

[2] C. Martínez, F. Molina (2022). Two Decoding Algorithms in Group Codes, Libro de Actas XVII, Reunión Española sobre Criptología y Seguridad de la Información. Universidad de Cantabria, Santander-España, 157-161.

[3] J.J. Bernal, A. del Río, J.J. Simón (2012). Partial Permutation Decoding for Abelian Codes. IEEE Transactions on Information Theory, 59 (8), 274-278.

[4] C. García, S. González, V. Markov, C. Martínez (2017). Non-abelian group codes over an arbitrary finite field. Journal of Mathematical Sciences, 223 (5), 504-507.

[5] C. García, S. González, V. Markov, C. Martínez, A. Nechaev (2016). New examples of non-abelian group codes. Advances in Mathematics of Communications, 10 (1), 1-10.

[6] C. García, S. González, V. Markov, C. Martínez, A. Nechaev (2013). Group codes over non-abelian groups. Journal of Algebra and Its Applications, 12 (7), 1350037.

[7] J.J. Bernal, A. del Río, J.J. Simón (2009). An intrinsical description of group codes. Designs, Codes and Cryptography, 51 (3), 289-300.

[8] C. García Pillado, S. González, V. Markov, C. Martínez, A. Nechaev (2011). Group codes which are not Abelian group codes. Journal of Algebra and Its Applications, 12 (7), 123-127.

[9] A. Piñera, I. F. Rúa, A. Suárez (2025). On the square code of group codes. Finite Fields and Their Applications, 103, 102548.

[10] B. García, I. F. Rúa (2024). Clasificación de códigos grupo binarios de longitud menor que 24, XVIII Reunión Española sobre Criptología y Seguridad de la Información. Universidad de León, 193-197.

[11] W. Bosma, J. Cannon, C. Playoust (1997). The Magma Algebra System I: The User Language. Journal of Symbolic Computation, 24 (3-4), 235–265.

A Taxonomy and Security Analysis of Rollup Architectures in Ethereum Blockchain

Adrià TORRALBA-AGELL*, Muoi TRAN†, Cristina PÉREZ-SOLÀ‡
*Universitat Oberta de Catalunya, *atorralbaag@uoc.edu*
†Chalmers University of Technology and University of Gothenburg, *muoi@chalmers.se*
‡Universitat Autònoma de Barcelona, *cristina.perez@uab.cat*

***Abstract*—Layer 2 rollups improve Ethereum's scalability by executing transactions off-chain while relying on on-chain verification for finality. Despite their growing importance with increasingly high stakes, rollups have received limited systematic security analysis. In this paper, we present a taxonomy of rollup architectures and examine their associated security threats. We identify three main architectural patterns, primarily determined by the design of the sequencer, a critical component that bridges off-chain transaction execution with on-chain finality. Our analysis also reveals inherent trade-offs between throughput, decentralization, and resilience, demonstrating that no single architecture simultaneously optimizes all these properties.**

Index Terms— Network Attacks, Rollups, Sequencer Architectures, Layer 2 Scaling, Denial-of-Service

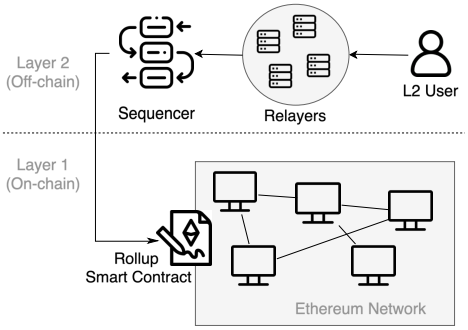


Figure 1. The transaction workflow in Ethereum rollups.

I. Introduction

Layer 2 (L2) scaling solutions have achieved significant adoption on the Ethereum blockchain, securing over $40 billion in total value locked [1] and processing more than 85% of Ethereum ecosystem transactions [2]. Among many L2 solutions, *rollups* have emerged as the primary paradigm, enabling near-instant transaction finality and high throughput. Rollup solutions are often grouped based on their security guarantee mechanisms, namely, optimistic rollups with interactive dispute workflow on Layer 1 (L1) [3] and ZK-rollups with validating zero-knowledge proofs [4]. At the core of any rollup solution lies a critical component called *sequencer* that orders and executes the user-submitted L2 transactions before batching results and posting them to L1.

With the extremely high stakes involved, L2 systems naturally exhibit security risks, several of which have already materialized in real-world incidents. In December 2023, the Arbitrum One sequencer experienced a critical two-hour outage when a software bug, exacerbated by high inscription-driven throughput, caused the system to stall [5]. Also, in June 2024, the Linea zkEVM rollup unilaterally paused its sequencer to censor attacker addresses following a $6.8 million exploit at the Velocore DEX [6], [7]. More recently, a Starknet sequencer failure in September 2025 caused a 3-hour outage and even forced a block reorganization [8]. Collectively, these incidents underscore the need for a deeper understanding of weak points in such emerging L2 solutions.

However, there exist many rollup implementations — L2Beat lists 24 different solutions, and this list is constantly growing [1]. Analyzing each of the solutions for their security risks is obviously not scalable. Moreover, to the best of our knowledge, there exists no comprehensive framework that identifies the threat models of rollup solutions in Ethereum.

This paper contends that all rollup solutions adhere to a limited set of recurring architectural patterns built from shared core components within their transaction workflows. To support this claim, we present a systematic taxonomy of Ethereum rollup architectures, focusing on their sequencer designs (§III). We also find that because these architectures share common components, rollups are exposed to similar attack surfaces and require comparable threat models (§IV). We conclude that no existing rollup design simultaneously satisfies the demands of throughput, decentralization, and resilience, and we outline several directions for future research (§V).

II. Background

A. Motivation for Rollups

Ethereum derives strong security guarantees from global consensus over the platform's state, which imposes a fundamental scalability limitation on transaction throughput captured by the blockchain trilemma [9]. To overcome this bottleneck without compromising security and decentralization, Ethereum has adopted a rollup-centric roadmap [10]. In this model, Ethereum L1 acts primarily as a secure settlement and data-availability layer, while L2 systems handle scalable transaction execution. Rollups accomplish this by moving execution off-chain, batching results back to L1 for verification, and relying on cryptographic proofs to guarantee the correctness of state transitions [11].

B. Identities and Roles in Rollups

Despite considerable diversity in their deployment models, Ethereum rollups share a common high-level architecture composed of four key entities, see Figure 1.

At the edge of the system are *users*, who submit transactions through publicly accessible RPC endpoints. From a user perspective, the desired properties mirror those of Ethereum itself: timely confirmation, deterministic execution, and eventual finality on L1. While users interact solely through wallet interfaces, their experience depends heavily on the downstream infrastructure responsible for confirming their transactions.

Sitting between users and the rollup protocol is a layer of *relayers* (i.e., public or semi-public RPC endpoints) that ingest user transactions and forward them to the sequencer. These relayers exhibit substantial variation in operational redundancy, geographic deployment, and trust assumptions, all of which directly influence transaction latency, censorship resistance, and privacy. In current rollup deployments, most public RPC relayers are operated either by third-party infrastructure providers (e.g., Alchemy [12], Infura [13]) or by the rollup teams themselves. They are typically easy to discover and are often openly listed in public registries [14].

The *sequencer* is the core operational component of any rollup. It can be a single operator, a set of redundant operators, or (in emerging systems) a distributed protocol. The sequencer is responsible for ordering incoming transactions, executing them against the L2 state machine, producing updated state roots, and periodically batching transaction data for submission to Ethereum L1. By maintaining a private mempool with full visibility into pending transactions, the sequencer determines the rollup's throughput, latency, and fairness properties.

Finally, each rollup includes a *verifier* contract deployed on Ethereum L1. The verifier acts as the authoritative on-chain arbiter of state correctness. It checks rollup state transitions using one of two verification models: fraud proofs (in optimistic rollups) or validity proofs (in ZK rollups). Regardless of the mechanism, the verifier enforces the rollup's security guarantees by accepting only state roots that can be cryptographically justified, thereby anchoring all off-chain activity to Ethereum's base consensus rules.

C. Transaction Workflow in Rollups

A user transaction in a rollup system progresses through three distinct stages, spanning off-chain execution, on-chain commitment, and eventual verification on L1 (Figure 1).

Stage 1: off-chain execution. The workflow begins when the sequencer receives a user transaction via RPC relayers. The sequencer orders the transaction deterministically relative to other pending transactions and executes it against the current L2 state machine, producing an updated state root. From the user's perspective, this yields a *soft confirmation*, i.e., rapid acknowledgment that the transaction has been accepted into the sequencer's private mempool. Importantly, soft confirmations rely on sequencer honesty, as until the results are posted and verified on L1, they remain provisional and subject to reordering or censorship.

Stage 2: on-chain commitment and data availability. Executed transactions are then aggregated into batches and posted to Ethereum L1 as either calldata [15] or blob data (EIP-4844 [16]), along with the sequencer's claimed post-state root. This posting provides *data availability*, the property that all transaction data necessary to reconstruct the rollup's state is made publicly accessible. Note that data availability enables credible neutrality, meaning anyone can run a full rollup node alongside a full Ethereum node by retrieving posted transaction data and re-executing state transitions, thereby verifying the state independently of the sequencer [17]. Also, data availability ensures forced withdrawal safety because even if the sequencer halts or becomes malicious, users can construct Merkle proofs of fund ownership and withdraw assets to L1, provided historical transaction data remains available on-chain [17]. Moreover, data availability serves as a foundational security property: if data were unavailable, users could neither independently verify the rollup's state nor withdraw funds in the case of a non-cooperative sequencer. Once a batch is posted, it enters a verification window on L1, whose duration and semantics depend on the underlying verification model.

Stage 3: verification and finality. The final stage occurs on Ethereum L1, where the verifier contract checks the sequencer's state transition claims using one of two following fundamentally different mechanisms. First, in optimistic rollups, commitments are accepted tentatively and placed into a challenge window (e.g., 7–14 days). During this period, any observer may present a fraud proof demonstrating that the sequencer produced an incorrect state transition. A successful challenge results in rejection of the batch, slashing of the sequencer's bond, and a reward to the challenger. If no valid challenge arises, the batch is finalized at the end of the window. Second, in the cryptographic proof model, ZK rollups accompany each batch with a succinct validity proof (e.g., a zk-SNARK [18] or zk-STARK [19]) attesting to the correctness of all included state transitions. The verifier contract checks the proof deterministically, and upon acceptance, finalizes the batch immediately without any delay or dispute period. Both workflows preserve the essential data availability invariant. Particularly, all transaction data is retained on L1, allowing independent state reconstruction and ensuring that users retain a safe withdrawal path even in the presence of sequencer failure or malicious behavior [20].

III. Rollup Infrastructures

We classify rollup systems based on the architecture of their most critical component, namely the sequencer. Indeed, although rollups vary widely in design goals and implementation details, their sequencers fall into three broad architectural patterns that meaningfully shape security, performance, and failure modes, i.e., stand-alone (§III-A), pipeline (§III-B), and distributed (§III-C). We also show all studied rollups and their corresponding architectures in Table I.

Stand-alone	Stand-alone (HA)	Pipeline	Distributed
Polygon zkEVM	Arbitrum One	Linea	Lighter
ZKSync Era	Base	Starknet	INTMAX
ZKSync Lite	Optimism	Katana	Facet Bluebird
Scroll	Unichain	Paradex	ZKBase
Zk.Money v1	Ink	Abstract	Starknet Grinta
Loopring	BOB	ZERO	
Zircuit			
Phala			
Space and Time			

Table I
EXISTING ROLLUPS [1] AND THEIR ARCHITECTURES.

A. Stand-alone Sequencer (with High Availability)

The most common and historically earliest deployment model for rollups is the stand-alone sequencer architecture, in which a single trusted operator is responsible for orchestrating the full lifecycle of L2 transactions. This design appears in two variants, namely simple and high-availability, reflecting different trade-offs between operational simplicity and resilience.

Simple stand-alone sequencer. The simplest instantiation of a rollup employs a single, centralized sequencer with direct RPC exposure and no internal redundancy. In this setup, the sequencer exclusively handles transaction ingestion, ordering, and execution, while also maintaining a persistent connection to an Ethereum node for posting transaction batches and corresponding state commitments to L1.

This simple architecture prioritizes minimal operational complexity, predictable behavior, and the lowest possible latency by avoiding any form of consensus or inter-operator coordination. However, these systems inherit a significant limitation: the sequencer represents a single point of failure. If it becomes unreachable (e.g., due to hardware failure, software bugs, DDoS attacks, or network partition), L2 production halts immediately. During downtime, users typically cannot force transaction inclusion or initiate withdrawals on L1 until the sequencer recovers. To partially mitigate this risk, some systems implement an escape hatch mechanism [20], which allows users to bypass the sequencer and submit certain critical transactions directly to Ethereum. Still, the recovery experience remains slow and disruptive, and the architecture offers limited resilience against targeted failures.

High-Availability (HA) stand-alone sequencer. To address these concerns while retaining the performance benefits of a centralized sequencer, some rollups deploy a high-availability variant of this architecture. Here, multiple sequencer replicas run in parallel, each deterministically executing the same transaction ordering logic and maintaining synchronized state. A dedicated availability coordinator designates one replica as the active sequencer responsible for ingesting user transactions, producing L2 blocks, and publishing L1 commitments, while the non-active replicas continuously stay in sync to enable a seamless failover if the active instance becomes unavailable. Figure 2a provides an overview of this design. To the user, this appears as a single RPC interface, which is typically fronted by a CDN or load balancer that routes incoming transactions to the active sequencer.

However, this redundancy introduces new operational risks absent in the simple model. Misconfigurations in the load balancer may route transactions to stale or incorrect replicas; compromised replicas or coordinators could collude to submit invalid state commitments; and the coordinator itself becomes a critical dependency whose failure can prevent replica selection or block production. Thus, while the high-availability architecture strengthens liveness, it broadens the trusted computing base and creates new attack surfaces that must be carefully managed.

B. Pipeline Sequencer

Pipeline sequencer architectures adopt a multi-stage design that decomposes the functionality of a monolithic sequencer into a set of specialized components. In this model, the end-to-end sequencing and proving workflow is organized as a pipeline consisting of the following stages. First, the sequencing node is responsible for transaction ordering and for producing a deterministic execution trace. A dedicated trace-generation module then executes the ordered transactions and outputs the corresponding trace, which may be optionally compressed by a Conflator. The resulting artifact is forwarded to a Coordinator, which schedules proving tasks to one or more provers, potentially arranged in a distributed or recursive configuration, to generate the final validity proof submitted to Ethereum. Figure 2b provides an overview of this workflow.

A key advantage of this design is that it decouples transaction latency from proof latency. Since the sequencer can publish a new state root immediately after completing execution, the system can offer near-instantaneous soft confirmations. Nevertheless, this architecture continues to exhibit network-level single points of failure, and it even worsens the situation since, instead of a single point of failure, this architecture presents several single point of failure components. Indeed, although functionality is decomposed, each L2 transaction must traverse every stage in the pipeline. Consequently, delays or failures in any intermediate component can propagate to downstream stages, creating bottlenecks and potentially causing cascading system-wide slowdowns.

C. Distributed Sequencer

The most recent generation of rollup designs explores distributed sequencer architectures, which replace the traditional single-operator sequencer (or its high-availability replication) with a set of independent sequencers that collectively order transactions. In these systems, sequencers are selected and coordinated through decentralized mechanisms, most commonly Byzantine fault-tolerant (BFT) consensus protocols [21], rotating-leader Proof-of-Stake (PoS) schemes [22], or auction-based selection. Rather than relying on a monolithic trusted operator, a committee of n sequencers participates in a protocol (e.g., Tendermint [23], PBFT [21]) to agree on transaction ordering, produce the canonical L2 block, and finalize the corresponding state transition. Figure 2c illustrates this architecture.

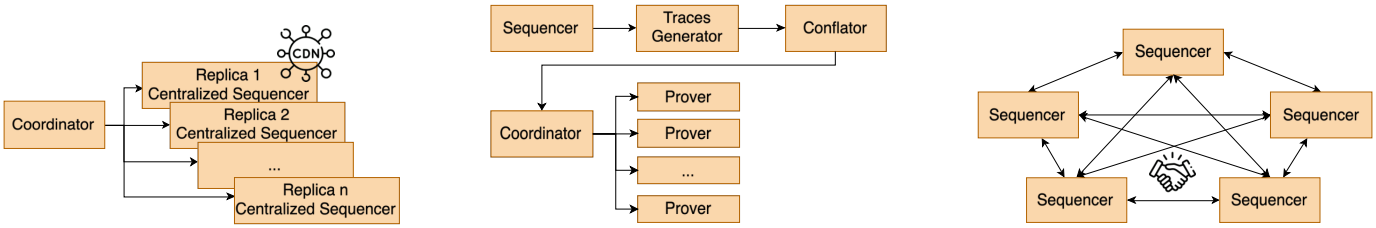


Figure 2. Three architectural designs of sequencers: (a) stand-alone (with high availability), (b) pipeline, (c) distributed.

The conceptual advantages of this design are substantial. First, censorship resistance is improved because as long as a threshold of sequencers (typically $\frac{2n}{3}$) remains honest, no single entity can unilaterally suppress or reorder user transactions. Second, fault tolerance and liveness become more robust, as progress depends on the collective behavior of a distributed committee rather than the availability of a single operator.

However, the design also introduces notable technical and economic challenges. Coordinating independent sequencers requires continuous consensus messaging, increasing communication overhead, and reducing throughput relative to centralized or pipeline-based designs. The multi-party environment also amplifies MEV-related risks, as competing sequencers may attempt to reorder transactions for profit. Systems must therefore adopt mitigations such as randomized leader election, encrypted mempools, or commit-reveal schemes. Furthermore, BFT-style consensus protocols remain vulnerable to liveness attacks, in which adversarial sequencers delay or obstruct the agreement process, temporarily halting L2 block production. Finally, the effectiveness of these architectures depends critically on incentive, e.g., robust mechanisms are needed to reward honest participation, deter equivocation or censorship, and enable slashing or exclusion of misbehaving sequencers.

IV. A Security Analysis of Rollups

Following our description of the rollup transaction workflow and its underlying architectural designs, we now turn to a systematic security analysis of rollup systems. In particular, we delineate the attack surfaces associated with each phase of transaction processing, namely before (§IV-A), during (§IV-B), and after (§IV-C) sequencer execution. For each attack surface, we examine concrete vulnerabilities and illustrate how both known and unexplored vectors may be leveraged.

A. Before Transaction Execution

Before user transactions reach the sequencer, they rely on off-chain pathways, primarily RPC relayers and Internet routing infrastructure, that constitute the pre-submission attack surface. We identify three key risks in this stage. First, public RPC relayers are the dominant interface through which users submit transactions, making them susceptible to a range of availability attacks, including volumetric DDoS, connection exhaustion, and application-layer abuse. Misconfigurations or software flaws in RPC gateways may further expose them to crashes or resource exhaustion. Because the sequencer depends on continuous transaction inflow, outages at the RPC layer can effectively halt user interaction with the rollup, even when the L2 backend remains fully operational.

Second, RPC operators can selectively censor or deprioritize transactions before they reach the sequencer. Such censorship may be explicit, motivated by regulatory constraints, address blacklisting, or compliance policies, or implicit, driven by MEV incentives. For example, an RPC provider could filter or delay transactions that conflict with the operator's own arbitrage or liquidation strategies. Since this filtering occurs upstream of the sequencer's mempool, it leaves no on-chain evidence, making such censorship difficult for users to detect.

Third, the communication path between users and RPC relayers traverses multiple network domains and is vulnerable to routing-layer and traffic-level attacks. Adversaries may perform BGP hijacks [24], route leaks, or targeted packet dropping to isolate certain users or regions selectively. State-level firewalls, ISP throttling, and local network interference can further disrupt or bias transaction propagation. These network-layer manipulations can systematically degrade access to the rollup or impose location-dependent censorship without compromising the relayer itself.

B. During Transaction Execution

Sequencer nodes constitute the most critical attack surface in any rollup system, as they are solely responsible for transaction ordering, execution, and producing state commitments. A compromised or stalled sequencer directly threatens liveness and, in some cases, short-term censorship resistance. The vulnerability profile differs significantly across architectural patterns (i.e., (HA) stand-alone, pipeline, and distributed), with increasing complexity generally widening the attack surface. We analyze the key risks across these architectural classes.

Attacks against stand-alone sequencers. In the simple stand-alone architecture, the sequencer represents a critical single point of failure. Key attack vectors include direct flooding, where adversaries overwhelm the sequencer with bursts of L2 transactions or malformed requests, saturating CPU, memory, or network resources. Because block production is tightly coupled to real-time execution, such resource exhaustion immediately halts L2 progress, preventing users from submitting transactions or initiating withdrawals until the operator manually restores service. Attackers may also submit

pathological or computationally expensive transactions (e.g., deeply nested smart contract executions) to exhaust sequencer validation resources. This depletes capacity, delaying or blocking legitimate transactions, and is particularly effective in systems with complex EVM semantics or limited transaction admission controls. Additionally, stand-alone sequencers must track L1 state, including forced transaction inclusions and escape hatch withdrawals [20]. By flooding Ethereum L1 with forced-inclusion transactions, adversaries can compel the sequencer to ingest, verify, and integrate these requests, delaying normal L2 processing and potentially degrading liveness.

HA stand-alone sequencer architectures replace a single sequencer node with multiple replicas coordinated through failover or leader-election mechanisms. While this design mitigates obvious single-node outages, it introduces additional attack surfaces targeting coordination, replication fidelity, and routing infrastructure. For instance, attackers may craft transaction sequences that exploit nondeterministic execution paths, race conditions, or timing variations across replicas. Even minor divergences in state roots or execution traces can trigger failover events or force replicas into resynchronization cycles, reducing throughput or stalling block production. The coordinator, which is often implemented via Redis, specialized databases, or custom leader-election services, represents another critical vulnerability. It can be targeted through connection exhaustion, state corruption, or network isolation. If the coordinator stalls, failover cannot occur, leaving the entire cluster without an active sequencer. Moreover, HA architectures rely on load balancers to distribute traffic evenly across replicas. Adversaries can exploit routing rules, improperly sanitized request headers, or edge-case behaviors to direct disproportionate traffic to weaker replicas. Such manipulations can overload certain nodes or induce state divergence, destabilizing the entire cluster.

Attacks against pipeline sequencers. Pipeline sequencer designs decompose transaction sequencing into multiple distributed components, including execution trace generators, conflators, batch coordinators, and prover networks. This modularization improves scalability and decouples execution from proof latency, but it also introduces inter-stage dependencies that attackers can exploit. Because pipeline sequencers rely on strict sequencing across interdependent services, failures in any stage can propagate downstream, amplifying the impact of attacks compared to monolithic designs. In particular, attackers can craft transactions with maximal computational complexity, placing disproportionate load on the trace generation component. Downstream components, such as conflators, coordinators, and provers, cannot proceed until trace outputs are available, causing delays that propagate through the entire pipeline and degrade overall system throughput. Network-layer attacks (e.g., BGP hijacks, targeted network interference) can also isolate critical pipeline components from either upstream sequencers or downstream provers. Such isolation halts batch formation and proof assignment, creating cascading delays that persist even when individual sequencer components remain operational. Misconfigured routing, load balancers, or internal APIs can further exacerbate these vulnerabilities by desynchronizing stages or overloading specific nodes. Moreover, pipeline architectures rely on distributed proving networks to generate cryptographic proofs for L1 verification. By submitting transactions that produce highly complex batches, attackers can saturate prover resources, delaying proof generation and finalization. In extreme cases, unproven batches may accumulate faster than the network can process, threatening system stability and extending L1 finality times.

Attacks against distributed sequencers. In distributed sequencer architectures, multiple independent nodes collectively determine transaction ordering and state transitions using consensus or sequencer selection protocols. While decentralization improves censorship resistance and reduces reliance on any single node, it introduces new attack surfaces targeting liveness, coordination, and economic incentives. For example, distributed sequencers typically rely on Byzantine fault-tolerant protocols (e.g., Tendermint [23], PBFT [21]) to agree on transaction ordering. Adversaries can delay, withhold, or reorder consensus messages to trigger repeated timeout rounds, slowing or stalling block production. Network-layer attacks can isolate subsets of nodes, preventing the formation of the required honest-majority quorum and effectively freezing the rollup until connectivity is restored. These attacks exploit the interdependence of nodes and amplify the impact of minor failures. If auction-based or PoS rotation mechanisms are used, attackers may acquire a disproportionate likelihood of being selected as sequencers. This enables targeted censorship of transactions or addresses and facilitates MEV extraction. Repeated influence over block proposals can compromise fairness and economic integrity, even when most sequencers act honestly. Also, some distributed designs rely on auxiliary services for leader election, task assignment, or state synchronization. These centralized or semi-centralized components can be targeted through connection exhaustion, state corruption, or misconfiguration. Compromising the coordinator stalls failover procedures and requires manual intervention, undermining the robustness benefits of a distributed architecture.

C. After Transaction Execution

Once the sequencer has executed transactions and produced a new state root, the rollup must post batch data to Ethereum L1 to guarantee data availability and enable independent verification. This data is published either as calldata or as blob space introduced in EIP-4844, forming the canonical record required for fraud proofs, validity proofs, and forced withdrawals [17]. At this stage, the attack surface shifts from L2-specific infrastructure to Ethereum's shared resource environment, which is outside the direct control of rollup operators. Although these attacks do not compromise correctness, they can substantially delay user withdrawals, extend finality times, and degrade the perceived responsiveness of the rollup. We highlight the following principal threats.

First, for rollups that rely on EIP-4844 blob storage, an adversary may flood the blob market with low-value or maliciously crafted blobs, consuming available capacity. This con-

gestion increases blob fees, delays data posting, and indirectly slows finality on the rollup. Since blob space is a shared global resource, attackers do not need to target a specific rollup to cause degradation effects.

Second, in the pipeline sequencer architecture where batch posting is delegated to a specialized component distinct from the transaction-executing sequencer, the batch poster becomes an additional point of failure. Targeted DoS attacks, resource exhaustion, or network isolation of the batch poster can prevent the timely submission of commitments to Ethereum, even when the sequencer operates correctly. Such decoupling improves scalability but enlarges the attack surface.

Third, for rollups using calldata, attackers can exploit Ethereum L1 fee dynamics by artificially inflating gas demand. By increasing calldata usage across unrelated applications or by submitting gas-intensive transactions, an adversary can drive up posting costs. Rollup operators may delay or batch more aggressively to reduce expenses, thereby increasing confirmation latency and extending the verification window for users awaiting finality or withdrawal.

V. Conclusion and Future Work

This work provides a systematic taxonomy of Ethereum Layer 2 rollups and analyzes the attack surfaces exposed throughout the transaction workflow. Our study demonstrates that architectural choices strongly influence both security and availability. Simple stand-alone sequencers offer operational simplicity and high throughput but are susceptible to single points of failure, resource exhaustion, and censorship. High-availability and pipeline designs mitigate some of these vulnerabilities, yet introduce new risks such as coordination failures, pipeline bottlenecks, and inter-component dependencies. Fully distributed sequencer architectures enhance censorship resistance and resilience under honest-majority assumptions but face practical challenges related to consensus liveness, network partitioning, and economic manipulation. Across all designs, the trade-offs between throughput, decentralization, and resilience are evident, highlighting that no single architecture optimally satisfies all requirements.

Several directions remain for advancing Ethereum rollup solutions that balance security, scalability, and decentralization. Formal verification of sequencer protocols and failover mechanisms could provide provable safety and liveness guarantees under adversarial conditions. Scalable Byzantine fault-tolerant consensus for distributed sequencers can reduce centralization risks while maintaining performance. Enhancing data availability, through redundancy, sampling, or tighter L1 integration, is critical for ensuring finality under network disruptions or DoS attacks. Protocol-level safeguards to limit MEV and prevent transaction manipulation are needed to preserve fairness and economic integrity. Finally, as Layer 2 ecosystems evolve, extending threat modeling to multi-rollup and cross-rollup interactions will be essential to secure interoperability and systemic resilience.

VI. Acknowledgments

This work was supported by the projects PID2021-125962OB-C33 SECURING/NET, PID2021-125962OB-C31 SECURING/CYBER, PID2024-156914OB-C43 SAFE-BLOCKCHAIN, and PID2024-156914OB-C41 SAFE-CYBER, funded by the Ministerio de Ciencia e Innovación, la Agencia Estatal de Investigación, and the European Regional Development Fund (ERDF); the ARTEMISA International Chair of Cybersecurity (C057/23) and the DANGER Strategic Project of Cybersecurity (C062/23), funded by the Spanish National Institute of Cybersecurity via the European Union – NextGenerationEU and the Recovery, Transformation and Resilience Plan; and the Catalan AGAUR grants SGR2021-00643 and SGR2021-01508. Additionally, A. Torralba-Agell is supported by AGAUR grant 2023 FI-100241 and a doctoral grant from the UOC.

References

[1] L2BEAT, "The state of the layer two ecosystem," https://l2beat.com/scaling/summary.
[2] Z. Spirkovski, "Layer 2s Now Handle 85% of All Transactions, Dune Report Says," 2025.
[3] Ethereum, "Optimistic Rollups," https://ethereum.org/developers/docs/scaling/optimistic-rollups/.
[4] ——, "Zero-knowledge rollups," https://ethereum.org/developers/docs/scaling/zk-rollups/.
[5] N. Grech, "Arbitrum Sequencer Outage | Root Cause Analysis," https://dedaub.com/blog/arbitrum-sequencer-outage/.
[6] Linea.eth, "Update on Velocore Incident," https://x.com/LineaBuild/status/1797283402745573837, 2024.
[7] ForkLog, "Velocore DEX Suffers $6.8 Million Loss in Exploit," https://forklog.com/en/velocore-dex-suffers-6-8-million-loss-in-exploit/, 2024.
[8] Z. Vardai, "Ethereum L2 Starknet suffers 2nd mainnet outage in 2 months," https://cointelegraph.com/news/starknet-outage-ethereum-l2-reliability-concerns, 2025.
[9] A. Torralba-Agell and C. Pérez-Solà, "A Comparison of Layer 2 Techniques for Scaling Blockchains," in *Proc. RECSI*, 2022.
[10] M. Neuder and A. Stokes, "Rollup-Centric Roadmap," https://notes.ethereum.org/@mikeneuder/rcr2vmsvftv, 2025.
[11] A. Torralba-Agell and C. Pérez-Solà, "Security Assumptions in Permissionless Blockchains and Layer 2 Scalability Solutions: A Comprehensive Taxonomy," *Blockchain: Research and Applications*, 2025.
[12] "Alchemy," https://www.alchemy.com/.
[13] "Infura," https://www.infura.io/.
[14] "ChainList," https://chainlist.org/.
[15] Ethereum, "Blockchain Data Storage Strategies," https://ethereum.org/developers/docs/data-availability/blockchain-data-storage-strategies/.
[16] V. Buterin, D. Feist, D. Loerakker, G. Kadianakis, M. Garnett, M. Taiwo, and A. Dietrichs, "EIP-4844: Shard Blob Transactions," https://eips.ethereum.org/EIPS/eip-4844.
[17] J. Neu, "Data Availability Sampling: From Basics to Open Problems," https://www.paradigm.xyz/2022/08/das, Aug. 2022.
[18] J. Groth, "On the size of pairing-based non-interactive arguments," in *Annual international conference on the theory and applications of cryptographic techniques*. Springer, 2016, pp. 305–326.
[19] E. Ben-Sasson, I. Bentov, Y. Horesh, and M. Riabzev, "Scalable, transparent, and post-quantum secure computational integrity," *Cryptology ePrint Archive*, 2018.
[20] F. G. Figueira, M. Derka, C. L. Chiu, and J. Gorzny, "A practical rollup escape hatch design," *arXiv*, 2025.
[21] M. Castro and B. Liskov, "Practical byzantine fault tolerance and proactive recovery," *ACM TOCS*, 2002.
[22] U. W. Chohan, "Proof-of-stake algorithmic methods: a comparative summary," *SSRN 3131897*, 2018.
[23] E. Buchman, "Tendermint: Byzantine Fault Tolerance in the Age of Blockchains," *PhD Thesis*, 2016.
[24] S. Cho, R. Fontugne, K. Cho, A. Dainotti, and P. Gill, "BGP hijacking classification," in *Proc. IEEE TMA*, 2019.

Secuencias binomiales sobre cuerpos primos

Miguel Beltrá
Universidad de Alicante
miguel.beltra@ua.es

Sara D. Cardell
Universidade Estadual Paulista
sd.cardell@unesp.br

Verónica Requena
Universidad de Alicante
vrequena@ua.es

***Abstract*—En este trabajo estudiamos las sucesiones binomiales p-arias para un primo p, sus características intrínsecas y sus reglas de formación. También demostramos que la familia de sucesiones p-arias con periodo una potencia de p forma un espacio vectorial sobre $\mathbb{F}_p$, y que la familia de sucesiones binomiales p-arias es una base de dicho espacio.**

***Index Terms*—Secuencias binomiales, cuerpos finitos, secuencias p-arias, matriz de Hadamard, producto de Kronecker.**

I. Introducción

Las sucesiones de recurrencia lineal sobre cuerpos finitos desempeñan un papel fundamental en teoría de códigos, criptografía y generación de sucesiones pseudoaleatorias, principalmente debido a su comportamiento periódico bien caracterizado. Los resultados clásicos proporcionan una descripción detallada de los polinomios característicos que generan sucesiones de periodo máximo, así como de la descomposición de las sucesiones lineales recurrentes en componentes periódicos y nulos. Investigaciones más recientes han ampliado este enfoque analizando polinomios característicos que admiten un conjunto restringido de posibles periodos y extendiendo el estudio de las sucesiones recurrentes a cuerpos no binarios, una línea motivada en parte por el renovado interés en la lógica multivaluada y en arquitecturas hardware no binarias.

En este artículo nos centramos en una familia estructuralmente rica de sucesiones: las sucesiones binomiales, definidas como las diagonales del triángulo de Pascal. Aunque sus contrapartes binarias han sido investigadas de forma exhaustiva —particularmente con respecto a su periodicidad, complejidad lineal y reglas recursivas de formación— el comportamiento de estas sucesiones sobre cuerpos primos sigue sin explorarse. Nuestro trabajo aborda este vacío desarrollando una teoría completa de las sucesiones binomiales p-arias. Establecemos sus propiedades algebraicas y periódicas fundamentales, determinamos sus polinomios característicos y complejidades lineales, y demostramos que forman una base del espacio de sucesiones p-arias cuyo periodo es una potencia de p. Estos resultados ofrecen nuevas perspectivas sobre la estructura de las sucesiones sobre cuerpos finitos y abren vías prometedoras para aplicaciones en criptografía, teoría de códigos y diseño de sucesiones.

II. Preliminares

Sea $\mathbb{F}_p$ el cuerpo de Galois de p elementos. Decimos que $\{s_n\}_{n\geq 0} = \{s_0, s_1, s_2, \ldots\}$ es una **sucesión p-aria** si $s_n \in \mathbb{F}_p$, para $n = 0, 1, 2, \ldots$ En lo que sigue, todas las sucesiones consideradas serán p-arias y escribiremos $\{s_n\}$ en lugar de $\{s_n\}_{n\geq 0}$.

Denotamos por $V(\mathbb{F}_p)$ al conjunto de todas las sucesiones p-arias. Junto con las operaciones dadas por $\{s_n\} + \{\sigma_n\} = \{s_n + \sigma_n\}$ y $c\{s_n\} = \{c\, s_n\}$, con $c \in \mathbb{F}_p$, el conjunto $V(\mathbb{F}_p)$ forma un espacio lineal sobre $\mathbb{F}_p$, siendo la sucesión cero $\{000\ldots\}$ el elemento neutro para la suma [1].

Una sucesión $\{s_n\}$ se dice **periódica** si existe un entero N tal que $s_{n+N} = s_n$ para todo $n \geq 0$. El entero positivo mínimo T tal que $s_{n+T} = s_n\ \forall n \geq 0$ se denomina **periodo** de la sucesión.

Sea r un entero positivo y sean $d_1, d_2, d_3, \ldots, d_r$ coeficientes constantes con $d_i \in \mathbb{F}_p$. Una sucesión p-aria $\{s_n\}$ que satisface la relación

$$s_{n+r} = d_r s_n + d_{r-1} s_{n+1} + \cdots + d_1 s_{n+r-1}, \quad (1)$$

se denomina una **sucesión lineal recurrente** (de r-**ésimo orden**) en $\mathbb{F}_p$. Los términos $\{s_0, s_1, \ldots, s_{r-1}\}$ se conocen como el **estado inicial** y determinan de manera única el resto de la sucesión.

Llamamos **operador de desplazamiento** a la aplicación $\mathbf{E} : V(\mathbb{F}_p) \to V(\mathbb{F}_p)$ dada por

$$\mathbf{E}\{s_0, s_1, s_2, \ldots\} = \{s_1, s_2, s_3, \ldots\}.$$

$\mathbf{E}$ es un operador lineal sobre $V(\mathbb{F}_p)$. Denotemos por E la acción de $\mathbf{E}$ sobre cada término de la sucesión, esto es, $Es_n = s_{n+1}$ y por lo tanto,

$$\mathbf{E}\{s_0, s_1, s_2, \ldots\} = \{Es_0, Es_1, Es_2, \ldots\}.$$

Si $E^t s_n = s_{n+t}$, podemos escribir (1) como

$$E^r s_n = d_r s_n + d_{r-1} E s_n + \cdots + d_1 E^{r-1} s_n.$$

Sea $p(x)$ el polinomio mónico dado por

$$p(x) = x^r - \sum_{i=1}^{r} d_i x^{r-i} \in \mathbb{F}_p[x].$$

Este polinomio satisface

$$p(E)s_n = 0, \quad n \geq 0,$$

y se denomina **polinomio característico** de $\{s_n\}$. La **complejidad lineal**, LC, de una sucesión $\{s_n\}$ se define como el orden de la relación de recurrencia lineal de menor orden que genera dicha sucesión (el grado del polinomio característico).

Definición 1. La **matriz de Pascal-Hadamard** de tamaño $p^L \times p^L$ módulo p es la matriz

$$H_L = \begin{bmatrix} \binom{0}{0} & \binom{1}{0} & \binom{2}{0} & \cdots & \binom{p^L-1}{0} \\ \binom{0}{1} & \binom{1}{1} & \binom{2}{1} & \cdots & \binom{p^L-1}{1} \\ \binom{0}{2} & \binom{1}{2} & \binom{2}{2} & \cdots & \binom{p^L-1}{2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \binom{0}{p^L-1} & \binom{1}{p^L-1} & \binom{2}{p^L-1} & \cdots & \binom{p^L-1}{p^L-1} \end{bmatrix}$$

donde cada entrada está reducida módulo p.

Definición 2. Sea $A = [a_{ij}]$ una matriz de tamaño $m \times n$ y $B = [b_{kl}]$ una matriz de tamaño $p \times q$. El producto de Kronecker de A y B se define como la matriz

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix},$$

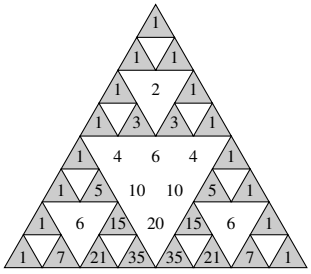
de tamaño $mp \times nq$.

Lema 1. *La matriz de Pascal-Hadamard satisface las siguientes propiedades:*

a) $H_L = H_1 \otimes H_{L-1}$.
b) H_L^{-1} *es la traspuesta de* H_L *respecto de su antidiagonal.*

III. Secuencias Binomiales p-arias

Fig. 1: Triángulo de Pascal



Los coeficientes binomiales módulo p nos permiten introducir las secuencias binomiales p-arias.

Definición 3. Dado un entero fijo $k \geq 0$, llamamos **k-ésima secuencia binomial p-aria** a la secuencia $\{s_n\}$ dada por:

$$s_n = \begin{cases} 0 & \text{si } n < k, \\ \binom{n}{k} \bmod p & \text{si } n \geq k. \end{cases}$$

Denotaremos a esta secuencia como $\left\{\binom{n}{k}\right\}$.

En [2], las secuencias binomiales binarias fueron estudiadas en profundidad, demostrando que las diagonales del triángulo de Sierpinski (el triángulo de Pascal módulo 2) corresponden a versiones desplazadas de estas secuencias (véase la Figura 1). Además, podemos observar que las filas de la matriz de Pascal–Hadamard son las secuencias binomiales.

En este trabajo estudiamos las secuencias binomiales sobre $\mathbb{F}_p$, con $p > 2$ un número primo.

A. Secuencias sobre $\mathbb{F}_3$

En esta sección estudiamos las secuencias binomiales 3-arias sobre $\mathbb{F}_3$ con el fin de ilustrar la idea del caso general $\mathbb{F}_p$. La Tabla I muestra las primeras 27 secuencias binomiales 3-arias.

Teorema 2. *Sea k un entero positivo y sean L, r enteros no negativos. El periodo de $\left\{\binom{n}{k}\right\}$, con $k = 3^L + r$ y $0 \leq r < 2 \cdot 3^L$, es $T = 3^{L+1}$. Además, las secuencias tienen la siguiente forma:*

1) *Para los múltiplos de potencias de 3:*

$$\left\{\binom{n}{3^L}\right\} = \left\{ \underbrace{00\ldots00}_{3^L}, \underbrace{11\ldots11}_{3^L}, \underbrace{22\ldots22}_{3^L} \right\},$$

$$\left\{\binom{n}{2\cdot 3^L}\right\} = \left\{ \underbrace{00\ldots00}_{2\cdot 3^L}, \underbrace{11\ldots11}_{3^L} \right\}.$$

2) *Para $i = 1, 2, \ldots, L$ y $3^{i-1} \leq r < 3^i$:*

$$\left\{\binom{n}{3^L + r}\right\} = \left\{ \underbrace{00\ldots00}_{3^L}, \underbrace{\left\{\binom{n}{r}\right\}}_{3^{L-i}}, \underbrace{2\left\{\binom{n}{r}\right\}}_{3^{L-i}} \right\},$$

$$\left\{\binom{n}{2\cdot 3^L + r}\right\} = \left\{ \underbrace{00\ldots00}_{2\cdot 3^L}, \underbrace{\left\{\binom{n}{r}\right\}}_{3^{L-i}} \right\}.$$

$\underbrace{\left\{\binom{n}{r}\right\}}_{3^{L-i}}$ *indica que* $\left\{\binom{n}{r}\right\}$ *se repite* 3^{L-i} *veces.*

B. Secuencias sobre $\mathbb{F}_p$

En esta sección estudiamos las propiedades de las secuencias binomiales sobre $\mathbb{F}_p$.

Teorema 3. *El periodo de la secuencia $\left\{\binom{n}{p^L+k}\right\}$, donde $0 \leq k < p^{L+1} - 1$, es $T = p^{L+1}$.*

El siguiente resultado proporciona una forma de obtener secuencias binomiales directamente. La

TABLE I: Primeras secuencias binomiales sobre $\mathbb{F}_3$

Secuencia binomial	Primeros términos
$\left\{\binom{n}{0}\right\}$	11111111111111111111111111111111
$\left\{\binom{n}{1}\right\}$	01201201201201201201201201201201
$\left\{\binom{n}{2}\right\}$	00100100100100100100100100100100
$\left\{\binom{n}{3}\right\}$	00011122200011122200011122200011122
$\left\{\binom{n}{4}\right\}$	00001202100001202100001202100001202
$\left\{\binom{n}{5}\right\}$	00000102200000102200000102200000102
$\left\{\binom{n}{6}\right\}$	00000011100000011100000011100000011
$\left\{\binom{n}{7}\right\}$	00000001200000001200000001200000001
$\left\{\binom{n}{8}\right\}$	00000000100000000100000000100000001
$\left\{\binom{n}{9}\right\}$	00000000011111111122222222011111111
$\left\{\binom{n}{10}\right\}$	00000000001201201202102102100000000
$\left\{\binom{n}{11}\right\}$	00000000000100100100200200200000000
$\left\{\binom{n}{12}\right\}$	00000000000011122200022211100000000
$\left\{\binom{n}{13}\right\}$	00000000000001202100002101200000000
$\left\{\binom{n}{14}\right\}$	00000000000000100200000200100000000
$\left\{\binom{n}{15}\right\}$	00000000000000011100000022200000000
$\left\{\binom{n}{16}\right\}$	00000000000000001200000002100000000
$\left\{\binom{n}{17}\right\}$	00000000000000000100000000200000000
$\left\{\binom{n}{18}\right\}$	00000000000000000011111111100000000
$\left\{\binom{n}{19}\right\}$	00000000000000000001201201200000000
$\left\{\binom{n}{20}\right\}$	00000000000000000000100100100000000
$\left\{\binom{n}{21}\right\}$	00000000000000000000011122200000000
$\left\{\binom{n}{22}\right\}$	00000000000000000000001202100000000
$\left\{\binom{n}{23}\right\}$	00000000000000000000000102200000000
$\left\{\binom{n}{24}\right\}$	00000000000000000000000011100000000
$\left\{\binom{n}{25}\right\}$	00000000000000000000000001200000000
$\left\{\binom{n}{26}\right\}$	00000000000000000000000000100000000

demostración se omite, ya que en la Sección IV introducimos un algoritmo alternativo para construir estas secuencias y donde probaremos su estructura.

Teorema 4. *Sea k un entero positivo y sean L, r enteros no negativos. Para $j \in \{1, 2, \ldots, p-1\}$, consideremos $\left\{\binom{n}{j}\right\} = \{s_0^{(j)}, s_1^{(j)}, \ldots, s_{p-1}^{(j)}\}$. Las secuencias $\left\{\binom{n}{k}\right\}$, donde $k = p^L + r$ y $0 \leq r < (p-1) \cdot p^L$, tienen las siguientes formas:*

1) *Para los múltiplos de potencias de p:*

$$\left\{\binom{n}{j \cdot p^L}\right\} = \left\{\underbrace{s_0^{(j)} \ldots s_0^{(j)}}_{p^L}, \ldots, \underbrace{s_{p-1}^{(j)} \ldots s_{p-1}^{(j)}}_{p^L}\right\},$$

para cualquier $j \in \{1, 2, \ldots, p-1\}$.

2) *Para $i = 1, 2, \ldots, L$ y $p^{i-1} \leq r < p^i$:*

$$\left\{\binom{n}{j \cdot p^L + r}\right\} = \left\{\underbrace{s_0^{(j)} \cdot \left\{\binom{n}{r}\right\}}_{p^{L-i}}, \ldots, \underbrace{s_{p-1}^{(j)} \cdot \left\{\binom{n}{r}\right\}}_{p^{L-i}}\right\},$$

para cualquier $j \in \{1, 2, \ldots, p-1\}$.

Como consecuencia del teorema anterior, obtenemos el siguiente resultado.

Corolario 5. *El periodo de $\left\{\binom{n}{p^L}\right\}$ es $T = p^{L+1}$ y su forma es*

$$\left\{\binom{n}{p^L}\right\} = \left\{\underbrace{0 \ldots 0}_{p^L}, \underbrace{1 \ldots 1}_{p^L}, \ldots, \underbrace{p-1 \ldots p-1}_{p^L}\right\}.$$

Ya estamos preparados para estudiar el polinomio característico de estas secuencias. Primero, necesitamos algunos resultados preliminares.

Lema 6. *[2, Lema 8]. Dados dos enteros positivos n y k, con $n > k \geq 1$, se tiene que*

$$\binom{n+1}{k} = \binom{n}{k} + \binom{n}{k-1}.$$

Este lema nos proporciona una definición concreta del operador $\mathbf{E}$ sobre las secuencias binomiales: la acción de $\mathbf{E}$ sobre el n-ésimo término de la k-ésima secuencia binomial viene dada por

$$E\binom{n}{k} = \binom{n+1}{k} = \binom{n}{k} + \binom{n}{k-1}.$$

Además, aplicando repetidamente dicha acción $\mathbf{E}^r = \underbrace{\mathbf{E} \circ \cdots \circ \mathbf{E}}_{r \text{ veces}}$ sobre el n-ésimo término de la k-ésima secuencia binomial es

$$E^r\binom{n}{k} = \binom{n}{k} + \sum_{i=0}^{r-1} \binom{n+i}{k-1}. \tag{2}$$

Proposición 7. *Para cualquier $k \geq 1$, la secuencia binomial $\left\{\binom{n}{k-1}\right\}$ coincide con $\left\{\binom{n+1}{k}\right\} - \left\{\binom{n}{k}\right\}$.*

El siguiente lema generaliza [3, Lema 2].

Lema 8. *Si $\{s_n\}$ y $\{u_n\}$ son secuencias p-arias tales que $u_n = s_{n+1} - s_n$. Entonces, $q(x) \in \mathbb{F}_p[x]$ genera $\{u_n\}$ si y solo si $(x-1)q(x)$ genera $\{s_n\}$.*

El siguiente teorema generaliza [3, Theorem 1].

Teorema 9. *Sea $\{s_n\}$ una secuencia p-aria y sea $\{u_n\}$ la secuencia p-aria definida por $u_n = s_{n+1} - s_n \ \forall n \geq 0$. Si $(x-1)m(x)$ es el polinomio característico de $\{s_n\}$, entonces $m(x)$ es el polinomio característico de $\{u_n\}$.*

Demostración. Supongamos que $(x-1)m(x)$ es el polinomio característico de $\{s_n\}$. Si $m'(x)$ es un polinomio con $\deg(m'(x)) < \deg(m(x))$ que genera $\{u_n\}$, entonces por el Lema 8 $(x-1)m'(x)$ genera $\{s_n\}$, con $\deg((x-1)m'(x)) < \deg((x-1)m(x))$, lo cual contradice que $(x-1)m(x)$ sea característico. ■

El recíproco del teorema anterior es cierto.

Teorema 10. *Sea $\{s_n\}$ una secuencia p-aria y $\{u_n\}$ la secuencia definida por $u_n = s_{n+1} - s_n \ \forall n \geq 0$. Sea $k \geq 1$. Entonces, $(x-1)^{k+1}$ es el polinomio característico de $\{s_n\}$ si y solo si $(x-1)^k$ es el polinomio característico de $\{u_n\}$.*

Demostración. La implicación directa se obtiene del Teorema 9. Supongamos ahora que $(x-1)^k$ es el polinomio característico de $\{u_n\}$. Por el Lema 8, $(x-1)^{k+1}$ genera $\{s_n\}$.

Como $\mathbb{F}_p[x]$ es un dominio de factorización única, el polinomio característico tiene la forma $(x-1)^t$ para algún $1 \leq t \leq k+1$. Si $t < k+1$, por el Lema 8, $(x-1)^{t-1}$ genera $\{u_n\}$ con $t-1 < k$,

que contradice que $(x-1)^k$ es el polinomio característico de $\{u_n\}$. Necesariamente $t = k+1$. ■

Como consecuencia de la proposición 7 y el teorema 10, podemos obtener la LC de las secuencias binomiales.

Teorema 11. *El polinomio característico de* $\left\{\binom{n}{k}\right\}$ *es* $(x-1)^{k+1}$. *Por tanto, su LC es* $k+1$.

Por el teorema anterior, tenemos lo siguiente.

Corolario 12. *Dada una secuencia binomial* $\sum_{i=1}^{t} c_i\left\{\binom{n}{k_i}\right\}$, *donde* $k_1 < k_2 < \cdots < k_t$ *son índices enteros y* $c_i \neq 0$ *para todo* $i = 1, 2, \ldots, t$, *entonces el polinomio característico de dicha secuencia es* $(x-1)^{k_t+1}$ *y su LC es* $k_t + 1$.

Demostración. Por el Teorema 11, $(x-1)^{k_t+1}$ es un polinomio generador de $\left\{\binom{n}{k_i}\right\}$ para todo $i = 1, 2, \ldots, t$. Por tanto, tenemos

$$\begin{aligned}
&(E-\mathrm{id})^{k_t+1}\sum_{i=1}^{t} c_i\binom{n}{k_i}\\
&=\sum_{i=1}^{t} c_i(E-\mathrm{id})^{k_t-k_i}(E-\mathrm{id})^{k_i+1}\binom{n}{k_i}\\
&=\sum_{i=1}^{t} c_i(E-\mathrm{id})^{k_t-k_i}\,0 = 0, \quad n \geq 0,
\end{aligned}$$

por lo que genera $\sum_{i=1}^{t} c_i\left\{\binom{n}{k_i}\right\}$. Ahora veamos que $(x-1)^{k_t}$ no genera esa secuencia:

$$\begin{aligned}
&(E-\mathrm{id})^{k_t}\sum_{i=1}^{t} c_i\binom{n}{k_i}\\
&=\sum_{i=1}^{t-1} c_i(E-\mathrm{id})^{k_t-k_i-1}(E-\mathrm{id})^{k_i+1}\binom{n}{k_i}\\
&\quad + c_t(E-\mathrm{id})^{k_t}\binom{n}{k_t}\\
&= c_t(E-\mathrm{id})^{k_t}\binom{n}{k_t} \neq 0, \quad n \geq 0,
\end{aligned}$$

ya que $(x-1)^{k_t}$ no genera $\left\{\binom{n}{k_t}\right\}$. Por tanto, $(x-1)^{k_t}$ no es un polinomio generador. Necesariamente, $(x-1)^{k_t+1}$ es el polinomio característico de $\sum_{i=1}^{t} c_i\left\{\binom{n}{k_i}\right\}$. ■

A continuación mostramos que toda secuencia en $\mathbb{F}_p$ de periodo p^L, para algún entero $L \geq 0$, puede escribirse como combinación lineal de secuencias binomiales de periodo p^ℓ, con $\ell \leq L$. Además, probamos que el conjunto de secuencias de periodo p^ℓ, con $\ell \leq L$, es un espacio lineal y que el conjunto de secuencias binomiales de periodo p^ℓ, con $\ell \leq L$, es una base de dicho espacio.

Teorema 13. *Toda secuencia de periodo* p^L *puede escribirse como combinación lineal de secuencias binomiales de periodo* p^ℓ, *con* $\ell \leq L$.

Demostración. Sea $\{s_n\}$ una secuencia arbitraria de periodo p^L. Como $\{s_n\}$ es periódica de periodo p^L, basta encontrar los coeficientes $c_0, c_1, c_2, \ldots, c_{p^L-1} \in \mathbb{F}_p$ tales que

$$\begin{aligned}
\{s_n\}_{n=0}^{p^L-1} &= c_0\left\{\binom{n}{0}\right\}_{n=0}^{p^L-1} + c_1\left\{\binom{n}{1}\right\}_{n=0}^{p^L-1} \\
&\quad + c_2\left\{\binom{n}{2}\right\}_{n=0}^{p^L-1} + \cdots + c_{p^L-1}\left\{\binom{n}{p^L-1}\right\}_{n=0}^{p^L-1}.
\end{aligned} \tag{3}$$

En forma matricial, esto equivale a

$$\begin{aligned}
&\begin{bmatrix} s_0 & s_1 & s_2 & \ldots & s_{p^L-1} \end{bmatrix}\\
&\qquad = \begin{bmatrix} c_0 & c_1 & c_2 & \ldots & c_{p^L-1} \end{bmatrix} H_L.
\end{aligned}$$

De acuerdo con el Lema 1, la matriz H_L es no singular. Por tanto, este sistema lineal tiene una única solución en $\mathbb{F}_p$, dada por:

$$\begin{aligned}
&\begin{bmatrix} c_0 & c_1 & c_2 & \ldots & c_{p^L-1} \end{bmatrix}\\
&\qquad = \begin{bmatrix} s_0 & s_1 & s_2 & \ldots & s_{p^L-1} \end{bmatrix} H_L^{-1}.
\end{aligned} \tag{4}$$

Luego, existen coeficientes $c_0, \ldots, c_{p^L-1}$ tales que la expresión en (3) es válida. ■

El siguiente corolario muestra la estructura del conjunto de secuencias de periodo p^ℓ, con $\ell \leq L$.

Corolario 14. *Sea* W *el conjunto de secuencias* p*-arias de periodo* p^ℓ *con* $\ell \leq L$. *Entonces* W *es un subespacio lineal de* $V(\mathbb{F}_p)$ *de dimensión* p^L *y* $\mathcal{B} = \left\{\left\{\binom{n}{k}\right\} \mid 0 \leq k < p^L\right\}$ *es una base de* W.

Además, como consecuencia del Teorema 13 y del Corolario 12, deducimos el siguiente resultado.

Teorema 15. *Toda secuencia* p*-aria de periodo* p^L *tiene como polinomio característico a* $(x-1)^\ell$ *y* $LC = \ell$, *con* $p^{L-1} < \ell \leq p^L$.

Definición 4. La representación $\sum_{i=0}^{LC-1} c_i\left\{\binom{n}{i}\right\}$ se conoce como **representación binomial** de la secuencia.

Como consecuencia de la demostración del Teorema 13, podemos recuperar la representación binomial a partir de la secuencia, y viceversa, usando la matriz de Pascal–Hadamard (expresión (4)).

Ejemplo 1. Consideremos $\{012122202\ldots\}$ de periodo $T = 9$ en $\mathbb{F}_3$. Usaremos la matriz H_2 para calcular su representación binomial:

$$\begin{aligned}
\{s_n\}\cdot H_2 &= \begin{bmatrix} 0&1&2&1&2&2&2&0&2 \end{bmatrix} \cdot \begin{bmatrix} 1&2&1&2&1&2&1&2&1\\ 0&1&1&0&2&2&0&1&1\\ 0&0&1&0&0&2&0&0&1\\ 0&0&0&1&2&1&1&2&1\\ 0&0&0&0&1&1&0&1&1\\ 0&0&0&0&0&1&0&0&1\\ 0&0&0&0&0&0&1&2&1\\ 0&0&0&0&0&0&0&1&1\\ 0&0&0&0&0&0&0&0&1 \end{bmatrix}\\
&= \begin{bmatrix} 0&1&0&1&0&2&0&0&0 \end{bmatrix}.
\end{aligned}$$

Por tanto, esta secuencia puede calcularse como:

$$\begin{aligned}
&\left\{\binom{n}{1}\right\} + \left\{\binom{n}{3}\right\} + 2\cdot\left\{\binom{n}{5}\right\}\\
&= \{012012012\} + \{000111222\} + 2\{000001002\}\\
&= \{012122202\},
\end{aligned}$$

y su rep. binomial es $\left\{\binom{n}{1}\right\} + \left\{\binom{n}{3}\right\} + 2\cdot\left\{\binom{n}{5}\right\}$.

El siguiente teorema nos proporciona un método para obtener la representación binomial de los desplazamientos de una secuencia. Su demostración es una consecuencia directa del Lema 6.

Teorema 16. *Sea* $\sum_{i=0}^{LC-1} c_i\left\{\binom{n}{i}\right\}$, $c_i \in \mathbb{F}_p$, *la representación binomial de* $\{s_n\}$. *Si desplazamos cíclicamente los términos de* $\{s_n\}$ *una posición a la izquierda, entonces su rep. binomial es:*

$$\sum_{i=1}^{LC-1} c_i\left[\left\{\binom{n}{i}+\binom{n}{i-1}\right\}\right]+c_0\left\{\binom{n}{0}\right\}.$$

El siguiente resultado nos proporciona la matriz del operador de desplazamiento para las secuencias de periodo p^ℓ con $\ell \leq L$. La demostración es una consecuencia directa del Teorema 16.

Corolario 17. *Sea* W *el espacio lineal de secuencias* p*-arias de periodo* p^ℓ *con* $\ell \leq L$, *y sea* $\boldsymbol{E}|_W : W \to W$ *el operador de desplazamiento en* W. *La matriz asociada a* $\boldsymbol{E}|_W$ *con respecto a la base* $\mathcal{B} = \left\{\left\{\binom{n}{k}\right\} \mid k = 0, 1, \ldots, p^L - 1\right\}$ *es*

$$M_{\mathcal{B}}(\boldsymbol{E}|_W) = \begin{bmatrix} 1 & 1 & & & \\ & 1 & 1 & & \\ & & \ddots & \ddots & \\ & & & 1 & 1 \\ & & & & 1 \end{bmatrix} \in \mathbb{F}_p^{p^L \times p^L}.$$

Resulta que la matriz $M_{\mathcal{B}}(\mathbf{E}|_W)$ presenta una fuerte conexión con los coeficientes binomiales, relación que se hace evidente al calcular sus potencias como se muestra en el corolario siguiente.

Corolario 18. *Si* $r \in \{0, 1, \ldots, p^L - 1\}$, *entonces*

$$M_{\mathcal{B}}(\boldsymbol{E}^r|_W) = \begin{bmatrix} \binom{r}{0} & \binom{r}{1} & \cdots & \binom{r}{r-1} & \binom{r}{r} & & & \\ & \ddots & \ddots & & & \ddots & & \\ & & \binom{r}{0} & \binom{r}{1} & \binom{r}{2} & \cdots & \binom{r}{r} & \\ & & & \ddots & \ddots & \ddots & \vdots & \\ & & & & \binom{r}{0} & \binom{r}{1} & \binom{r}{2} & \\ & & & & & \binom{r}{0} & \binom{r}{1} & \\ & & & & & & \binom{r}{0} & \end{bmatrix},$$

y, además, $M_{\mathcal{B}}(\boldsymbol{E}^{p^L}|_W) = I_{p^L} \in \mathbb{F}_p^{p^L \times p^L}$.

La matriz del Corolario 18 es conocida en la literatura y ha aparecido en otros contextos, como en Teoría de Códigos (véase, por ejemplo, [4]).

Teorema 19. *Dado un número primo* p *y un entero* $k \geq 1$, *se tiene que para todo* n *se cumple:*

$$\begin{aligned}\binom{n+r}{k} &\equiv \binom{n}{k} + \sum_{j=0}^{r-1} \binom{n+j}{k-1} \\ &\equiv \sum_{j=0}^{\min\{r,k\}} \binom{r}{j}\binom{n}{k-j} \mod p.\end{aligned}$$

Como corolario del teorema anterior, deducimos cómo actúa el operador de desplazamiento sobre una secuencia de periodo p^L.

Corolario 20. *Sea* $\{s_n\}$ *una secuencia* p*-aria periódica de periodo* p^L *con representación binomial* $\sum_{i=0}^{LC-1} c_i\left\{\binom{n}{i}\right\}$. *Entonces*

$$\begin{aligned}\{s_{n+r}\} \equiv \boldsymbol{E}^r|_W(\{s_n\}) &\equiv \sum_{i=0}^{LC-1} c_i\left\{\binom{n+r}{i}\right\} \pmod p \\ &\equiv \sum_{i=0}^{LC-1} c_i\left\{\sum_{j=0}^{\min\{r,i\}} \binom{r}{j}\binom{n}{i-j}\right\} \pmod p.\end{aligned}$$

IV. Descomposición de las secuencias binomiales

En esta sección mostramos que todas las secuencias binomiales módulo p pueden obtenerse mediante el producto de Hadamard de dos secuencias y la expansión de una secuencia.

Definición 5. Sean $\{a_n\}$ y $\{b_n\}$ dos secuencias p-arias. Definimos su **producto de Hadamard** (**producto componente a componente**) como

$$\{a_n\} \odot \{b_n\} = \{a_n \cdot b_n\}.$$

Para evitar el uso de múltiples paréntesis, consideramos que el producto de Kronecker tiene prioridad sobre el producto de Hadamard.

Definición 6. Dado un número primo p, definimos el **bloque básico de expansión** p**-aria** B_p como

$$B_p = \underbrace{\{11\ldots1\}}_{p \text{ términos}}.$$

Además, para $\ell \geq 1$, tenemos que

$$B_{p^\ell} = B_{p^{\ell-1}} \otimes B_p = \underbrace{\{11\ldots1\}}_{p^\ell \text{ términos}}.$$

En el siguiente lema mostramos que: la expansión p-repetida de una secuencia binomial módulo p es otra secuencia binomial módulo p; y en qué casos el producto de Hadamard de dos secuencias binomiales es otra secuencia binomial.

Lema 21. *Sea* p *un número primo, y* k, ℓ *enteros positivos, y* $j \in \{0, 1, \ldots, p^\ell - 1\}$. *Entonces:*

a) $\left\{\binom{n}{kp}\right\} \equiv \left\{\binom{n}{k}\right\} \otimes B_p \pmod p$.

b) $\left\{\binom{n}{mp^\ell+j}\right\} = \left\{\binom{n}{mp^\ell}\right\} \odot \left\{\binom{n}{j}\right\}$.

El siguiente teorema muestra que cualquier secuencia binomial puede construirse a partir de las primeras p secuencias binomiales $\left\{\binom{n}{t}\right\}$, para $0 \leq t < p$.

Teorema 22. *Sea* $k = k_\ell p^\ell + \cdots + k_1 p + k_0$ *la expansión* p*-aria de* k, $k_i \in \{0, 1, \ldots, p-1\}$. *La* k*-ésima secuencia binomial módulo* p *cumple*

$$\begin{aligned}\left\{\binom{n}{k}\right\} = &\left\{\binom{n}{k_\ell}\right\} \otimes B_{p^\ell} \odot \left\{\binom{n}{k_{\ell-1}}\right\} \otimes B_{p^{\ell-1}} \odot \\ &\cdots \odot \left\{\binom{n}{k_1}\right\} \otimes B_p \odot \left\{\binom{n}{k_0}\right\} \otimes B_{p^0}.\end{aligned}$$

Demostración. El Lema 21 implica que

$$\left\{\binom{n}{k}\right\} = \left\{\binom{n}{k_\ell p^\ell + k_{\ell-1}p^{\ell-1} + \cdots + k_1 p + k_0}\right\}$$
$$= \left\{\binom{n}{k_\ell p^\ell}\right\} \odot \left\{\binom{n}{k_{\ell-1}p^{\ell-1} + \cdots + k_1 p + k_0}\right\}.$$

De nuevo, por el Lema 21 tenemos

$$\left\{\binom{n}{k_\ell p^\ell}\right\} = \left\{\binom{n}{k_\ell p^{\ell-1}}\right\} \otimes B_p = \cdots$$
$$= \left\{\binom{n}{k_\ell}\right\} \otimes \underbrace{B_p \otimes \cdots \otimes B_p}_{\ell \text{ veces}} = \left\{\binom{n}{k_\ell}\right\} \otimes B_{p^\ell}.$$

Combinando ambas igualdades concluimos que

$$\left\{\binom{n}{k}\right\} = \left\{\binom{n}{k_\ell}\right\} \otimes B_{p^\ell} \odot \left\{\binom{n}{k_{\ell-1}p^{\ell-1} + \cdots + k_1 p + k_0}\right\}.$$

Reiterando el proceso con $\left\{\binom{n}{k_{\ell-1}p^{\ell-1}+\cdots+k_1 p+k_0}\right\}$ se obtiene el resultado. ∎

Como consecuencia inmediata del Teorema 22, se obtiene el Teorema 4 tal como sigue: sean L, r enteros no negativos. Para $j \in \{1, 2, \ldots, p-1\}$, denotamos $\left\{\binom{n}{j}\right\} = \{s_0^{(j)}, s_1^{(j)}, \ldots, s_{p-1}^{(j)}\}$. Entonces:

1) Para los múltiplos de potencias de p,

$$\left\{\binom{n}{jp^L}\right\} = \left\{\binom{n}{j}\right\} \otimes B_{p^L}$$

2) Para $i = 1, 2, \ldots, L$ y $5^{i-1} \leq r < 5^i$

$$\left\{\binom{n}{jp^L + r}\right\} = \left\{\binom{n}{j}\right\} \otimes B_{p^L} \odot \left\{\binom{n}{r}\right\}$$

Concluimos con un teorema que muestra cómo escribir una secuencia de periodo p^L como combinación de las primeras p secuencias binomiales.

Teorema 23. *Sea $\{s_n\}$ una secuencia con representación binomial $\sum_{k=0}^{p^L-1} c_k \left\{\binom{n}{k}\right\}$, y sea $k = k_\ell p^\ell + \cdots + k_1 p + k_0$ la representación p-aria de k. La secuencia $\{s_n\}$ puede escribirse como:*

$$\sum_{k=0}^{p^L-1} c_k \left\{\binom{n}{k}\right\} = \sum_{k=0}^{p^L-1} c_k \left(\bigodot_{i=0}^{\ell} \left\{\binom{n}{k_i}\right\} \otimes B_{p^i}\right).$$

V. Conclusiones y trabajo futuro

En este trabajo hemos ampliado los resultados de [2] sobre secuencias binomiales de $\mathbb{F}_2$ a $\mathbb{F}_p$. Además, hemos demostrado algunas propiedades estructurales de secuencias en $\mathbb{F}_p$ de periodo p^L; hemos establecido una conexión entre el polinomio característico y la complejidad lineal de estas secuencias y los de las secuencias binomiales de su descomposición binomial; y hemos estudiado la acción del operador de desplazamiento sobre dichas secuencias. Hemos probado que cualquier secuencia binomial puede obtenerse a partir de las primeras p secuencias binomiales.

Se sabe que las secuencias binomiales binarias pueden generarse mediante autómatas celulares lineales. Como trabajo futuro, pretendemos investigar si estructuras similares sobre cuerpos finitos pueden producir secuencias binomiales p-arias, así como el comportamiento de sus combinaciones lineales. Además, tenemos la intención de explorar generadores de secuencias de decimación irregular en el contexto de cuerpos finitos.

Agradecimientos

El trabajo del primer y del tercer autor fue parcialmente financiado por el proyecto español de I+D+i PID2022-142159OB-I00 del Ministerio de Ciencia e Innovación, el proyecto de I+D+i CIAICO/2022/167 de la Generalitat Valenciana y el proyecto de I+D+i VIGROB-287 de la Universidad de Alicante. El segundo autor contó con el apoyo del CNPq mediante el proceso 405842/2023-6 y de la FAPESP mediante el proceso 2024/05051-7.

References

[1] S. W. Golomb and G. Gong, *Signal Design for Good Correlation: For Wireless Communication, Cryptography, and Radar*. Cambridge University Press, 2005.

[2] S. D. Cardell and A. Fúster-Sabater, "Binomial representation of cryptographic binary sequences and its relation to cellular automata," *Complexity*, vol. 2019, pp. 1–13, 2019.

[3] S. D. Cardell and A. Fúster-Sabater, "Linear models for the self-shrinking generator based on ca," *Journal of Cellular Automata*, vol. 11, no. 2-3, pp. 195–211, 2016.

[4] R. Smarandache, H. Gluesing-Luerssen, and J. Rosenthal, "Strongly mds convolutional codes, a new class of codes with maximal decoding capability," in *Proceedings IEEE International Symposium on Information Theory,*, 2002, pp. 426–.

Migration Strategies for Ethereum Externally Owned Accounts under the Quantum Threat

Marco López
Decentralized Security, Málaga
marco@decentralizedsecurity.es

Isaac Agudo
NICS Lab, Málaga
isaac@uma.es

***Abstract*—A cryptographically relevant quantum computer (CRQC) running Shor's algorithm would break secp256k1-ECDSA, enabling private key extraction from any Ethereum externally owned account (EOA) whose public key has been revealed, either on-chain or by observing a signed transaction off-chain (e.g., in the mempool or relays). The monetary exposure is systemic: EOAs secure user funds, DeFi control flow, and operational keys across a large economy. Long-term remediation is widely expected to rely only on either native post-quantum (PQ) signature schemes or native account abstraction (AA), but both directions demand multi-year coordination across the protocol, client software, wallets, tooling, and users.**

This paper surveys the main migration strategies currently discussed in the Ethereum community for EOAs under a CRQC threat, including native PQ signatures, ERC-4337 smart-contract wallets, emerging native AA proposals (RIP-7560/EIP-7701), and EIP-7702-based delegation. We compare their security properties and UX impact, highlight compatibility challenges (for example contracts relying on `ECRECOVER`, ERC-2612 `permit` flows or `tx.origin`), and discuss their expected deployment timelines.

***Index Terms*—Ethereum, quantum security, post-quantum cryptography, blockchain security**

I. Introduction

Ethereum transactions from externally owned accounts (EOAs) are authorized with secp256k1 ECDSA signatures [1]. Bitcoin makes the same basic bet on elliptic curves, historically ECDSA and, since Taproot, Schnorr, so the shared assumption is the hardness of the elliptic-curve discrete logarithm problem [2]. In Ethereum, an EOA's address is the least significant 20 bytes of its public key hash, which means that the public key remains hidden until the account sends a transaction. A key nuance specific to Ethereum is that public keys are not explicitly included in transactions but are *recoverable* from ECDSA signatures. Concretely, transaction signatures include the recovery identifier alongside (r, s), which allows reconstructing the signer's secp256k1 public key from the message hash and the signature (as exposed in the protocol via the `ECRECOVER` precompile) [1]. Therefore, once a signature is observable, the corresponding public key becomes derivable, and a future cryptographically relevant quantum computer (CRQC) running Shor's algorithm could compute the corresponding private key, gaining ownership of the EOA and being able to sign arbitrary transactions [3]. The monetary risk is large: Ethereum secures user balances, protocol treasuries, DeFi positions, and more. The need for quantum resistance in public blockchains, and for Ethereum in particular, is already widely recognized in both academic surveys and ecosystem discussions where post-quantum signature schemes and stronger forms of account abstraction are often identified as likely end-states [4], [5].

While the *end-state* may be native AA and PQ cryptography, getting from today's ECDSA-centric pipeline to a fully quantum-safe architecture will be a lengthy process. Wallets, client software, explorers, bridges, smart contracts, signers, and user habits must all adapt; users would need to migrate their accounts; DeFi positions and token allowances need to be re-bound; and Layer-2 networks must align their account models with Layer 1 changes. Given that pace, there is a need for a built-in mechanism that buys time, and keeps flexibility in the choice of PQ scheme. In this paper, we survey the main strategies for Ethereum under a CRQC threat and compare their trade-offs, with a focus on transitional mechanisms that can be deployed before long-term solutions are available.

A. Contributions and Organization

In this article, we: (i) survey migration and mitigation strategies for EOAs under a CRQC threat, including native PQ signatures, ERC-4337 account abstraction (AA), emerging native AA proposals, and EIP-7702 delegation and key deactivation variants; (ii) provide a comparative analysis of these approaches and their trade-offs, summarized in *Table 1*; and (iii) discuss security, usability and compatibility challenges, including the impact on contracts that rely on `ECRECOVER`, `permit` patterns, `tx.origin` and cross-chain ownership assumptions.

II. Background & Threat Model

A. CRQC threat to EOAs

A sufficiently powerful quantum computer running Shor's algorithm breaks ECDSA, allowing an adversary to derive the private key from a known public key. In Ethereum, once an EOA produces an observable ECDSA signature, its public key becomes recoverable and the account becomes a candidate for CRQC-based key recovery.

Public-key recovery in Ethereum.: Ethereum uses ECDSA over secp256k1 to authorize EOA transactions. In Ethereum, an ECDSA signature is typically represented by the tuple (v, r, s), where r and s are the standard ECDSA values and v is a recovery identifier that disambiguates the corresponding elliptic-curve point. Given the signed message hash and (v, r, s), the signer's public key can be recovered, and the EOA address can be re-derived as the last 20 bytes of `keccak256(pubkey)` [1]. This capability is exposed via the `ECRECOVER` precompile and is widely used by smart contracts for signature-based authorization.

Off-chain exposure before inclusion.: Public-key exposure is not limited to transactions that are eventually included on-chain. Any party that observes a signed transaction off-chain—for example in the public mempool, peer-to-peer gossip, or relay/builder infrastructure—can already recover the sender's public key before confirmation. While Ethereum's short block interval reduces the *typical* exposure window, inclusion can still be delayed (e.g., due to fee underpricing, congestion, replacement transactions, temporary censorship, or relay dynamics). Therefore, in CRQC-oriented threat models, "broadcast" should be treated as sufficient for public-key disclosure even if the transaction is never included.

Once the public key is exposed, a future CRQC adversary could compute the corresponding private key and forge transactions, enabling theft of funds and account takeover. Such an adversary could also exploit any smart contract that relies on signature recovery for authentication. We assume hash functions remain secure against preimage attacks aside from known quadratic speed-ups (Grover's algorithm) [6]. Thus, hash-based signature schemes (e.g., XMSS, WOTS+, SPHINCS+) remain viable in principle with adequately large parameters and careful instantiation [4], [7].

B. Timelines and migration friction

Adopting PQ-safe cryptography for Ethereum's execution layer involves changes at multiple levels. In this work we focus on the transaction pipeline and the account model, not on PQ-secure consensus or networking, where complementary post-quantum designs are being explored [8]. The changes include: (1) the transaction validation process (new signature algorithms and verification rules); (2) address formats or how public keys are represented (since PQ public keys or signatures may be much larger than current ones); (3) wallet key derivation, and recovery processes (which currently rely on elliptic-curve keys and BIP-39/BIP-44 standards); (4) existing DeFi contracts and token allowances (which are bound to specific EOA addresses and ECDSA signatures via constructs like permit signatures, as in ERC-2612 [9]); and (5) interoperability with Layer-2 (L2) networks and bridges. Even once the community agrees on a specific PQ signature scheme, the engineering, testing, and user experience (UX) updates would span multiple network upgrades and client releases [10].

These migration challenges coexist with ongoing efforts to make accounts more flexible. A form of account abstraction is already available at the application layer via ERC-4337, which allows users to use smart-contract wallets with custom verification logic [11]. However, even ERC-4337 does not remove ECDSA from the transaction pipeline today: the *bundler* (the off-chain agent that packages user operations) currently must submit an ECDSA-signed transaction to the EntryPoint contract on L1 [12]. Truly eliminating ECDSA from transactions requires deeper protocol changes. Two notable proposals, RIP-7560 and its associated EIP-7701, aim to implement native account abstraction in Ethereum's protocol, removing the legacy EOA model altogether [13], [14]. These proposals would let the protocol accept transactions authorized by arbitrary verification logic, but achieving this in a backward-compatible, secure, and scalable way is a complex endeavor. It implies significant changes to how transactions are handled in the consensus layer, how gas and fees are managed (to prevent abuse of flexible verification), and how tooling (such as wallets and block explorers) represent accounts. Additionally, any L2 or bridging solution would need to adapt to these changes to maintain compatibility.

In summary, the transition to a post-quantum Ethereum will take time and faces coordination friction. During this interim period, EOAs remain vulnerable to a sudden breakthrough in quantum computing. The next section outlines the range of approaches being considered to mitigate the CRQC threat for EOAs, from long-term protocol upgrades that deeply modify the execution layer to more incremental mechanisms that can be adopted sooner, albeit still requiring non-trivial protocol and ecosystem work.

III. Design Space of Responses

We now survey the main strategies for securing EOAs against the threat of quantum attackers. *Table 1* provides a side-by-side summary of the approaches, comparing where the verification takes place, residual reliance on ECDSA, implementation complexity, impact on user experience, estimated timelines, and other notes.

A. Native PQ signature

One direction is to integrate PQ signatures directly into the Ethereum protocol verification path, either via precompiles or efficient EVM code. Candidate families include hash-based signatures (such as LMS, XMSS, WOTS+, or SPHINCS+) and lattice-based signatures (such as Falcon or Dilithium), as well as emerging multivariate designs like MAYO, all of which are discussed in the post-quantum blockchain literature [4], [5]. Notably, hash-based schemes come in *stateful* vs *stateless* variants. LMS (Leighton-Micali Signature) and XMSS are stateful schemes, meaning the signer must track a one-time key index and update it for each signature to avoid reuse [15]. In contrast, SPHINCS+ is stateless, it does not require maintaining state at the cost of larger signatures and more extensive verification work. Standards bodies have generally favored stateless designs for broad use (indeed NIST selected SPHINCS+ as a standard PQ digital signature in 2024) [16] due to their simpler key management and no risk of state misuse. In practice, stateless schemes trade some efficiency for this convenience, for example, SPHINCS+ signatures are significantly larger (on the order of tens of kilobytes) and tend to be slower to verify than stateful schemes. Nonetheless, in a blockchain context the "state" required by XMSS/LMS can be managed. The global state (e.g. an account's storage) could track the next unused one-time signature index, preventing reuse. Indeed, the Quantum Resistant Ledger (QRL) blockchain has successfully used XMSS (a stateful scheme) for all transactions, demonstrating that on-chain key state tracking is feasible (albeit with some added operational complexity for users and apps) [17].

On-chain verification cost varies widely across different PQ schemes. Recent measurements report that verifying a hash-based one-time signature or an XMSS tree can consume many millions of gas, whereas an optimized Falcon verification (especially if aided by a Number Theoretic Transform precompile) could be on the order of a few million gas [18], [19], [10]. Signature and key sizes also differ greatly between

schemes (from a few hundred bytes for Falcon to tens of kilobytes for some SPHINCS+ parameters), which affects transaction sizes and how addresses are derived. Introducing native PQ signatures would require careful evaluation of gas costs, performance, and security trade-offs, but it offers a straightforward conceptual path: replace ECDSA with a quantum-resistant scheme for EOAs. Prototypes like ETH-FALCON (an effort to implement Falcon on the EVM) show that it is feasible to experiment with PQ signature verification on-chain today [20]. However, reaching consensus on a specific algorithm and implementing it across all clients is a long-term effort, and any chosen scheme must be considered secure against quantum attacks (which is an evolving target as cryptanalysis progresses).

B. ERC-4337 Account Abstraction (AA)

ERC-4337 is a standard that enables account abstraction without requiring changes to the consensus protocol [11]. It introduces the concept of UserOperations that are validated by a special contract (the EntryPoint) rather than by the protocol directly. Users can have smart contract wallets with arbitrary verification logic (including the ability to use PQ signatures in that logic). Because the verification happens within the contract, a wallet contract could today be designed to require a PQ signature scheme for authorizing operations, theoretically making the user's account quantum-resistant. The catch is that the user's operation still needs to be packaged into a transaction by a bundler. Currently, the bundler must sign an ECDSA transaction to get the UserOperation included on L1 [21]. This means the overall flow is not fully PQ-safe, since an attacker with a CRQC could target the bundler's key or the protocol's requirement for an ECDSA signature on the final transaction. Nevertheless, ERC-4337 significantly reduces the exposure of user keys to ECDSA-based quantum attack (the user's own key can be PQ-safe and never used in an ECDSA context). It is also available now on Ethereum without any protocol changes, making it an attractive mid-term solution. The downsides include added complexity (it introduces mempool-like infrastructure and pays for operations via fees in a separate mechanism) and that it doesn't protect the entire system (just the end-user wallets). It's considered a stepping stone and is often discussed in conjunction with the road to full account abstraction and post-quantum transaction flows [22].

C. Native AA proposals (RIP-7560/EIP-7701)

Native AA proposals aim to remove the special status of EOAs entirely. Instead of an EOA being implicitly defined by an ECDSA key, any account would simply have associated verification logic (which could be a smart contract or some code specified in the protocol). RIP-7560 and EIP-7701 sketch out a design where transactions undergo a validation phase (running the account's logic to check signatures or other conditions), an execution phase, and possibly a post-operation phase [13], [14]. In such a design, it would be possible to plug in a post-quantum verification algorithm at the protocol level for an account, or use multisigs, biometrics, etc., as needed. This approach cleanly eliminates ECDSA and makes the chain agnostic to the type of signature or scheme being used. It is the "clean" end-state in terms of security architecture. However, native AA requires fundamental changes to Ethereum's consensus rules and transaction format. It raises questions about gas costs, how to handle accounts that currently use `tx.origin` or have other implicit assumptions, and how to migrate millions of existing users safely. The timeline for any native AA is uncertain, it could be many years before such a change is agreed upon and deployed, considering the caution required for Ethereum upgrades and the need for extensive testing (including how such changes propagate to Layer 2 systems). Thus, while native AA is a powerful solution (and inherently quantum-proof if PQ algorithms are used for verification), it might not arrive before large quantum computers do.

D. EIP-7702 delegation

EIP-7702 lets an EOA delegate to code, effectively turning it into a "delegated" account [23]. This mechanism lets a regular account execute arbitrary validation and execution logic on a per-transaction basis, without permanently converting the account into a contract. The immediate benefit is that an EOA could use this delegation to gain features of contract accounts (multi-signatures, additional checks, batching of operations, etc.) on demand. Under a CRQC threat, a user could delegate to a contract that implements a quantum-resistant validation (for example, requiring a PQ signature or multiple different signatures). In other words, 7702 can serve as a hook to plug in crypto-agile verification for an EOA without waiting for a full protocol change. The approach does not force a long-term commitment to a specific scheme (the user could change the delegation target or revert to normal ECDSA use for subsequent transactions if desired). However, a crucial limitation is that the original EOA's private key remains a master authority. At any time, the owner could simply delegate to a different contract by signing a new delegation transaction with the ECDSA key. This means if an adversary obtains the ECDSA key (e.g., via CRQC), they could bypass the delegated contract by issuing their own delegation. In summary, EIP-7702 provides flexibility and an immediate path to experiment with PQ-aware account logic while keeping EOAs and their addresses, in contrast to ERC-4337 where users migrate to separate smart-contract accounts. Like account abstraction mechanisms, 7702 increases flexibility by moving validation into code, but it does so via a protocol-level delegation primitive that lets existing EOAs temporarily attach contract-like logic instead of changing the account model wholesale. At the same time, it does not by itself revoke the powers of the vulnerable ECDSA key. This residual risk must be addressed if 7702 delegation is to be a serious quantum-mitigation strategy.

E. EIP-7702 + Key Deactivation Variant

To fully close the loop on the above approach, the community has discussed additions like EIP-7851 (Deactivate/Reactivate a Delegated EOA's Key), which would allow an EOA's original key to be temporarily or permanently deactivated once it has delegated to a contract [24]. In practice, this would mean creating a special transaction (or flag) after which the protocol no longer accepts direct ECDSA signatures from that account, only the delegated contract logic would be

honored. Such a mechanism would preserve the user's address (so all assets and contracts tied to the address remain intact) but would route control exclusively through the contract. It effectively transforms the EOA into a contract account while maintaining its identity. This idea is analogous to earlier "account migration" proposals like EIP-7377 (a transaction to convert an EOA to a contract account) [12], but tailored specifically to disable ECDSA usage in 7702-delegated EOAs. Implementing this variant would require a consensus change or a new transaction type, and great care in handling edge cases. If achieved, however, it would combine the flexibility of 7702 delegation with a true removal of ECDSA risk for that account. We note that as of late 2025, key-deactivation mechanisms such as EIP-7851 remain proposals under discussion, not deployed features [24].

IV. Discussion: Trade-offs and Challenges

Each of the approaches above comes with trade-offs in terms of security, complexity, and usability. Native PQ signatures and native AA are the most comprehensive solutions, they would fundamentally solve the quantum threat by replacing vulnerable primitives or by removing fixed schemes entirely. However, both require long lead times and significant consensus-building. ERC-4337 and EIP-7702 delegation, on the other hand, leverage what is already possible on Ethereum today to provide a measure of quantum safety without waiting for protocol upgrades. However, these interim solutions inherit some limitations of the current infrastructure: ERC-4337 still involves ECDSA at the bundler level, and EIP-7702 still relies on the EOA's original key to initiate delegations.

Table 1 encapsulates the high-level comparison of the approaches. As shown, the only strategies that completely eliminate ECDSA from the picture are native solutions (either PQ signatures or native AA). Strategies that retain some dependence on ECDSA, even indirectly, carry a residual risk that a CRQC could exploit that weakest link.

Security posture and residual key risk

A central axis in this design space is the *residual ECDSA authority* that remains after applying each mitigation. Native PQ signatures and native AA can in principle eliminate ECDSA from transaction validation. ERC-4337 and EIP-7702, by contrast, retain ECDSA at different layers: the bundler in ERC-4337 must submit an ECDSA-signed transaction, and the underlying EOA in EIP-7702 continues to be the root of authority as long as the protocol accepts its signatures. This residual authority is particularly problematic in a CRQC setting. Even if an account is fronted by PQ-aware validation logic, a future quantum adversary that recovers the underlying ECDSA key could sidestep those protections. In the ERC-4337 case, this may manifest as attacks on bundler keys or on any remaining ECDSA-based meta-transaction infrastructure; in the EIP-7702 case, the attacker can simply issue a new delegation or a direct transaction that ignores the intended validation contract. Key-disabling variants aim to address precisely this failure mode by introducing a way for an EOA to *irreversibly* relinquish its ability to authorize transactions directly once a more flexible validation mechanism is in place. Proposals such as EIP-7851 illustrate this direction by suggesting flags or special transaction types after which the protocol will refuse plain ECDSA signatures from that address [24]. However, reducing ECDSA usage does not automatically eliminate all ECDSA-based risks: off-chain signatures, historical permits, and application-layer protocols may continue to rely on them.

Backward compatibility

Any change away from EOAs and secp256k1 affects the broader Ethereum ecosystem. For example, many smart contracts today use the `ECRECOVER` precompile to check signatures (for instance, verifying an ECDSA signature within a contract for login, meta-transactions, or permit functions). If users move to PQ signatures or to AA-style accounts, those contracts would not recognize the new signatures unless they are upgraded. Similarly, contracts that rely on identifying EOAs (versus contract accounts) might use `tx.origin`, assume that `extcodesize(address)` being zero implies an EOA, or implement authorization rules that depend on an account having no code. These assumptions break down under both AA and EIP-7702, where accounts can be represented by arbitrary logic or delegate to contracts on demand. A notable example is the ERC-2612 permit pattern, which allows an EOA to pre-sign an approval for a token and have someone else submit it on-chain. These signed permits remain valid as long as the deadline has not passed and are not automatically invalidated by any migration. If an attacker ever gains control of the private key of an EOA, they can sign arbitrarily many new `permit` messages. Even if the protocol later refuses direct transactions from that key (for example, after a key-deactivation step), contracts that verify ECDSA signatures via `ECRECOVER` will still accept those permits. Off-chain signatures cannot be retroactively cancelled or globally revoked once issued, so migration strategies must account for this persistent attack surface.

Cross-chain and Layer-2 consistency

Ethereum's multi-chain environment further complicates EOA migration. A user might adopt a contract-based or delegated account model on Ethereum mainnet (L1), but that change does not automatically propagate to Layer-2 networks or sidechains where the same address is still treated as a classical EOA with a valid ECDSA key. From the perspective of an Optimistic or ZK rollup, for example, the address may remain fully controlled by the underlying key unless a coordinated migration is performed there as well. This disparity can lead to *state drift*, where the same logical identity corresponds to a contract or delegated account on one chain and to a vulnerable EOA on another. Attackers could exploit this by targeting the weakest environment, for instance using the original key on a less secure chain to move assets or to influence protocols that affect L1 positions via bridges. Conversely, attempts to lock down ECDSA usage on one chain (e.g., via a key-deactivation mechanism) might have unintended consequences if other chains still rely on that key for safety-critical operations. Several research directions emerge from this observation. One is to design cross-chain migration attestations, where an L1 migration produces a verifiable signal that can be consumed by L2s and sidechains to trigger analogous changes. Another is to build migration-aware bridge contracts that refuse to move assets for addresses whose status is inconsistent across

TABLE I
COMPARISON OF EOA RESPONSE STRATEGIES UNDER A CRQC THREAT. EACH APPROACH IS CHARACTERIZED BY WHERE TRANSACTION VERIFICATION OCCURS, WHETHER ECDSA REMAINS IN THE CRITICAL PATH, COMPLEXITY AND GAS COST CONSIDERATIONS, IMPACT ON USER EXPERIENCE, A ROUGH TIMELINE ESTIMATE FOR DEPLOYMENT OR USABILITY, AND ADDITIONAL NOTES.

Approach	Where verified	ECDSA rel.	Gas/complexity	UX impact	Timeline	Notes
Native PQ Signatures e.g., SPHINCS+, Falcon, XMSS/WOTS+	Protocol level	None	High. Requires consensus on scheme, significant changes to Ethereum protocol and client software. Needs wallet changes, new transaction format, new precompiles, etc.	Moderate; new key derivation and possibly new addresses	Long-term	Strong end-state; careful sizing
ERC-4337 AA	Contract (EntryPoint)	Yes (bundler)	Medium; off-chain infra	Users need to switch from EOAs to contract wallets	Live	Enables custom verification in wallets, but not end-to-end PQ secure since the bundler uses ECDSA
Native AA (RIP-7560/EIP-7701)	Protocol level	No	High. Requires significant changes to Ethereum protocol and client software. Needs new transaction format, new gas rules for validation phase, etc.	Eventually seamless for users, but big changes for tooling and contracts that assume EOAs	Long-term	Clean solution but increases complexity
EIP-7702 Delegation	Delegated contract	Yes	Low–Medium	Minimal change. Wallet needs to support crafting delegation transactions	Live	Enables crypto-agile validation; residual key risk [23].
EIP-7702 + Key Deactivation	Delegated contract	No (for that EOA)	Medium. Introduces a new transaction type or flag at protocol level. Some complexity in key management	Moderate; wallet must handle rotating keys	Mid-term	Preserves address while closing the main ECDSA loophole. Requires consensus change

domains or that require additional checks when a source or destination chain still accepts vulnerable keys. More broadly, surveys on PQ DLTs highlight how heterogeneous ledger designs complicate coordinated upgrades [5], [4].

Wallets, UX, and operational risk

Achieving broad adoption of any EOA migration strategy will require deep integration with wallet providers (browser extensions, hardware devices, mobile wallets) and careful UX design. From the user's perspective, migrating to a quantum-resilient account should be as simple and reversible as possible, yet many of the underlying cryptographic schemes introduce new operational burdens. For native PQ signatures, users may need to handle larger keys and signatures, different mnemonic formats, and new backup and recovery procedures. Hardware wallets must support new algorithms, which may require firmware updates and careful side-channel evaluations. These considerations suggest that any proposed migration path must be evaluated not only for its protocol-level security but also for its end-to-end operational risk profile. A design that is cryptographically sound but operationally fragile may leave many users no better off in practice.

Research directions for EIP-7702 + key-deactivation

Extended EIP-7702 approaches, in which the EOA delegates to richer validation logic and the protocol subsequently restricts or disables direct use of the underlying ECDSA key, represent a promising but complex family of mitigations. At a high level, such mechanisms aim to preserve the user's address and its existing relationships (balances, allowances, contract roles) while routing control through a crypto-agile layer that can enforce PQ signatures, key rotation, or other advanced policies.

Designing these mechanisms raises several open questions:

- **Safety of deactivation and recovery.** Once an EOA's direct signing capability is restricted, what recovery paths remain if the delegated or validator logic fails, is upgraded incorrectly, or is compromised? How can protocols distinguish between a legitimate recovery action and an attacker's attempt to regain ECDSA-based control?
- **Interaction with existing contracts and standards.** How should deactivation interact with ERC-2612-style permits, meta-transaction relayers, and contracts that use `ECRECOVER` internally? Can migrated accounts advertise their status in a standardized way so that applications can adapt their logic?
- **MEV and transaction-ordering effects.** Migration and deactivation transactions may be highly valuable to users and therefore to searchers. Analysing whether new MEV opportunities arise (e.g., racing deactivation with malicious permits or with cross-chain withdrawals) is necessary to avoid subtle incentive attacks.
- **Operational guidance and tooling.** Finally, even if the protocol mechanisms are well specified, practitioners will need guidance, static analysis tools, and best practices to perform migrations safely at scale. This includes detection of dangerous patterns (e.g., reliance on `tx.origin`), visualization of an account's outstanding off-chain authorizations, and simulation tools for proposed migration plans.

We see extended EIP-7702 and key-deactivation not as a single scheme, but as a broader research direction. Key next

steps are specification, prototyping, and end-to-end evaluation across the Ethereum ecosystem.

V. Conclusion

The prospect of a cryptographically relevant quantum computer has cast a long shadow over current public-key cryptography. For Ethereum, the threat is especially acute for externally owned accounts protected by ECDSA, a foundational assumption that underpins the security of user funds and contracts. In this paper, we surveyed the landscape of possible mitigation strategies, from deploying post-quantum signature schemes at the protocol level to leveraging various forms of account abstraction (both contract-based and protocol-native) and delegation-based mechanisms such as EIP-7702. *Table 1* and our discussion highlight that each strategy has its strengths and weaknesses: some offer comprehensive security but require long timelines and deep protocol changes, while others can be implemented sooner but leave residual risks.

A key insight from this exploration is that **timing matters**. The community cannot afford to wait until a perfect long-term solution is ready if a CRQC might emerge in the meantime. Therefore, a layered approach is advisable. Intermediate solutions can provide **crypto-agility** and reduce the window of exposure, particularly for high-value accounts, while longer-term native upgrades are designed. In this context, EIP-7702 delegation with key-deactivation variant appears as a promising ingredient for transitional migrations. It preserves existing addresses and relationships while offering robust validation logic. However, as our discussion of trade-offs shows, making such approaches truly safe and usable requires careful attention to residual key authority, compatibility with existing contracts and standards, cross-chain consistency, and wallet UX. Fully specifying and evaluating concrete schemes along these lines is an important avenue for future work.

At the same time, pursuing **native upgrades** remains crucial. Features like native account abstraction and built-in support for PQ signatures are the endgame for a truly quantum-resistant Ethereum. To maintain the health of Ethereum's ecosystem during this transition, a broad community effort is needed. Protocol developers, wallet providers, smart contract developers, and end-users all have roles to play in hardening accounts against the quantum threat. Education will be as important as engineering: many researchers and developers in the broader security community are still coming up to speed on Ethereum-specific concepts, so clear communication will help in driving home the importance of proactive migration.

Acknowledgments

We thank the Ethereum community for public discussions and artifacts that informed this position paper.

References

[1] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," Ethereum Yellow Paper, 2014. [Online]. Available: https://ethereum.github.io/yellowpaper/paper.pdf

[2] S. D. Galbraith and P. Gaudry, "Recent progress on the elliptic curve discrete logarithm problem," *Designs, Codes and Cryptography*, vol. 78, no. 1, pp. 51–72, 2016. [Online]. Available: https://doi.org/10.1007/s10623-015-0146-7

[3] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, 1997. [Online]. Available: https://doi.org/10.1137/S0097539795293172

[4] T. M. Fernández-Caramès and P. Fraga-Lamas, "Towards post-quantum blockchain: A review on blockchain cryptography resistant to quantum computing attacks," *IEEE Access*, vol. 8, pp. 21 091–21 116, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.2968985

[5] N. K. Parida, C. Jatoth, V. D. Reddy, M. M. Hussain, and J. Faizi, "Post-quantum distributed ledger technology: A systematic survey," *Scientific Reports*, vol. 13, p. 20729, 2023. [Online]. Available: https://doi.org/10.1038/s41598-023-47331-1

[6] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC)*, 1996, pp. 212–219. [Online]. Available: https://doi.org/10.1145/237814.237866

[7] A. Hülsing, D. Butin, S. L. Gazdag, J. Rijneveld, and T. Mohaisen, "XMSS: extended merkle signature scheme," RFC 8391, IETF, 2018. [Online]. Available: https://datatracker.ietf.org/doc/rfc8391/

[8] J. Drake, D. Khovratovich, M. Kudinov, and B. Wagner, "Hash-based multi-signatures for post-quantum ethereum," Cryptology ePrint Archive, Paper 2025/055, 2025. [Online]. Available: https://eprint.iacr.org/2025/055

[9] F. Royer, "EIP-2612: Permit extension for erc-20," Ethereum Improvement Proposal, 2019. [Online]. Available: https://eips.ethereum.org/EIPS/eip-2612

[10] J. Juaristi and I. Agudo, "Benchmarking post-quantum cryptography in ethereum-based blockchains," in *Trust, Privacy and Security in Digital Business (TrustBus 2024)*, ser. Lecture Notes in Computer Science. Springer, 2024, vol. 15263, pp. 79–95. [Online]. Available: https://doi.org/10.1007/978-3-031-73397-0_6

[11] Y. Dong *et al.*, "EIP-4337: Account abstraction using alternative mempool," Ethereum Improvement Proposal, 2021. [Online]. Available: https://eips.ethereum.org/EIPS/eip-4337

[12] A. Gluchowski, "EIP-7377: Migration transaction," Ethereum Improvement Proposal (draft), 2023. [Online]. Available: https://eips.ethereum.org/EIPS/eip-7377

[13] V. Buterin *et al.*, "Native account abstraction," Ethereum RIP Specification, 2023. [Online]. Available: https://github.com/ethereum/RIPs/blob/master/RIPS/rip-7560.md

[14] G. Kent and collaborators, "EIP-7701: Native account abstraction," Ethereum Improvement Proposal, 2024. [Online]. Available: https://eips.ethereum.org/EIPS/eip-7701

[15] F. Campos, T. Kohlstadt, S. Reith, and M. Stöttinger, "LMS vs XMSS: Comparison of Stateful Hash-Based Signature Schemes on ARM Cortex-M4," in *Progress in Cryptology – AFRICACRYPT 2020*, ser. LNCS, vol. 12174. Springer, 2020, pp. 258–277.

[16] National Institute of Standards and Technology, "Stateless Hash-Based Digital Signature Standard, FIPS 205," NIST, Tech. Rep., 2024. [Online]. Available: https://doi.org/10.6028/NIST.FIPS.205

[17] "The quantum resistant ledger (qrl)," https://www.theqrl.org, 2016, accessed: 2025-11-20.

[18] R. Kysil, A. Varre, M. Katiyar, and V. Kovtun, "Poqeth: Efficient post-quantum signature verification on ethereum," in *Proceedings of the 2025 ACM International Conference on Distributed and Event-Based Systems (DEBS)*, 2025, pp. 181–192, also available as IACR ePrint 2025/091. [Online]. Available: https://eprint.iacr.org/2025/091

[19] R. Dubois, "NTT as post-quantum and STARKs settlements helper precompile," Ethereum Research Forum Post (non peer-reviewed), Feb. 2025. [Online]. Available: https://ethresear.ch/t/ntt-as-postquantum-and-starks-settlements-helper-precompile/21775

[20] ZKNoxHQ, "ETHFALCON: Study and implementation for falcon adaptations for the ethereum ecosystem," GitHub repository (implementation artifact), 2025. [Online]. Available: https://github.com/ZKNoxHQ/ETHFALCON

[21] OpenZeppelin, "The entrypoint contract (ERC-4337)," Documentation Page, 2025. [Online]. Available: https://docs.erc4337.io/smart-accounts/entrypoint-explainer.html

[22] A. Sanso, "The road to post-quantum ethereum transaction is paved with account abstraction (aa)," Ethereum Research Forum Post, Feb. 2025. [Online]. Available: https://ethresear.ch/t/the-road-to-post-quantum-ethereum-transaction-is-paved-with-account-abstraction-aa/21783

[23] V. Buterin, "EIP-7702: Set eoa account code for one transaction (delegation)," Ethereum Improvement Proposal, 2024. [Online]. Available: https://eips.ethereum.org/EIPS/eip-7702

[24] J. Norvill, "EIP-7851: Deactivate/reactivate a delegated eoa's key," Ethereum Improvement Proposal (draft), 2024. [Online]. Available: https://eips.ethereum.org/EIPS/eip-7851

Uso de LLMs para el análisis de vulnerabilidades en Solana "Programs"

Laura De Haro García
E.T.S.I Informática
Universidad de Málaga
lauradhg18@uma.es

Isaac Agudo
NICS Lab
Universidad de Málaga
isaac@uma.es

***Resumen*—Este estudio analiza el uso de grandes modelos de lenguaje (LLMs) para detectar vulnerabilidades en contratos inteligentes o *Programs* de la *Blockchain* Solana escritos en Rust. Se examinan once tipos de vulnerabilidades, entre ellas verificación de propiedad y firma, control de privilegios, desbordamientos de enteros, confusión de cuentas, verificación de cuentas y parámetros pasados, control de instrucciones invocadas y uso de funciones obsoletas, en un conjunto de 126 contratos. Para ello, se emplean cuatro modelos (*GPT Nano 4.1*, *Gemini Flash 2.0*, *DeepSeek V3.1* y *Qwen3 235B A22B*) con distintas especificaciones de *prompt*. El análisis estadístico se centra en evaluar la capacidad de clasificación binaria de cada vulnerabilidad y la precisión en el conteo de apariciones mediante métricas de matriz de confusión y funciones de pérdida (MAE y RMSE). El trabajo busca valorar tanto el potencial como los riesgos de aplicar modelos de lenguaje en la seguridad de los *Programs* de Solana.**

***Index Terms*—Blockchain, Solana, Program, Smart contract, LLM, vulnerabilidad, ciberseguridad**

I. Introducción

La seguridad en el desarrollo de *smart contracts* es un aspecto crítico debido a la naturaleza inmutable, transparente y descentralizada de las cadenas de bloques en las que se ejecutan. A diferencia del software tradicional, un contrato inteligente desplegado en una red pública no puede corregirse fácilmente mediante parches o actualizaciones rápidas, lo que convierte cualquier error en una vulnerabilidad potencialmente permanente. Esta característica, si bien aporta confianza y resistencia a la censura, exige un nivel de rigor especialmente elevado en su diseño, implementación y verificación. La ejecución automática y determinista de los *smart contracts* elimina la necesidad de intermediarios, pero también amplifica el impacto de cualquier fallo lógico, ya que las transacciones maliciosas o incorrectas pueden propagarse sin posibilidad de intervención humana directa.

La historia reciente de las tecnologías *Blockchain* demuestra que las vulnerabilidades en *smart contracts* pueden tener consecuencias económicas severas [1]. Uno de los casos más emblemáticos es el ataque a *The DAO* en Ethereum en 2016, donde un fallo de *reentrancy* permitió a un atacante drenar aproximadamente 60 millones de dólares y forzó a la comunidad de Ethereum a ejecutar un *hard fork*, originando así la división entre Ethereum y Ethereum *Classic*. Desde entonces, se han sucedido múltiples ataques que han evidenciado que los contratos que custodian grandes cantidades se convierten en foco de atacantes, convirtiendo cada vulnerabilidad en una pérdida económica directa y, a menudo, irreversible. La combinación de incentivos económicos, alta complejidad técnica y un ecosistema en rápida evolución convierte a los *smart contracts* en un objetivo atractivo para atacantes cada vez más sofisticados.

Por todo ello, el análisis y la auditoría de seguridad en *smart contracts* se han convertido en disciplinas fundamentales dentro del desarrollo *Blockchain* [2]. La descentralización que caracteriza a estas plataformas no reduce su superficie de ataque; al contrario, la exposición pública del código y la ejecución transparente hacen que cualquier error sea visible y explotable para adversarios globales. A medida que el valor total bloqueado en protocolos descentralizados continúa creciendo, también lo hace la necesidad de herramientas avanzadas que permitan detectar vulnerabilidades con mayor precisión y anticipación. Este contexto motiva la exploración de nuevas metodologías, entre ellas el uso de modelos de lenguaje de gran tamaño (LLMs), cuya capacidad para analizar patrones complejos en el código está transformando el campo del análisis de seguridad.

El uso de modelos de lenguaje de gran tamaño para el análisis de seguridad en software ha crecido de forma considerable en los últimos años. Diversos estudios muestran que los LLMs pueden complementar o incluso superar a técnicas clásicas de análisis estático cuando se diseñan *prompts*, contextos y verificaciones adecuados. En [3] se presenta un análisis del rendimiento de los LLMs en la detección de vulnerabilidades, incluyendo comparativas con modelos tradicionales y métricas específicas de precisión y sensibilidad, analizando además la influencia de aspectos poco estudiados como el desequilibrio entre clases y el efecto de la longitud del código. Asimismo, trabajos como *LLMVulExp* [4] demuestran que los LLMs pueden no solo identificar vulnerabilidades sino también generar explicaciones útiles para auditores humanos. En [5] se analizan técnicas para mejorar la efectividad de los LLMs mediante *prompts* conscientes del código y estrategias híbridas. En [6] se hace un repaso muy completo del uso de LLMs para detección de vulnerabilidades en código, haciendo hincapié también en las limitaciones actuales y oportunidades de trabajo futuro. Estos estudios demuestran que modelos avanzados como GPT-4o y Claude-3.5 Sonnet tienen un fuerte potencial en la detección de vulnerabilidades y que ajustes simples en los *prompts* pueden alterar significativamente los resultados.

En el ámbito específico de los *smart contracts*, los primeros trabajos emplearon modelos de aprendizaje automático y análisis estático avanzado, como *SmartCheck* [7], *SmartShield* [8] y *SoliAudit* [9], que sentaron las bases para evaluaciones automáticas en Solidity. Con la aparición de los LLMs, han surgido marcos especializados para auditoría semántica y

análisis multi-agente. En [10] se presenta un sistema multi-agente potenciado por LLMs que mejora la cobertura de vulnerabilidades y reduce falsos negativos mediante cooperación entre agentes especializados.

Aunque la literatura se ha centrado principalmente en el ecosistema Ethereum y el lenguaje de programación Solidity, comienzan a aparecer trabajos específicos para Solana [11]. Un estudio sistemático sobre problemas de seguridad en Solana [12], matiza que aún se encuentra por detrás de Ethereum en cuestión de análisis y herramientas de seguridad, añadiendo que el futuro apunta a la integración de IA y la adaptación de las técnicas utilizadas dentro del ecosistema Ethereum para fortalecer la seguridad de sus contratos inteligentes. Sin embargo, todavía existe una carencia de evaluaciones detalladas sobre el uso de LLMs aplicados a la semántica particular de Solana, donde la lógica propia del sistema, incluyendo la estructura de cuentas, la comunicación entre transacciones y programas, entre otros aspectos, introduce complejidades que no se presentan en otros entornos.

En este trabajo abordamos el análisis de vulnerabilidades en Solana mediante el uso de diferentes modelos de lenguaje, con el objetivo de evaluar su eficacia en la detección de fallos de seguridad y comparar su rendimiento entre sí. Para ello, se estudiará el lenguaje Rust, se analizarán vulnerabilidades reales y se revisarán las buenas prácticas para el desarrollo de contratos inteligentes en Solana. Además, se diseñarán *prompts* específicos para estudiar cómo influyen en las respuestas generadas por los distintos LLMs en este contexto.

II. Vulnerabilidades en Programas de Solana

El primer paso en este trabajo ha sido identificar un conjunto suficientemente representativo de vulnerabilidades comunes en los *Programs* de Solana, el equivalente a los *Smart Contracts* de Ethereum. Si bien los *Smart Contracts* de Ethereum están programados en Solidity, un lenguaje de alto nivel similar a similar a JavaScript, los programas de Solana se desarrollan principalmente en Rust, un lenguaje de bajo nivel que ofrece un control más explícito sobre la memoria y la ejecución. Cabe resaltar que, a partir de este momento, nos referiremos a los *Programs* de Solana simplemente como programa o contrato.

Se han identificado 8 categorías y 11 fuentes de riesgo que pueden permitir la explotación del código que constituye un programa en Solana.

Controles fundamentales sobre cuentas:

- *Owner check* (OWC): verifica que el propietario de la cuenta pasada al programa sea el correcto, es decir, el único autorizado a modificarla.
- *Signature check* (SNC): comprueba que el usuario que inicia la transacción haya firmado efectivamente con su clave privada.
- *PDA check* (PDAC): confirma que la cuenta PDA (*Program Derived Address*) suministrada es la esperada y no una cuenta distinta.

Controles aritméticos:

- Dado que Rust dispone de funciones para detectar errores aritméticos, por ejemplo *checked_add* y *checked_sub*, es importante comprobar si los modelos señalan su uso cuando se realizan sumas o restas directas. Se harán pruebas con el tipo *u64*. Las vulnerabilidades a analizar se denominaran *Overflow check* (OFC) y *Underflow check* (UFC).

Inicialización de cuentas:

- Se recomienda incluir un campo en la estructura de las nuevas cuentas que indique si están inicializadas (por ejemplo, un booleano) para evitar usos indebidos de cuentas sin inicializar. La vulnerabilidad a analizar se denominara *Account Initialization check* (AIC).

Confusión de cuentas:

- Dos tipos de cuentas con campos similares podrían intercambiarse accidentalmente en funciones. Rust no diferencia tipos estructurales idénticos, por lo que conviene añadir un campo de tipo o marcador que impida el uso indebido. La vulnerabilidad a analizar se denominara *Account confusion* (AC).

Invocación a programas externos:

- Verificar que las llamadas a programas externos (*Cross Program Invokation*) correspondan efectivamente al programa esperado y no a uno malicioso que suplante su identidad. La vulnerabilidad a analizar se denominara *CPI check* (CPIC).

Privilegios de administrador:

- Ciertas operaciones deben limitarse a cuentas administrativas, los modelos deberían detectar si la cuenta que invoca la función posee los permisos adecuados. La vulnerabilidad a analizar se denominara *Admin Privileges check* (APC).

***Bump* vulnerable:**

- Una cuenta PDA en Solana permite que un programa controle cuentas y datos de forma segura sin usar una clave privada. Se utiliza para guardar estado, manejar fondos o actuar como autoridad, garantizando que solo el programa pueda modificar la cuenta. Por ejemplo, un programa puede crear una PDA para almacenar y actualizar información asociada a un usuario.
  Una PDA se deriva del *ID* del programa y unas entradas llamadas *seeds*, junto con un valor adicional llamado *bump*, que asegura que la dirección no tenga clave privada. Para evitar riesgos de seguridad, se recomienda usar la función *find_program_address*, que genera automáticamente un *bump* único y seguro, en lugar de permitir que el usuario lo controle. En cambio, la función *create_program_address* permite introducir el *bump* de manera manual, lo que puede dar lugar a explotaciones si no se valida correctamente. La vulnerabilidad analizada en este contexto se denomina *Bump Vulnerable Check* (BVC).

Uso de funciones obsoletas:

- Un *exploit* permitió acuñar 120 wETH en Solana por el uso de funciones obsoletas como *load_current_index* y *load_instruction_at*, que permiten acceder a instrucciones previas de una transacción [13]. Estas funciones no verificaban que se hubiera pasado la cuenta correcta *sysvar_instructions*, permitiendo inyectar un *payload* malicioso y evadir controles a la hora de acuñar *Tokens*. La vulnerabilidad a analizar se denominara *Load Funtion check* (LFC).

III. Diseño de programas vulnerables

Para realizar las pruebas con los LLMs, se han diseñado dos programas con funcionalidades diferenciadas, para poder integrar en cada uno de ellos vulnerabilidades específicas. Esto nos ha permitido generar una familia de programas "equivalentes" donde se han ido introduciendo diversas vulnerabilidades de forma que se pueda medir la precisión de los LLMs al detectar las vulnerabilidades introducidas.

III-A. Programa Wallet

Este programa crea y gestiona cuentas *wallets* derivadas (PDAs) asociadas a usuarios, permitiendo depositar, retirar y transferir fondos entre ellas. La Figura 1 muestra su diagrama de casos de uso. El programa contiene cuatro funciones clave:

- initialize_wallet_account (IWA): crea la cuenta *wallet* con saldo 0 y marca el campo *initialized* como *true*.
- deposit_amount (DA): aumenta el saldo.
- withdraw_amount (WA): reduce el saldo.
- transfer_amount (TA): mueve fondos entre *wallets*.

Las vulnerabilidades de cada función se recogen en la Tabla I.

Tabla I: Vulnerabilidades presentes en el programa Wallet

Función	OWC	SNC	PDAC	AIC	OFC	UFC
IWA	0	1	1	0	0	0
DA	1	1	1	1	1	0
WA	1	1	1	1	0	1
TA	1	1	1	1	1	1
Total	3	4	4	3	2	2

Cabe resaltar que el entorno es experimental, modifica saldos directamente para probar controles aritméticos, mientras que en producción el manejo de tokens debería realizarse con el programa *Token* de Solana.

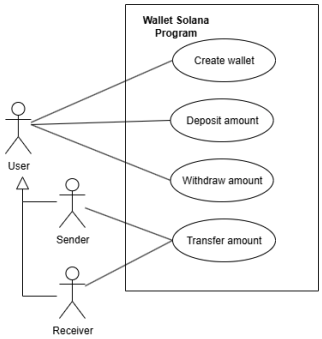


Figura 1: Caso de uso del programa Wallet

III-B. Programa Voting

Programa que simula un DAO (*Decentralized Autonomous Organization*) en Solana para gestionar propuestas, votaciones y acciones para depositar y retirar fondos en un *Vault* comunitario. La Figura 2 muestra su diagrama de casos de uso.

Las funciones clave son las siguientes:

- create_admin (CAD): creación de cuenta *admin*, es única en todo el programa, tiene un usuario asociado.
- update_admin (UAD): actualiza el usuario asociado a la cuenta *admin*.
- create_account (CAC): creación de cuenta de usuario, tiene asociado unos puntos de gobernanza que constituyen el peso de su voto.
- open_proposal (OP): Abrir propuesta
- vote_process (VP): Votar propuesta, por cada voto se crea una cuenta derivada con el identificador del voto y el peso de este.
- manage_governance_points (MGP): gestión de puntos de gobernanza de los usuarios
- execute_instruction (EI): ejecución de CPIs hacia el programa externo que maneja el *Vault* comunitario
- make_decision (MD): toma de decisión final

Para comprender mejor el funcionamiento del programa, se exponen unos puntos clave de su funcionamiento:

- Los usuarios crean una cuenta en el DAO para proponer, votar y depositar, recibiendo 10 puntos de gobernanza iniciales.
- Una cuenta de administrador única gestiona puntos, toma decisiones finales y controla el *Vault*. Los usuarios normales solo pueden depositar.
- Se incorporan controles anti-doble voto, creando cuentas derivadas que registran el peso y el ID del voto.
- El sistema permite abuso por confusión de cuentas, usando votos de cuentas ajenas durante la votación.
- Se incluyen llamadas a un programa externo para el *Vault*, generando vulnerabilidad de CPI, además de instrucciones obsoletas y uso de *sysvar* (relevante en *Wormhole*).
- La vulnerabilidad del parámetro *bump* se explota reemplazando *find_program_address* por *create_program_address*, dando control total al usuario.

Las vulnerabilidades de cada función se recogen en la Tabla II.

Tabla II: Vulnerabilidades presentes en el programa Voting

Función	BVC	APC	AC	CPIC	OFC	UFC	LFC
CAD	1	0	0	1	0	0	0
UAD	1	1	0	1	0	0	0
CAC	1	0	0	0	0	0	0
OP	1	0	1	1	0	0	0
VP	1	0	1	1	1	0	0
MGP	1	1	1	0	1	1	0
EI	1	1	0	1	1	0	1
MD	1	1	0	0	1	0	0
Total	8	4	3	7	2	1	1

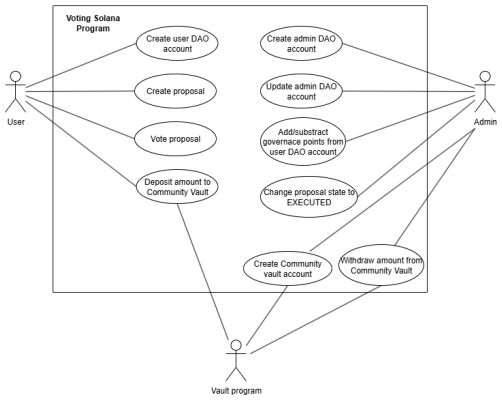


Figura 2: Caso de uso del programa Voting

IV. Selección de modelos, recopilación de datos y diseño de Prompts

El estudio se ha estructurado en tres fases: la selección de los modelos de lenguaje a evaluar, la generación del conjunto de datos y de los *prompts*, y la definición de las métricas empleadas. Esta estructura permite comparar de manera controlada el rendimiento de los modelos en la detección de vulnerabilidades en programas de Solana desarrollados en Rust.

IV-A. Selección de LLMs

Para comparar los modelos de lenguaje, se han analizado los resultados en diversos *benchmarks* oficiales, ya que no existe uno específico para la detección de vulnerabilidades en código Rust para Solana. Se han empleado principalmente los *benchmarks* SWE-Bench, SWE-Bench Verified y SWE-Bench Multilingual, centrados en tareas de ingeniería de software, junto con Aider Polyglot (que incluye Rust), LiveCodeBench (resolución de problemas en Python), IFEval (seguimiento de instrucciones) y MMLU-Pro (razonamiento y comprensión general).

Se han evaluado cuatro modelos:

- **GPT Nano 4.1**, versión ligera y rápida enfocada en tareas de codificación e instrucciones.
- **Gemini Flash 2.0**, modelo multimodal con alta velocidad y gran ventana de contexto.
- **DeepSeek V3.1**, basado en arquitectura Mixture of Experts con modos rápido y de razonamiento.
- **Qwen3 235B A22B**, modelo eficiente de Alibaba que activa solo parte de sus parámetros.

Todos los modelos se han probado con las configuraciones por defecto, priorizando opciones económicas para evaluar su rendimiento mediante *prompt tuning* sin activar modos de razonamiento.

IV-B. Generación de datos

Se construyeron dos conjuntos de datos principales, uno por cada programa desarrollado (*Wallet* y *Voting*). Cada uno incluye múltiples versiones del programa vulnerable. A partir de estos datos se obtendrán las respuestas según el modelo y el *prompt* seleccionado. La metodología para generar los datos fue la siguiente:

- **Identificación de vulnerabilidades**: Se contabilizaron 6 tipos de vulnerabilidades por programa y se generaron todas las combinaciones posibles sin repeticiones. De esta manera se generan contratos con la misma funcionalidad pero con una cantidad variable de vulnerabilidades no controladas. En total se obtuvieron 63 contratos diferentes para cada programa.
- **Construcción de programas derivados**: Mediante *scripts* en Python, cada programa se modificó según la vulnerabilidad seleccionada:
  - Eliminación de mitigaciones.
  - Sustitución de código seguro por vulnerable.
  - Omisión de fragmentos que previenen vulnerabilidades (ej. *account confusion*).
- **Diseño de *prompts***: Se utilizó Python para estructurar los *prompts* en formato JSON, con *system_message* (instrucciones al modelo) y *user_message* (pregunta). Se crearon versiones básicas y complejas para evaluar el efecto de la especificación de la instrucción.
- **Obtención de respuestas**: Cada modelo recibió solicitudes independientes para cada combinación de programa y *prompt*. Se almacenaron las respuestas y se registraron los tiempos de procesamiento. En total, se obtuvieron 16 muestras y 1008 respuestas. En la Tabla III y IV se muestran los tiempos totales, en segundos, correspondientes al procesamiento de las respuestas generadas por cada modelo.
  En la Tabla V y VI se presenta el coste total asociado al procesamiento de las respuestas de cada modelo por cada 1 millón de *tokens* ingeridos.

Tabla III: Tiempo total de procesamiento respuestas en segundos (*Wallet*)

Modelo	Prompt Básico	Prompt Complejo
GPT Nano 4.1	330.02	368.55
Gemini Flash	381.52	256.68
DeepSeek V3.1	1438.75	1420.24
Qwen3	1361.80	1146.80

Tabla IV: Tiempo total de procesamiento respuestas en segundos (*Voting*)

Modelo	Prompt Básico	Prompt Complejo
GPT Nano 4.1	593.16	541.26
Gemini Flash	511.78	473.54
DeepSeek V3.1	2119.59	-
Qwen3	2144.87	2441.03

Tabla V: Precio por cada 1M de tokens en $ (*Wallet*)

Modelo	Prompt Básico	Prompt Complejo
GPT Nano 4.1	0.023	0.024
Gemini Flash	0.026	0.027
DeepSeek V3.1	0.043	0.044
Qwen3	0.040	0.042

Tabla VI: Precio por cada 1M de tokens en $ (*Voting*)

Modelo	Prompt Básico	Prompt Complejo
GPT Nano 4.1	0.050	0.051
Gemini Flash	0.060	0.061
DeepSeek V3.1	0.063	0.064
Qwen3	0.093	0.094

IV-C. Criterios de evaluación y metodología del análisis

Para analizar las respuestas de los modelos se utilizó una tabla estructurada por columnas que incluyen el identificador de cada programa, las vulnerabilidades reales (con sufijo *Real*) y las vulnerabilidades detectadas por los modelos (con prefijo *Predicted*). Las columnas cuantifican cuántas veces aparece cada vulnerabilidad. La Tabla VII muestra un ejemplo de como esta constituida.

Tabla VII: Tabla ejemplo de evaluación de muestras

ID Programa	**Vuln_1l Real**	**Vuln_X Real**	**Vuln_1 Predicted**	**Vuln_X Predcited**
1	3	0	1	2
X	0	2	0	2

La detección se considera correcta si:

- Mención en la función - la vulnerabilidad se identifica explícitamente o se indica la falta de un control.
- Referencia global - la vulnerabilidad se menciona de forma explícita a nivel general.

Para vulnerabilidades más específicas se usan además los siguientes criterios:

- *Bump* vulnerable: se considera detectada si se menciona riesgo por control de *bump* o uso inseguro de *create_program_address*.
- *load_function_at*: se considera detectada si se indica que la función es insegura/deprecada o que la cuenta *sysvar instruction* puede ser manipulada.

V. Análisis de los resultados

Esta sección se centra en el análisis de las respuestas obtenidas por los distintos modelos y, además, compara en cada caso la influencia del *prompt* empleado. Para evaluar la capacidad de los modelos en detectar la presencia de una vulnerabilidad específica en un contrato, se han utilizado las medidas de **precisión**, **sensibilidad** y **especificidad**, obtenidas a partir de la matriz de confusión correspondiente a cada vulnerabilidad. Para este análisis, únicamente se considera si la vulnerabilidad está presente, sin tener en cuenta cuántas veces aparece ni en cuántas funciones del contrato.

El segundo análisis se enfoca en los errores cometidos por los modelos, incluyendo la comparación del efecto del *prompt* al intentar detectar múltiples ocurrencias de distintas vulnerabilidades en un mismo contrato. Para ello, se emplean las métricas **MAE**, **RMSE** y el **sesgo**.

Se han generado numerosos datos y obtenido múltiples gráficas que muestran los resultados estadísticos del estudio. El análisis estadístico detallado y el conjunto completo de gráficos pueden consultarse en el Trabajo Fin de Grado de la autora, accesible públicamente en su repositorio de GitHub [14]. Con el fin de mejorar la claridad y facilitar la interpretación, a continuación se presentan los puntos más relevantes identificados:

- **Qwen3 destaca claramente entre todos los modelos**. En todos los casos obtiene las mejores métricas y comete los menores errores incluso utilizando un *prompt* simple. Al añadir un *prompt* más específico, sus métricas mejoran aún más, consolidándose como el modelo más fiable en todas las tareas evaluadas. En las Tablas VIII y IX se muestra de manera general la tendencia del comportamiento de los modelos frente a todas las vulnerabilidades del programa *Wallet* con *prompt* simple y complejo.
- **Gemini resulta ser el menos recomendable para estas tareas**. Generalmente obtiene valores muy bajos en la mayoría de las métricas y tiende a ser el modelo que comete más errores. Además, con un *prompt* complejo sus resultados empeoran ligeramente para las vulnerbilidades SNC, PDAC Y AIC, lo que indica que, para este modelo, la influencia del *prompt* puede no aportar mejoras. Completamente el caso contrario que para los modelos GPT Nano 4.1 y DeepSeek V3.1 cuyas métricas con un *prompt* simple son muy bajas y con un *prompt* complejo mejoran notablemente.
- **La especificidad no es una métrica fiable**. En las respuestas del programa *Wallet*, la especificidad es siempre muy alta (cercana al 100 %), pero contrasta con una sensibilidad muy baja, inferior al 50 % en algunos casos. Esto demuestra que la especificidad, por sí sola, no es una métrica fiable para evaluar estos modelos, ya que únicamente indica que son buenos identificando la ausencia de vulnerabilidades, pero no detectando las que sí existen. Por ello, no puede utilizarse como criterio principal para determinar qué modelo es mejor.
- **En las respuestas del programa *Voting*, la especificidad tiende a disminuir**. Esta reducción es aún mayor cuando se utiliza un *prompt* complejo. Aunque sería necesario realizar más pruebas para confirmarlo, una posible hipótesis es que, ante un *prompt* muy detallado y un contrato complejo con mucha lógica, los modelos se vuelven más "paranoicos", señalando vulnerabilidades incluso cuando no las hay.
- **Las vulnerabilidades más fáciles de detectar son *Overflow* y *Underflow***. No todos los modelos identifican con facilidad vulnerabilidades propias del entorno Solana (por ejemplo OWC, SGC, PDAC, etc.), Qwen es la excepción, mientras que el resto presenta una sensibilidad y precisión bajas en estos casos. Sin embargo, es cierto que para un contrato simple como es *Wallet*, el *prompt* complejo mejora notablemente las métricas de GPT 4.1 y DeepSeek V3.1.
- **La vulnerabilidad de Confusión de cuentas no ha sido detectada en ningún caso**. Esto podría deberse a que se trata de un concepto relativamente abstracto y a que los modelos no parecen reconocer que el lenguaje Rust presenta este tipo de riesgo.
- **En general, los contratos más complejos provocan más fallos en los modelos**. Cuanta más lógica y complejidad presenta el contrato, mayor es la tasa de error y peor las métricas de precisión y sensibilidad en la detección de cualquier tipo de vulnerabilidad. Esto se puede observar en las Tablas X y XI que muestran de manera general la tendencia del comportamiento de los modelos frente a todas las vulnerabilidades del programa *Voting* con *prompt* simple y complejo. Esto sugiere que, cuanto mayor es la complejidad del contrato, menor es la capacidad del *prompt* para influir positivamente en la detección de vulnerabilidades.

Tabla VIII: Global (*Wallet simple*)

Modelo	Precisión	Sensibilidad	Especificidad
GPT Nano 4.1	0.75	0.51	1.0
Gemini Flash	0.69	0.39	0.99
DeepSeek V3.1	0.72	0.45	1.0
Qwen3	0.95	0.91	0.99

Tabla IX: Global (*Wallet complejo*)

Modelo	Precisión	Sensibilidad	Especificidad
GPT Nano 4.1	0.86	0.75	0.98
Gemini Flash	0.71	0.43	1.0
DeepSeek V3.1	0.90	0.81	0.99
Qwen3	0.98	0.97	0.98

Tabla X: Global (*Voting simple*)

Modelo	Precisión	Sensibilidad	Especificidad
GPT Nano 4.1	0.58	0.28	0.88
Gemini Flash 2.0	0.52	0.062	0.99
DeepSeek V3.1	0.54	0.10	0.98
Qwen3	0.75	0.52	0.98

Tabla XI: Global (*Voting complejo*)

Modelo	Precisión	Sensibilidad	Especificidad
GPT Nano 4.1	0.70	0.56	0.84
Gemini Flash 2.0	0.64	0.30	0.99
DeepSeek V3.1	0.81	0.73	0.89
Qwen3	0.80	0.66	0.95

VI. Conclusiones

En relación con los modelos usados, se concluye que el modelo Gemini Flash 2.0, a pesar de su rapidez y bajo coste, debería descartarse para este tipo de análisis debido a su bajo rendimiento y poca capacidad de mejora. Los modelos Deepseek V3.1 y GPT-Nano 4.1 han demostrado poder mejorar sus análisis afinando el *prompt* inicial. Sin embargo, siguen obteniendo valores bajos en la sensibilidad para controles fundamentales como PDA o SGC, por lo que no son fiables para realizar estas tareas sin estar supervisados. Finalmente, Qwen3 se posiciona como la opción más fiable y precisa especialmente en el análisis de programas muy simples, como fue el caso del programa *Wallet*.

Los resultados obtenidos evidencian que modelos como Gemini Flash 2.0, optimizados para una respuesta rápida, grandes ventanas de contexto y bajo coste computacional, presentan limitaciones significativas en tareas que requieren razonamiento complejo.

Por el contrario, Qwen3 destaca por una arquitectura orientada a la profundidad analítica y a la calidad del razonamiento. Aunque dispone de 235B de parámetros totales, solo activa 22B por token durante la generación, logrando así un equilibrio eficiente entre capacidad computacional y rendimiento. Todo esto se traduce en resultados superiores frente al resto de los modelos evaluados.

En consecuencia, esta comparativa pone de manifiesto que los modelos ligeros y orientados a la velocidad no resultan adecuados para problemas de alta complejidad. Para este tipo de tareas, es necesario priorizar arquitecturas especializadas, aun a costa de un mayor consumo de recursos y tiempos de respuesta.

Otro aspecto relevante es la diferencia en el desempeño de los modelos según el programa analizado. En el caso del programa *Voting*, cuya estructura es más extensa y compleja y en el que las vulnerabilidades se repiten en múltiples secciones, los modelos suelen tener dificultades para identificarlas correctamente. En algunos casos, incluso llegan a alucinar o inferir problemas inexistentes, mientras que en otros no detectan vulnerabilidades reales, como la confusión de cuentas. Esta limitación se refleja de forma evidente en los valores más bajos obtenidos en comparación con los resultados del programa *Wallet*.

Las únicas vulnerabilidades que mostraron ser bien identificadas en prácticamente todos los casos por los modelos eran las relacionadas con controles aritméticos, esto podría deberse a que este tipo de errores también se presentan en otros lenguajes de programación, por lo que resulta lógico que los modelos estén mejor entrenados para reconocerlos.

Todo esto nos indica que estos modelos aun no están preparados para poder ser usados en lógicas muy complejas ni para analizar tipos de vulnerabilidades específicos de Solana.

Futuras investigaciones por esta misma línea, podrían explorar ajustes de parámetros como la temperatura, el número máximo de tokens de salida de las respuestas, la redacción de prompts, así como el uso de modelos más potentes. O también se podrían implementar técnicas avanzadas como RAGs o Swarms, para mejorar modelos como Qwen3 el cual ha mostrado ser bastante prometedor.

Referencias

[1] S. Sayeed, H. Marco-Gisbert, and T. Caira, "Smart contract: Attacks and protections," *IEEE Access*, vol. 8, pp. 24 416–24 427, 2020.

[2] S. Chaliasos, M. A. Charalambous, L. Zhou, R. Galanopoulou, A. Gervais, D. Mitropoulos, and B. Livshits, "Smart contract and defi security tools: Do they meet the needs of practitioners?" in *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*, ser. ICSE '24. New York, NY, USA: Association for Computing Machinery, 2024. [Online]. Available: https://doi.org/10.1145/3597503.3623302

[3] X. Zhou, T. Zhang, and D. Lo, "Large language model for vulnerability detection: Emerging results and future directions," in *ACM/IEEE 44th International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER'24)*, 2024, p. 47–51. [Online]. Available: https://doi.org/10.1145/3639476.3639762

[4] Q. Mao, Z. Li, X. Hu, K. Liu, X. Xia, and J. Sun, "Towards explainable vulnerability detection with large language models," vol. 51, no. 10, pp. 2957–2971, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2406.09701

[5] R. Feng, H. Pearce, P. Liguori, and Y. Sui, "Cgp-tuning: Structure-aware soft prompt tuning for code vulnerability detection," *IEEE Transactions on Software Engineering*, vol. 51, no. 9, pp. 2533–2548, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2501.04510

[6] X. Zhou, S. Cao, X. Sun, and D. Lo, "Large language model for vulnerability detection and repair: Literature review and the road ahead," *ACM Transactions on Software Engineering Methodology*, vol. 34, no. 5, p. 145, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2404.02525

[7] S. Tikhomirov, E. Voskresenskaya, I. Ivanitskiy, R. Takhaviev, E. Marchenko, and Y. Alexandrov, "Smartcheck: Static analysis of ethereum smart contracts," in *Proc. IEEE/ACM Int. Workshop Emerging Trends Softw. Eng. Blockchain (WETSEB)*, Gothenburg, Sweden, 2018, pp. 9–16.

[8] Y. Zhang, S. Ma, J. Li, K. Li, S. Nepal, and D. Gu, "Smartshield: Automatic smart contract protection made easy," in *Proc. IEEE Int. Conf. Softw. Anal., Evol. Reeng. (SANER)*, London, ON, Canada, 2020, pp. 23–34.

[9] J.-W. Liao, T.-T. Tsai, C.-K. He, and C.-W. Tien, "Soliaudit: Smart contract vulnerability assessment based on machine learning and fuzz testing," in *Proc. Int. Conf. Internet Things: Syst., Manage. Secur. (IOTSMS)*, Granada, Spain, 2019, pp. 458–465.

[10] Z. Wei *et al.*, "Advanced smart contract vulnerability detection via llm-powered multi-agent systems," *IEEE Transactions on Software Engineering*, vol. 51, no. 10, pp. 2830–2846, 2025.

[11] S. Cui, G. Zhao, Y. Gao, T. Tavu, and J. Huang, "Vrust: Automated vulnerability detection for solana smart contracts," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, 2022, pp. 639–652.

[12] X. Wu, J. Xing, and X. Li, "Exploring vulnerabilities and concerns in solana smart contracts," *arXiv preprint arXiv:2504.07419*, 2025. [Online]. Available: https://doi.org/10.48550/arXiv.2504.07419

[13] CertiK, "Wormhole bridge exploit incident analysis," https://www.certik.com/resources/blog/wormhole-bridge-exploit-incident-analysis, 2022.

[14] L. D. Haro, "Uso de llms para el análisis de vulnerabilidades en solana programs," https://github.com/LauraDhg/Trabajo-Fin-de-Grado, 2025.

Hacia nuevas formas de componer aleatoriamente en privacidad diferencial

Àlex Miranda-Pascual
Universitat Politècnica de Catalunya
Karlsruhe Institute of Technology
alex.miranda.pascual@upc.edu

Javier Parra-Arnau
Universitat Politècnica de Catalunya
Barcelona, España
javier.parra@upc.edu

Thorsten Strufe
Karlsruhe Institute of Technology
Karlsruhe, Alemania
thorsten.strufe@kit.edu

***Resumen*—La composición es una propiedad clave de la privacidad diferencial (DP) que garantiza que la combinación de mecanismos DP sea DP. En este artículo, presentamos una nueva estrategia de composición para (ε, δ)-DP: la *composición paralela probabilística*. En este escenario, los mecanismos reciben como entrada una partición disjunta de la base de datos, como en la composición paralela, pero cada registro se asigna aleatoriamente en lugar de determinísticamente. Proponemos dos teoremas cubriendo diferentes maneras de selección aleatoria: el primero, para DP *unbounded*, introduce cada registro en una única entrada siguiendo una distribución fija; mientras que en el segundo, para DP *bounded*, las entradas son particiones de tamaño fijo seleccionadas uniformemente. Nuestros métodos de composición aumentan la privacidad al añadir incertidumbre y ofrecen parámetros de privacidad menores que los de las composiciones paralela y secuencial clásicas, pero tienen menor aplicabilidad. No obstante, creemos que estas técnicas pueden servir para diseñar nuevos mecanismos DP.**

I. Introducción

En los últimos años, la privacidad diferencial (DP[1], por sus siglas en inglés) [1], [2] se ha consolidado como el estándar de privacidad debido a sus sólidas garantías de privacidad y su robusta formulación matemática. Entre sus muchas propiedades útiles, destaca la *composición* como herramienta para combinar y crear nuevos mecanismos DP. Los teoremas de composición permiten combinar un número finito de mecanismos DP $\mathcal{M}_i$ para obtener un nuevo mecanismo DP $\mathcal{M}$ y proporcionan una cota de los parámetros de privacidad de $\mathcal{M}$, midiendo así la pérdida de privacidad. La primera y más amplia estrategia, la *composición secuencial* [2], permite componer cualquier número finito de mecanismos DP arbitrarios, obteniendo una cota de los parámetros de privacidad de $\mathcal{M}$ equivalente a la suma de los parámetros de cada $\mathcal{M}_i$. En este caso, la privacidad se degrada linealmente.

Un caso interesante que reduce las cotas de los parámetros de privacidad de $\mathcal{M}$ es la *composición paralela* [3], que los acota por el máximo de los parámetros de $\mathcal{M}_i$ en lugar de por la suma. Sin embargo, la composición paralela impone condiciones adicionales sobre los mecanismos $\mathcal{M}_i$: sus entradas tienen que estar definidas como una partición disjunta del universo de datos (véase el Teorema II.2). A pesar de tener una aplicabilidad más limitada, la composición paralela sigue siendo ampliamente utilizada en DP.

En este trabajo, vamos un paso más allá y proporcionamos nuevos resultados de composición que reducen aún más estas cotas. Esta nueva estrategia de composición, la *composición paralela probabilística*, realiza particiones aleatorias de las bases de datos de entrada de $\mathcal{M}_i$ en lugar de particiones deterministas como en el caso clásico de la composición paralela. Presentamos dos teoremas que permiten dos tipos de particiones aleatorias, ambas proporcionando cotas menores que el máximo dado por la composición paralela.

En el primer teorema (III.1), definido para (ε, δ)-DP *unbounded*, cada registro se incluye en una única entrada siguiendo una distribución fija, es decir, se incluye en la entrada de $\mathcal{M}_i$ con probabilidad p_i (con $p_1 + \cdots + p_k = 1$). La cota resultante corresponde a una media ponderada de los parámetros de privacidad de $\mathcal{M}_i$ con peso p_i.

El segundo teorema (III.2) se define para tratar el caso donde cada mecanismo $\mathcal{M}_i$ debe tomar un número específico n_i de registros como entrada. En este caso, la partición disjunta se selecciona uniformemente entre todas las particiones de la base de datos que aseguren que las entradas de $\mathcal{M}_i$ tengan n_i registros. Dado que los tamaños n_i de las entradas están fijados bajo esta estrategia, el teorema se enuncia para (ε, δ)-DP *bounded* en lugar de *unbounded*. Curiosamente, nuestro resultado de composición paralela probabilística siempre garantiza un mecanismo DP *bounded* y evita el problema de la composición paralela para DP *bounded* que nos dice que no se puede garantizar que la composición paralela de mecanismos DP *bounded* sea DP *bounded* [4], [5]. Nuestras cotas de los parámetros de $\mathcal{M}$ son más abstractas que las del primer teorema, pero siguen siendo inferiores a las de la composición paralela en DP *unbounded*, y a las de la solución propuesta [5] para la composición paralela en DP *bounded*.

Además, nuestros teoremas sirven como generalizaciones de la *amplificación de privacidad mediante muestreo de Poisson* [6], [7] y *mediante muestreo sin reemplazo* [7], [8], respectivamente. En este sentido, nuestros teoremas disfrutan de un beneficio similar al principio de «amplificación de privacidad mediante muestreo» [9], [6], ya que disminuyen la cota de los parámetros de privacidad de $\mathcal{M}$ y se gana incertidumbre sobre qué registro se introduce en qué mecanismo.

En resumen, nuestras contribuciones son las siguientes:

- Hasta donde sabemos, somos los primeros en introducir una composición (paralela) probabilística en DP. Para nuestra composición, los mecanismos deben tomar como entrada una partición aleatoria de la base de datos.
- Introducimos dos teoremas que acotan los parámetros de privacidad del mecanismo compuesto para dos tipos de partición aleatoria diferentes. Nuestros teoremas proporcionan cotas menores que la cota máxima ofrecida por el teorema clásico de composición paralela.

[1]También utilizamos «DP» con el significado de «diferencialmente privado», que es usado para calificar a un mecanismo que satisface la definición de la DP, por analogía con su uso en inglés.

II. Preliminares

II-A. Privacidad diferencial

La privacidad diferencial (DP) [1] es una noción popular de privacidad que se usa con el objetivo de ocultar la presencia o ausencia de un individuo en una base de datos, garantizando así que un analista pueda seguir extrayendo información global sobre la base de datos, mientras que un adversario solo pueda obtener una cantidad limitada de información sobre cada individuo. Formalmente,

Definición II.1 ((ε, δ)-privacidad diferencial [10], [2])**.** Sean $\varepsilon, \delta \geq 0$ y $\mathbb{D}$ una clase de bases de datos. Un mecanismo aleatorio $\mathcal{M}$ con dominio $\mathbb{D}$ es (ε, δ)-*DP (DP aproximada)* si para todas las bases de datos adyacentes $D, D' \in \mathbb{D}$ y para todos los subconjuntos medibles $S \subseteq \mathrm{Range}(\mathcal{M})$,

$$\mathrm{P}\{\mathcal{M}(D) \in S\} \leq \mathrm{e}^{\varepsilon}\, \mathrm{P}\{\mathcal{M}(D') \in S\} + \delta.$$

Si $\delta = 0$, decimos que $\mathcal{M}$ es ε-*DP (DP pura)*.

En este artículo, consideraremos dos tipos de adyacencia: *unbounded* y *bounded* [11], que son las más populares en la literatura. Se dice que dos bases de datos son *unbounded-adyacentes* si se obtiene una a partir de la otra añadiendo o eliminando un registro, y se dice que son *bounded-adyacentes* si se obtiene una a partir de la otra cambiando un solo registro. Cada definición de adyacencia ofrece una variación diferente de la DP: en nuestro caso, *DP unbounded* y *DP bounded*.

El nivel de privacidad en DP viene dado por sus dos *parámetros de privacidad*, ε y δ, que corresponden a cotas superiores de la pérdida de privacidad. Intuitivamente, cuanto menores sean los valores, mayor será la privacidad. Además, según la definición de la DP, si $\mathcal{M}$ es (ε, δ)-DP, entonces también es (ε', δ')-DP para todo $\varepsilon \leq \varepsilon'$ y $\delta \leq \delta'$. Por lo tanto, para proporcionar una intuición precisa sobre el nivel (o pérdida) de privacidad de los mecanismos, es conveniente dar cotas *estrictas* sobre los parámetros de privacidad siempre que sea posible; es decir, obtener valores ε y δ tales que no existan $\varepsilon'' \leq \varepsilon$ y $\delta'' \leq \delta$ (no ambos iguales) que verifiquen que $\mathcal{M}$ es también $(\varepsilon'', \delta'')$-DP. Cabe señalar que ε y δ se compensan mutuamente: al disminuir ε podemos aumentar δ, y viceversa. Esto nos permite obtener diferentes cotas estrictas para un mismo mecanismo (véase el «perfil de privacidad» [7]).

II-B. Composición paralela

Una de las propiedades más importantes en DP es la *composición*, que permite combinar múltiples mecanismos DP para obtener un nuevo mecanismo DP y dar sus parámetros de privacidad. La *composición secuencial* permite combinar k mecanismos con un mismo dominio $\mathbb{D}$, cada uno $(\varepsilon_i, \delta_i)$-DP, para obtener un mecanismo $(\sum_{i=1}^{k} \varepsilon_i, \sum_{i=1}^{k} \delta_i)$-DP [2]. Sin embargo, en el caso particular donde el dominio de los mecanismos corresponde a una partición disjunta del dominio inicial, obtenemos que el mecanismo compuesto es $(\max_{i\in[k]} \varepsilon_i, \max_{i\in[k]} \delta_i)$-DP. Esta estrategia se conoce como *composición paralela* [3] y se puede formalizar para (ε, δ)-DP de la siguiente manera:

Teorema II.2 (Composición paralela [3, reformulación])**.** *Sean $\mathcal{X}$ un conjunto de registros y $\{\mathcal{X}_1, \ldots, \mathcal{X}_k\}$ una partición de $\mathcal{X}$. Para cada $i \in [k]$, sea $\mathcal{M}_i$ $(\varepsilon_i, \delta_i)$-DP unbounded con dominio $\mathbb{D}_{\mathcal{X}_i}$, las bases de datos extraídas de $\mathcal{X}_i$. Entonces, $\mathcal{M} = (\mathcal{M}_1, \ldots, \mathcal{M}_k)$ es $(\max_{i\in[k]} \varepsilon_i, \max_{i\in[k]} \delta_i)$-DP unbounded, donde $\mathcal{M}$ está definido para toda $D \in \mathbb{D}$ como $\mathcal{M}(D) = (\mathcal{M}_1(D_1), \ldots, \mathcal{M}_k(D_k))$ y D_i es el multiconjunto que contiene todo $x \in D$ con multiplicidad $m_{D_i}(x) = \mathbf{1}_{\mathcal{X}_i}(x)\, m_D(x)$.*

Es importante destacar que el teorema anterior solo es válido para DP *unbounded* y no para DP *bounded* [5], [4], a diferencia de la composición secuencial, que es válida para ambas. Para DP *bounded*, la composición paralela podría devolver un mecanismo que no es DP *bounded* [5]. No obstante, existe una configuración alternativa, denominada «dominio común» [5], en la que los mecanismos $\mathcal{M}_i$ se definen directamente sobre el mismo dominio que $\mathcal{M}$ (es decir, $\mathbb{D}_{\mathcal{X}}$). Esta configuración nos permite componer k mecanismos ε_i-DP *bounded* con entradas disjuntas, obteniendo que $\mathcal{M}$ es $(\max_{i,j\in[k],\, i\neq j}(\varepsilon_i + \varepsilon_j))$-DP *bounded* [5]. Se puede derivar un resultado análogo para DP aproximada: al componer k mecanismos $(\varepsilon_i, \delta_i)$-DP *bounded*, obtenemos un mecanismo $(\max_{i,j\in[k],\, i\neq j}(\varepsilon_i + \varepsilon_j), \max_{i,j\in[k],\, i\neq j}(\delta_i + \delta_j))$-DP *bounded* (esta es una cota trivial). Obsérvese que las expresiones de los parámetros consisten en máximos sobre dos índices diferentes, lo que se debe a que cambiar un registro implica cambiar las entradas de, como máximo, dos mecanismos $\mathcal{M}_i$.

II-C. La amplificación de privacidad mediante muestreo

Una técnica bastante conocida para reducir los parámetros de privacidad de un mecanismo DP es el *muestreo*. El muestreo se modela como una función aleatoria $\mathcal{S}$ sobre $\mathbb{D}$ tal que $\mathcal{S}(D)$ genera subconjuntos de D (es decir, muestras) para todos los $D \in \mathbb{D}$ [7]. Las estrategias de muestreo más conocidas son el muestreo de Poisson y el muestreo sin reemplazo (SWOR, por sus siglas en inglés), que mejoran los parámetros cuando se aplican a mecanismos DP *unbounded* y DP *bounded* arbitrarios, respectivamente:

Teorema II.3 (Amplificación de privacidad: Poisson [6], [7], [8])**.** *Sea $\mathbb{D}$ una clase de bases de datos cerrada por subconjuntos, y sea $\mathcal{S}_{\mathrm{Poiss}}$ un muestreo de Poisson sobre $\mathbb{D}$, i.e., donde se incluye cada registro $x \in D$ en $\mathcal{S}_{\mathrm{Poiss}}(D)$ con probabilidad p. Luego, si $\mathcal{M}$ es (ε, δ)-DP unbounded con dominio $\mathbb{D}$, entonces $\mathcal{M} \circ \mathcal{S}_{\mathrm{Poiss}}$ con dominio $\mathbb{D}$ es $(\ln(p\mathrm{e}^{\varepsilon} + (1-p)), p\delta)$-DP unbounded.*

Teorema II.4 (Amplificación de privacidad: SWOR [7], [8])**.** *Sean $\mathbb{D}$ una clase de bases de datos de tamaño n y $\mathbb{D}_m = \{D' \in \mathbb{D} \mid D' \subseteq D \in \mathbb{D} \text{ y } |D| = m\}$ para $m < n$, y sea $\mathcal{S}_{\mathrm{SWOR}}$ un muestreo sin reemplazo (SWOR), i.e., $\mathcal{S}(D)$ selecciona uniformemente un subconjunto entre todo subconjunto de D de tamaño m. Luego, si $\mathcal{M}$ es (ε, δ)-DP bounded con dominio $\mathbb{D}_m$, entonces $\mathcal{M} \circ \mathcal{S}_{\mathrm{SWOR}}$ con dominio $\mathbb{D}$ es $(\ln(\frac{m}{n}\mathrm{e}^{\varepsilon} + (1 - \frac{m}{n})), \frac{m}{n}\delta)$-DP bounded.*

Los parámetros de privacidad de $\mathcal{M} \circ \mathcal{S}$ son esencialmente equivalentes en ambos casos: Obsérvese que $\mathrm{P}\{x \in \mathcal{S}_{\mathrm{Poiss}}(D)\} = p$ y $\mathrm{P}\{x \in \mathcal{S}_{\mathrm{SWOR}}(D)\} = \frac{m}{n}$ para todo $D \in \mathbb{D}$ y $x \in D$. Además, estas cotas son estrictas [7], [8].

A partir de estos teoremas, podemos observar el llamado principio de «amplificación de privacidad mediante muestreo» [6], [7], que establece que las garantías de privacidad de un mecanismo DP pueden mejorarse cuando se aplica a

un subconjunto pequeño de registros con respecto a la base de datos original. En este caso, los parámetros de privacidad de $\mathcal{M} \circ \mathcal{S}$ son menores que los correspondientes de $\mathcal{M}$. Esta mejora en privacidad también se explica informalmente por el aumento de la incertidumbre, ya que se desconoce qué registros se han muestreado o eliminado [9].

Nuestros teoremas de composición paralela probabilística que presentamos a continuación son también generalizaciones de estos resultados de muestreo y disfrutan de una amplificación de privacidad análoga.

III. Teoremas de composición paralela probabilística

En esta sección, presentamos nuestros principales resultados: dos teoremas de composición paralela probabilística. Al igual que el teorema de composición paralela original, nuestros teoremas acotan los parámetros de privacidad de la composición de k mecanismos DP que toman como entrada subconjuntos disjuntos de la base de datos D. Sin embargo, en lugar de elegir de forma determinista qué mecanismo toma qué registro de D, esta elección es probabilística. En particular, la elección aleatoria de particiones nos permite obtener cotas inferiores (o iguales) de los parámetros del mecanismo compuesto que las obtenidas por la composición paralela original. En particular, la composición de mecanismos DP puros sigue dando un mecanismo DP puro (es decir, sin aumentar δ).

Cada uno de nuestros teoremas presenta diferentes maneras de seleccionar particiones. Una diferencia fundamental es que uno está definido para DP *unbounded* y el otro para DP *bounded*, y cada uno generaliza uno de los resultados de muestreo que hemos introducido. Las pruebas de nuestros resultados se incluyen en el Apéndice.

III-A. Composición paralela probabilística (DP unbounded)

En el teorema para DP *unbounded*, cada elemento x de D se incluye en la entrada de un único mecanismo $\mathcal{M}_i$ con probabilidad fija p_i (con $p_1 + \cdots + p_k = 1$). Nótese que todos los x se incluyen siguiendo una misma distribución.

Teorema III.1 (Composición paralela probabilística (DP *unbounded*))**.** *Sea $\mathbb{D}$ una clase de bases de datos cerrada por subconjuntos (es decir, $D' \in \mathbb{D}$ para todo $D' \subseteq D \in \mathbb{D}$). Para todos los $i \in [k]$, sea $\mathcal{M}_i$ un mecanismo $(\varepsilon_i, \delta_i)$-DP unbounded con dominio $\mathbb{D}$ y tomemos $\mathcal{M} = \mathcal{M}_1 \times \cdots \times \mathcal{M}_k$.*

Sea π un algoritmo aleatorio con dominio $\mathbb{D}$ tal que, para toda $D \in \mathbb{D}$, $\pi(D) = (\pi_1(D), \ldots, \pi_k(D))$ y cada elemento $x \in D$ sea incluido en un único $\pi_i(D)$ con probabilidad p_i (i.e., $p_1 + \cdots + p_k = 1$).

Entonces, $\mathcal{M} \circ \pi$ con dominio $\mathbb{D}$ es (ε, δ)-DP unbounded con $\varepsilon = \ln(\sum_{i=1}^{k} p_i \mathrm{e}^{\varepsilon_i})$ y

$$\delta = \max\left\{ \sum_{i=1}^{k} p_i \delta_i, \frac{\sum_{i=1}^{k} p_i \mathrm{e}^{-\varepsilon_i} \delta_i}{\sum_{i=1}^{k} p_i \mathrm{e}^{-\varepsilon_i}} \right\}. \tag{III.1}$$

En primer lugar, observamos que los parámetros de privacidad resultantes, ε y δ, verifican que $\varepsilon \leq \max_{i \in [k]} \varepsilon_i$ y $\delta \leq \max_{i \in [k]} \delta_i$, con igualdad solo cuando $\varepsilon_1 = \cdots = \varepsilon_k$ y $\delta_1 = \cdots = \delta_k$. Por lo tanto, realizar una composición paralela sobre entradas disjuntas seleccionadas aleatoriamente proporciona una mejor cota de privacidad que el caso determinista clásico. Cabe señalar también que la cota resultante es bastante intuitiva: obtenemos valores más pequeños seleccionando con mayor probabilidad los mecanismos con menor ε_i, asignados de forma similar a una media ponderada (véase Apartado IV). En este caso, el mínimo ε posible es $\min_{i \in [k]} \varepsilon_i$.

Los parámetros de privacidad son estrictos cuando el término $\sum_{i=1}^{k} p_i \delta_i$ de la Ecuación (III.1) alcanza el máximo de δ, que se consigue si y solo si $0 \leq \sum_{i,j=1}^{k} p_i p_j (\mathrm{e}^{-\varepsilon_j} - \mathrm{e}^{-\varepsilon_i})(\delta_i - \delta_j)$ (véase Proposición V.1). Esta desigualdad se cumple en algunos casos notables, por ejemplo, cuando podemos reordenar los subíndices tales que $\varepsilon_1 \leq \varepsilon_2 \leq \cdots \leq \varepsilon_k$ y $\delta_1 \leq \delta_2 \leq \cdots \leq \delta_k$ (o, alternativamente, con $\geq$), incluyendo los casos en los que $\varepsilon_1 = \cdots = \varepsilon_k$ o $\delta_1 = \cdots = \delta_k$. El otro término del máximo de la Ecuación (III.1) es necesario para nuestra prueba, pero creemos que podría ser en realidad superfluo.

III-B. Composición paralela probabilística (DP bounded)

Por otro lado, nuestro teorema para DP *bounded* garantiza que cada mecanismo toma como entrada un número prefijado n_i de registros de la base de datos.

Teorema III.2 (Composición paralela probabilística (DP *bounded*))**.** *Sea $k \in \mathbb{N}$, $k \geq 2$, sean $n_1 + n_2 + \cdots + n_k = n$ números enteros positivos, y sea $p_i = \frac{n_i}{n}$ para todo $i \in [k]$. Sea $\mathbb{D}$ una clase de bases de datos de tamaño n. Para todo $i \in [k]$, sea $\mathbb{D}_{n_i} = \{D^* \mid D^* \subseteq D \in \mathbb{D}, |D^*| = n_i\}$, y sea $\mathcal{M}_i$ un mecanismo $(\varepsilon_i, \delta_i)$-DP bounded con dominio $\mathbb{D}_{n_i}$. Sea $\mathcal{M} = \mathcal{M}_1 \times \cdots \times \mathcal{M}_k$.*

Sea π un algoritmo aleatorio con dominio $\mathbb{D}$ tal que, para cada $D \in \mathbb{D}$, $\pi(D) = (\pi_1(D), \ldots, \pi_k(D))$ es una variable aleatoria que selecciona uniformemente una partición entre todas las particiones $(D_1, \ldots, D_k)$ de D tales que $|D_i| = n_i$ para todo $i \in [k]$.

Entonces $\mathcal{M} \circ \pi$ con dominio $\mathbb{D}$ es $(\varepsilon_\lambda, \delta_\lambda)$-DP bounded para cualquier $\lambda = (\lambda_1, \ldots, \lambda_k) \in [0,1]^k$, donde

$$\mathrm{e}^{\varepsilon_\lambda} = \max_{i \in [k]} \left\{ \left((1 - \lambda_i) + \sum_{\substack{j=1 \\ j \neq i}}^{k} \lambda_j \frac{p_j}{1 - p_j} \mathrm{e}^{\varepsilon_j} \right) \mathrm{e}^{\varepsilon_i} \right\}$$

y

$$\delta_\lambda = \sum_{i=1}^{k} \left(1 + \sum_{\substack{j=1 \\ j \neq i}}^{k} \lambda_j \frac{p_j}{1 - p_j} \right) p_i \delta_i.$$

Cabe señalar que obtenemos que $(\varepsilon_\lambda, \delta_\lambda)$-DP *bounded* para cada $\lambda \in [0,1]^k$, donde λ no es más una variable que describe el dominio de parámetros de privacidad posibles. Por lo tanto, conviene elegir los valores de λ que minimicen las expresiones de ε_λ y δ_λ para obtener los parámetros de privacidad más estrictos posibles. Se puede calcular directamente el mínimo para parámetros de privacidad y probabilidades dadas, pero dejamos la búsqueda del mínimo para variables generales como tarea futura debido a su complejidad. No obstante, para los lectores interesados, proporcionamos una cota simbólica aceptable (aunque generalmente no es la mínima): Al elegir

$$\lambda_j = \frac{1 - p_j}{p_j \mathrm{e}^{\max_{i \neq j} \varepsilon_i} + (1 - p_j)} \in [0,1]$$

para todos los $j \in [k]$, obtenemos

$$\mathrm{e}^{\varepsilon_\lambda} \leq \sum_{j=1}^{k} \frac{p_j \mathrm{e}^{\max_{i \neq j} \varepsilon_i}}{p_j \mathrm{e}^{\max_{i \neq j} \varepsilon_i} + (1 - p_j)} \mathrm{e}^{\varepsilon_j}.$$

Aunque no podamos calcular la cota estricta exacta, aún podemos extraer conclusiones prácticas. Una vez más, el teorema proporciona cotas inferiores o iguales para los parámetros de privacidad de la composición paralela para DP *bounded* que derivamos en Apartado II-B: $\varepsilon = \max_{i,j\in[k],\, i\neq j}(\varepsilon_i + \varepsilon_j)$ y $\delta = \max_{i,j\in[k],\, i\neq j}(\delta_i+\delta_j)$. También proporciona cotas iguales o menores que $\max_{i\in[k]} \varepsilon_i$ y $\max_{i\in[k]} \delta_i$ de la composición paralela para *unbounded DP* (e.g., podemos verlo eligiendo la cota simbólica anterior o $\lambda_j = 0$ para todos los $j \in [k]$). Además, como se menciona en Apartado II-B, la composición paralela de mecanismos DP *bounded* podría no ser DP *bounded* (fuera del contexto de «dominio común» de [5]). Sin embargo, nuestro teorema de composición también funciona incluso fuera del contexto de «dominio común» y nos permite componer mecanismos DP *bounded* para obtener siempre un mecanismo DP *bounded*.

III-C. Un comentario sobre los parámetros de privacidad dados por la amplificación de privacidad mediante muestreo

Nuestros teoremas para DP *unbounded* y *bounded* generalizan el muestreo de Poisson y SWOR, respectivamente. En este caso, obtenemos los teoremas de muestreo originales eligiendo $k = 2$ y $\mathcal{M}_2$ como el mecanismo vacío $\mathcal{M}_\varnothing$, es decir, el mecanismo que siempre devuelve $\varnothing$, que es trivialmente 0-DP.

Este hecho nos permite hacer una observación interesante sobre los parámetros de privacidad obtenidos mediante el muestreo, $\varepsilon' = p\mathrm{e}^{\varepsilon} + (1-p)$ y $\delta' = p\delta$, que no hemos visto mencionado en ningún otro lugar y que pueden ayudar a comprender mejor la expresión de estos parámetros. En particular, los parámetros estrictos obtenidos mediante el muestreo son

$$\mathrm{e}^{\varepsilon'} = p\mathrm{e}^{\varepsilon} + (1-p)\mathrm{e}^{0} \quad \text{y} \quad \delta' = p\delta + (1-p)0.$$

Esto muestra que los parámetros de privacidad de $\mathcal{M}\circ\mathcal{S}$ son una media ponderada de los parámetros de $\mathcal{M}$ y $\mathcal{M}_\varnothing$, es decir, p es la probabilidad de que un registro sea muestreado (es decir, pase a formar parte de la entrada de $\mathcal{M}$, un mecanismo (ε, δ)-DP), y $1-p$ es la probabilidad de que un registro sea eliminado (es decir, que pase a formar parte de la entrada de $\mathcal{M}_\varnothing$, un mecanismo $(0, 0)$-DP).

IV. Discusión

Nuestros dos teoremas de composición paralela probabilística proporcionan parámetros de privacidad inferiores (o iguales) a los de la composición paralela determinista, lo que nos permite mejorar el nivel de privacidad con un efecto similar al del principio de «amplificación de privacidad mediante muestreo». Además, obtenemos informalmente cierta incertidumbre sobre qué mecanismos toman cada registro de la base de datos como entrada en el caso de que el nivel de perturbación de los mecanismos $\mathcal{M}_i$ varíe.

En el caso de la DP *unbounded*, obtenemos una interpretación interesante del parámetro de privacidad resultante ε (así como δ cuando el término izquierdo alcanza el máximo). El e^{ε} resultante corresponde a una media ponderada de $\mathrm{e}^{\varepsilon_i}$ con peso p_i. En particular, ε corresponde a una media ponderada LogSumExp de ε_i. Además, el efecto resultante es mejor que el máximo obtenido por la composición paralela clásica. En particular, nuestra composición puede ser útil, ya que permite compensar los mecanismos en la composición que tengan ε y δ más bajos con aquellos que tengan mayor coste de privacidad.

Sin embargo, observamos que, aunque aumenta el nivel de privacidad al pasar de la composición paralela a la composición paralela probabilística, su aplicabilidad se vuelve más limitada (de forma similar a lo que ocurre al pasar de la composición secuencial a la composición paralela). Las ventajas que se obtienen de la composición paralela probabilística conllevan la pérdida de la capacidad de decidir en qué partición se incluye cada registro, lo cual es un escenario habitual en la composición paralela. Nuestros resultados no permiten elegir en este sentido, ya que la distribución de adónde se incluye cada registro es la misma para todos ellos. No obstante, con nuestros resultados, los analistas pueden seguir especificado la proporción esperada de datos que cada mecanismo toma como entrada e, incluso, pueden seleccionar cantidades fijas utilizando el resultado para DP *bounded* (Teorema III.2).

Aunque la aplicación de la composición paralela probabilística es más limitada que la de la composición paralela clásica, creemos que estos resultados pueden seguir siendo útiles en casos concretos y pueden ser de interés para la comunidad. Por ejemplo, podrían usarse en casos donde un porcentaje de registros deban usarse para otra finalidad, como entrenamiento; para seleccionar *mini-batches* en descenso de gradiente estocástico; o para calcular estadísticos en bases de datos multivariables, donde es común dividir en grupos la base de datos y usar cada grupo para calcular la media de una de las variables [12]. En todos estos casos, es indiferente en qué entrada vaya cada registro, siempre y cuando se respete la proporción de registros en cada entrada. Por lo tanto, son aplicaciones directas que podrían beneficiarse de la composición paralela probabilística. Además, nuestros teoremas se pueden usar para generalizar los muestreos en casos donde se necesiten varias muestras disjuntas.

V. Conclusión y trabajo futuro

En este trabajo, presentamos la composición paralela probabilística, en la que los mecanismos toman como entrada particiones aleatorias de la base de datos. Presentamos dos teoremas (III.1 y III.2), uno para DP *unbounded* y otro para DP *bounded*, que muestran dos técnicas de partición aleatoria potencialmente interesantes. Más importante aún, derivamos cotas sobre la pérdida de privacidad de esta composición y mostramos que son inferiores a las obtenidas por la composición paralela clásica. En particular, la cota dada por el Teorema III.1 corresponde a una media ponderada de los parámetros de privacidad, lo que permite compensar potencialmente la pérdida de privacidad de otros mecanismos. Aunque tengan menos aplicaciones que la composición paralela clásica, creemos que estos teoremas podrían ser herramientas útiles para desarrollar nuevos mecanismos DP y, en general, para mejorar las garantías de privacidad de los usuarios.

Como trabajo futuro, destacamos la evaluación de nuestra composición para ciertas aplicaciones, como las mencionadas anteriormente. De esta manera, podríamos comprender su verdadero potencial y la mejora en privacidad que nuestra composición puede aportar a la población. Además, sería interesante obtener cotas estrictas en el Teorema III.2 o una expresión más simple que nos ayudara a entender mejor el coste en privacidad de esta nueva composición.

APÉNDICE: DEMOSTRACIONES Y OTROS RESULTADOS

Para simplificar la notación, denotamos $P_{\mathcal{M}}(A \mid S) \coloneqq \mathrm{P}\{\mathcal{M}(A) \in S\}$ y $P^{(i)}_{\mathcal{M}}(A \mid S) \coloneqq \mathrm{P}\{\mathcal{M}_i(A_i) \in S_i\}$ en las pruebas. Nótese que $\prod_{i=1}^k P^{(i)}_{\mathcal{M}}(A \mid S) = P_{\mathcal{M}}(A \mid S)$.

Demostración del Teorema III.1: Fijamos un subconjunto medible $S = (S_1, \ldots, S_k) \subseteq \mathrm{Range}(\mathcal{M} \circ \pi)$ y un par de bases de datos *unbounded*-adyacentes $D, D' \in \mathbb{D}$. Suponemos que $D' = D \cup \{y\}$, y por lo tanto tendremos que ver que estas dos ecuaciones se cumplen:

$$P_{\mathcal{M}}(\pi(D) \mid S) \leq \mathrm{e}^{\varepsilon} P_{\mathcal{M}}(\pi(D') \mid S) + \delta,$$
$$P_{\mathcal{M}}(\pi(D') \mid S) \leq \mathrm{e}^{\varepsilon} P_{\mathcal{M}}(\pi(D) \mid S) + \delta.$$

Nótese que $\pi(D)$ es una variable aleatoria discreta con soporte $\Pi_D \coloneqq \{A = (A_i)_{i\in[k]} \mid A_i \cap A_j = \varnothing, i \neq j \text{ y } A_1 \cup \cdots \cup A_k = D\}$ y cuya función de probabilidad es $\mathrm{P}\{\pi(D) = A\} = \prod_{i=1}^k p_i^{|A_i|}$ para todo $A \in \Pi_D$.

Por el teorema de la probabilidad total, tenemos que

$$P_{\mathcal{M}}(\pi(D) \mid S) = \sum_{A \in \Pi_D} P_{\mathcal{M}}(A \mid S)\, \mathrm{P}\{\pi(D) = A\},$$

y, análogamente para $P_{\mathcal{M}}(\pi(D') \mid S)$. Podemos establecer una biyección entre Π_D y $\{A \in \Pi_{D'} \mid y \in A'_j\}$: cada elemento de $A' \in \{A \in \Pi_{D'} \mid y \in A'_j\}$ puede escribirse como $A' = (A_1, \ldots, A_j \cup \{y\}, \ldots, A_k)$ para un único $A = (A_1, \ldots, A_k) \in \Pi_D$. Por lo tanto,

$$P_{\mathcal{M}}(\pi(D') \mid S) = \sum_{j=1}^k \sum_{\substack{A' \in \Pi_{D'} \\ y \in A'_j}} \prod_{i=1}^k P^{(i)}_{\mathcal{M}}(A' \mid S)\, p_i^{|A'_i|}$$

$$= \sum_{j=1}^k \sum_{A \in \Pi_D} \left[P_{\mathcal{M}_j}(A_j \cup \{y\} \mid S_j)\, p_j^{|A_j|+1} \prod_{\substack{i=1 \\ i\neq j}}^k P^{(i)}_{\mathcal{M}}(A \mid S)\, p_i^{|A_i|} \right].$$

Como $\mathcal{M}_j$ son $(\varepsilon_j, \delta_j)$-DP para toda $j \in [n]$, tenemos que

$$\frac{P^{(j)}_{\mathcal{M}}(A \mid S) - \delta_j}{\mathrm{e}^{\varepsilon_j}} \leq P_{\mathcal{M}_j}(A_j \cup \{y\} \mid S_j) \leq \mathrm{e}^{\varepsilon_j} P^{(j)}_{\mathcal{M}}(A \mid S) + \delta_j, \tag{V.1}$$

y sustituyendo la segunda desigualdad (para toda $j \in [n]$), obtenemos que

$$\begin{aligned} P_{\mathcal{M}}(\pi(D') \mid S) &\leq \sum_{j=1}^k \sum_{A \in \Pi_D} \Bigg[(\mathrm{e}^{\varepsilon_j} P^{(j)}_{\mathcal{M}}(A \mid S) + \delta_j)\, p_j^{|A_j|+1} \\ &\qquad \cdot \prod_{\substack{i=1 \\ i\neq j}}^k P^{(i)}_{\mathcal{M}}(A \mid S)\, p_i^{|A_i|} \Bigg] \\ &\leq \sum_{j=1}^k p_j \left(\mathrm{e}^{\varepsilon_j} \sum_{A \in \Pi_D} \left[\prod_{i=1}^k P^{(i)}_{\mathcal{M}}(A \mid S)\, p_i^{|A_i|} \right] + \delta_j \right) \\ &= \sum_{j=1}^k p_j \mathrm{e}^{\varepsilon_j}\, \mathrm{P}\{\mathcal{M}(\pi(D)) \in S\} + \sum_{j=1}^k p_j \delta_j. \end{aligned}$$

Siguiendo pasos análogos, de la primera desigualdad en la Ecuación (V.1) obtenemos la desigualdad restante:

$$P_{\mathcal{M}}(\pi(D') \mid S) \geq \left(\sum_{j=1}^k \frac{p_j}{\mathrm{e}^{\varepsilon_j}} \right) P_{\mathcal{M}}(\pi(D) \mid S) - \sum_{j=1}^k \frac{p_j \delta_j}{\mathrm{e}^{\varepsilon_j}}.$$

Tomando el máximo, obtenemos que $\mathcal{M} \circ \pi$ es (ε, δ)-DP con la δ del enunciado y

$$\varepsilon = \max\left\{ \frac{1}{\sum_{j=1}^k \frac{p_j}{\mathrm{e}^{\varepsilon_j}}}, \sum_{j=1}^k p_j \mathrm{e}^{\varepsilon_j} \right\} = \sum_{j=1}^k p_j \mathrm{e}^{\varepsilon_j}.$$

La última igualdad se puede demostrar fácilmente, por ejemplo, usando la desigualdad de Cauchy–Schwarz. ∎

Proposición V.1 (Condición para cota estricta en el Teorema III.1)**.** *La cota dada en el Teorema III.1 es estricta cuando*

$$\sum_{i=1}^k p_i \delta_i \geq \frac{\sum_{i=1}^k p_i \mathrm{e}^{-\varepsilon_i} \delta_i}{\sum_{i=1}^k p_i \mathrm{e}^{-\varepsilon_i}} \Leftrightarrow 0 \leq \sum_{i,j=1}^k p_i p_j (\mathrm{e}^{-\varepsilon_j} - \mathrm{e}^{-\varepsilon_i})(\delta_i - \delta_j).$$

Demostración: Primero, notamos que la equivalencia siempre es cierta, ya que, para todas las sucesiones de números no negativos $\{a_i\}_{i\in[k]}$, $\{b_i\}_{i\in[k]}$ y $\{c_i\}_{i\in[k]}$ tales que $\sum_{i=1}^k a_i = 1$, se tiene que

$$\left[\sum_{i=1}^k a_i b_i \right] \left[\sum_{i=1}^k a_i c_i \right] - \sum_{i=1}^k a_i b_i c_i = \frac{1}{2} \sum_{i,j=1}^k a_i a_j (b_i - b_j)(c_j - c_i).$$

Esto es un resultado algebraico que se puede comprobar fácilmente. Ahora probamos que la cota es estricta bajo la condición del enunciado.

Conseguimos una cota estricta seleccionando $\mathbb{D}$ como la clase de bases de datos binarias y $\mathcal{M}_i$ como un *leaky randomized response* (o a un *randomized response* si $\delta_i = 0$) que sea estrictamente $(\varepsilon_i, \delta_i)$-DP. En particular, dicho mecanismo $\mathcal{M}_i$ tiene rango en $\{0, 1, a, b\}$ con las siguientes probabilidades

$$\mathrm{P}\{\mathcal{M}_i(D) = 0\} = \begin{cases} \frac{\mathrm{e}^{\varepsilon_i}}{\mathrm{e}^{\varepsilon_i}+1}(1-\delta_i) =: P_0^{(i)} & \text{si } 1 \notin D, \\ \frac{1}{\mathrm{e}^{\varepsilon_i}+1}(1-\delta_i) =: Q_0^{(i)} & \text{si } 1 \in D, \end{cases}$$

$$\mathrm{P}\{\mathcal{M}_i(D) = 1\} = \begin{cases} \frac{1}{\mathrm{e}^{\varepsilon_i}+1}(1-\delta_i) =: P_1^{(i)} & \text{si } 1 \notin D, \\ \frac{\mathrm{e}^{\varepsilon_i}}{\mathrm{e}^{\varepsilon_i}+1}(1-\delta_i) =: Q_1^{(i)} & \text{si } 1 \in D, \end{cases}$$

$$\mathrm{P}\{\mathcal{M}_i(D) = a\} = \begin{cases} \delta_i =: P_a^{(i)} & \text{si } 1 \notin D, \\ 0 =: Q_a^{(i)} & \text{si } 1 \in D, \end{cases}$$

$$\mathrm{P}\{\mathcal{M}_i(D) = b\} = \begin{cases} 0 =: P_b^{(i)} & \text{si } 1 \notin D, \\ \delta_i =: Q_b^{(i)} & \text{si } 1 \in D, \end{cases}$$

Ahora consideramos π como en el Teorema III.1 y $\mathcal{M} = \mathcal{M}_1 \times \cdots \times \mathcal{M}_k$. Fijamos dos bases de datos *unbounded*-adyacentes: $D = \{0, \ldots, 0\}$ y $D' = \{1, 0, \ldots, 0\}$. Haciendo los cálculos, obtenemos que

$$\mathrm{P}\{\mathcal{M}(\pi(D)) = s\} = \prod_{i=1}^k P_{s_i}^{(i)} \qquad \text{y} \tag{V.2}$$

$$\mathrm{P}\{\mathcal{M}(\pi(D')) = s\} = \sum_{l=1}^k p_l Q_{s_l}^{(l)} \prod_{\substack{i=1 \\ i\neq l}}^k P_{s_i}^{(i)}. \tag{V.3}$$

Nótese que, cuando se cumple la desigualdad del enunciado, se obtiene $\delta = \sum_{i=1}^k p_i \delta_i$ en el Teorema III.1. Ahora veremos que las cotas dadas por el teorema son estrictas. Es decir, veremos que existe $S \subseteq \mathrm{Range}(\mathcal{M} \circ \pi)$ tal que

$$\mathrm{P}\{\mathcal{M}(\pi(D')) \in S\} = \mathrm{e}^{\varepsilon}\, \mathrm{P}\{\mathcal{M}(\pi(D)) \in S\} + \delta \tag{V.4}$$

para $\varepsilon = \ln(\sum_{i=1}^k p_i \mathrm{e}^{\varepsilon_i})$ y $\delta = \sum_{i=1}^k p_i \delta_i$.

En este caso, tomamos $S = \{(1, \ldots, 1)\} \cup S_b$ donde S_b es el conjunto de elementos del rango con un solo b. Calculando los valores con las Ecuaciones (V.2) y (V.3), obtenemos que

$$\mathrm{P}\{\mathcal{M}(\pi(D)) \in S\} = \prod_{i=1}^{k} P_1^{(i)} + 0 \qquad \text{y}$$

$$\mathrm{P}\{\mathcal{M}(\pi(D')) \in S\} = \sum_{l=1}^{k} p_l Q_1^{(l)} \prod_{\substack{i=1 \\ i \neq l}}^{k} P_1^{(i)} + \sum_{i=1}^{k} p_i \delta_i.$$

Finalmente, sustituyendo $Q_1^{(l)} = \mathrm{e}^{\varepsilon_l} P_1^{(l)}$, podemos comprobar que se cumple la igualdad de la Ecuación (V.4). ■

Prueba del Teorema III.2: Fijamos un subconjunto medible $S = (S_1, \ldots, S_k) \subseteq \mathrm{Range}(\mathcal{M} \circ \pi)$ y un par de bases de datos *bounded*-adyacentes $D, D' \in \mathbb{D}$. Como $|D| = |D'| = n$, indexamos los elementos de D y D' de manera que l represente el índice de los elementos dispares y los elementos de índice $i \neq l$ sean iguales. Para cada partición $u = (u_1, \ldots, u_k)$ de $[n]$, denotamos $D_u = (D_{u_1}, \ldots, D_{u_k})$ donde D_{u_i} es el subconjunto de elementos con los índices en u_i. Denotamos $\mathcal{U}$ la variable aleatoria que selecciona uniformemente una partición u entre las k-particiones de $[n]$ tales que $|u_i| = n_i$, y denotamos q la probabilidad de cada selección. Con esta notación, tenemos que $\pi(D) = D_{\mathcal{U}}$ y $\pi(D') = D'_{\mathcal{U}}$ (igualdad en distribución).

Denotamos $Q_j \coloneqq \mathrm{P}\{(\mathcal{M}(\pi(D)) \in S) \cap (l \in \pi_j(D))\}$ (análogamente para Q'_j). Tenemos claramente que $P_{\mathcal{M}}(\pi(D) \mid S) = \sum_{j=1}^{k} Q_j$, y, por el teorema de la probabilidad total,

$$Q_j = \sum_{\substack{u \in \mathrm{supp}(\mathcal{U}) \\ l \in u_j}} P_{\mathcal{M}}(D_u \mid S)\, q = \sum_{\substack{u \in \mathrm{supp}(\mathcal{U}) \\ l \in u_j}} q \prod_{i=1}^{k} P_{\mathcal{M}}^{(i)}(D_u \mid S).$$

Si $l \in u_j$, entonces, por construcción, $D_{u_i} = D'_{u_i}$ para todo $i \neq j$, y D_{u_j} y D'_{u_j} son siempre *bounded*-adyacentes. Entonces, $P_{\mathcal{M}}^{(j)}(D_u \mid S) \leq \mathrm{e}^{\varepsilon} P_{\mathcal{M}}^{(j)}(D'_u \mid S) + \delta$, y

$$Q_j \leq \sum_{\substack{u \in \mathrm{supp}(\mathcal{U}) \\ l \in u_j}} q(\mathrm{e}^{\varepsilon_j} P_{\mathcal{M}}^{(j)}(D'_u \mid S) + \delta_j) \prod_{\substack{i=1 \\ i \neq j}}^{k} P_{\mathcal{M}}^{(i)}(D'_u \mid S)$$

$$\leq \mathrm{e}^{\varepsilon_j} \sum_{\substack{u \in \mathrm{supp}(\mathcal{U}) \\ l \in u_j}} P_{\mathcal{M}}(D'_u \mid S) q + p_j \delta_j = \mathrm{e}^{\varepsilon_j} Q'_j + p_j \delta_j =: A.$$

Ahora acotamos Q_j de manera alternativa. Sustituimos $\sum_{i=1:\, i \neq j}^{k} \sum_{v \in \mathrm{supp}(\mathcal{U}):\, l \in v_i} \frac{1}{n - n_j} = 1$ en la expresión de Q_j:

$$Q_j = \sum_{\substack{u \in \mathrm{supp}(\mathcal{U}) \\ l \in u_j}} \left[\sum_{\substack{i=1 \\ i \neq j}}^{k} \sum_{\substack{v \in \mathrm{supp}(\mathcal{U}) \\ l \in v_i}} \frac{1}{n - n_j} \right] \left[\prod_{i=1}^{k} P_{\mathcal{M}}^{(i)}(D_u \mid S) \right] q$$

$$\leq \sum_{\substack{u \in \mathrm{supp}(\mathcal{U}) \\ l \in u_j}} \sum_{\substack{i=1 \\ i \neq j}}^{k} \sum_{\substack{v \in \mathrm{supp}(\mathcal{U}) \\ l \in v_i}} \Bigg[\prod_{\substack{r=1 \\ r \neq j,i}}^{k} P_{\mathcal{M}}^{(r)}(D'_v \mid S) \cdot \prod_{r \in \{j,i\}} \min\{\mathrm{e}^{\varepsilon_r} P_{\mathcal{M}}^{(r)}(D'_v \mid S) + \delta_r, 1\} \Bigg] \frac{1}{n - n_j} q$$

$$\leq \sum_{\substack{i=1 \\ i \neq j}}^{k} \sum_{\substack{v \in \mathrm{supp}(\mathcal{U}) \\ l \in v_i}} \left[\mathrm{e}^{\varepsilon_i + \varepsilon_j} P_{\mathcal{M}}(D'_v \mid S) q + q(\delta_i + \delta_j) \right] \frac{n_j}{n - n_j}$$

$$= \sum_{\substack{i=1 \\ i \neq j}}^{k} \left(\mathrm{e}^{\varepsilon_i + \varepsilon_j} Q'_i + p_i(\delta_i + \delta_j) \right) \frac{p_j}{1 - p_j}$$

$$= \frac{p_j}{1 - p_j} \mathrm{e}^{\varepsilon_j} \sum_{\substack{i=1 \\ i \neq j}}^{k} \mathrm{e}^{\varepsilon_i} Q'_i + p_j \delta_j + \frac{p_j}{1 - p_j} \sum_{\substack{i=1 \\ i \neq j}}^{k} p_i \delta_i =: B.$$

Agrupando ambas desigualdades, obtenemos que $Q_j \leq \min\{A, B\}$ y, usando que $\min\{\alpha, \beta\} \leq (1 - \lambda)\alpha + \lambda\beta$ para toda $\alpha, \beta \geq 0$ y $\lambda \in [0, 1]$, obtenemos que para toda $\lambda_j \in [0, 1]$

$$\sum_{j=1}^{k} Q_j \leq \sum_{i=1}^{k} \left((1 - \lambda_i) + \sum_{\substack{j=1 \\ j \neq i}}^{k} \lambda_j \frac{p_j}{1 - p_j} \mathrm{e}^{\varepsilon_j} \right) \mathrm{e}^{\varepsilon_i} Q'_i + \sum_{i=1}^{k} \left(1 + \sum_{\substack{j=1 \\ j \neq i}}^{k} \lambda_j \frac{p_j}{1 - p_j} \right) p_i \delta_i \leq \mathrm{e}^{\varepsilon_\lambda} \sum_{i=1}^{k} Q'_i + \delta_\lambda.$$

Finalmente, dado que $\sum_{j=1}^{k} Q_j = P_{\mathcal{M}}(\pi(D) \mid S)$, obtenemos que $P_{\mathcal{M}}(\pi(D) \mid S) \leq \mathrm{e}^{\varepsilon_\lambda} P_{\mathcal{M}}(\pi(D') \mid S) + \delta_\lambda$. Por lo tanto, $\mathcal{M}$ es $(\varepsilon_\lambda, \delta_\lambda)$-DP para toda $\lambda \in [0, 1]^k$. ■

AGRADECIMIENTOS

Javier Parra-Arnau es el beneficiario de un contrato posdoctoral Ramón y Cajal (ref. RYC2021-034256-I) financiado por el Ministerio de Ciencia e Innovación de España y la Unión Europea—"NextGenerationEU"/PRTR (Plan de Recuperación, Transformación y Resiliencia). Este trabajo ha sido parcialmente financiado por el Gobierno de España a través del proyecto "DIstributed Smart Communications with Verifiable EneRgy-optimal Yields (DISCOVERY)" PID2023-148716OB-C32; por la Generalitat de Catalunya, mediante la ayuda AGAUR "2021 SGR 01413"; por el Topic Engineering Secure Systems de la Asociación Helmholtz (HGF), apoyado por KASTEL Security Research Labs, Karlsruhe; y por la Exzellenzstrategie alemana (EXC 2050/1–Projekt ID 390696704–Exzellenzcluster CeTI).

REFERENCIAS

[1] C. Dwork, "Differential privacy," in *Proc. Int. Colloq. Automata, Lang., Program. (ICALP)*, 2006.

[2] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, 2014.

[3] F. McSherry, "Privacy integrated queries," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (MOD)*, 2009.

[4] N. Li, M. Lyu, D. Su, and W. Yang, *Differential Privacy: From Theory to Practice*, 2016, vol. 8.

[5] P. Guerra-Balboa, À. Miranda-Pascual, J. Parra-Arnau, and T. Strufe, "Composition in differential privacy for general granularity notions," in *Proc. IEEE Comput. Security Found. Symp. (CSF)*, 2024.

[6] N. Li, W. Qardaji, and D. Su, "On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy," in *Proc. ACM Symp. Inform. Comput. Commun. Secur. (ASIACCS)*, 2012.

[7] B. Balle, G. Barthe, and M. Gaboardi, "Privacy amplification by subsampling: Tight analyses via couplings and divergences," in *Proc. Int. Conf. Neural Inform. Process. Syst. (NIPS)*, vol. 31, 2018.

[8] T. Steinke, "Composition of differential privacy & privacy amplification by subsampling," 2022.

[9] A. Smith, "Differential privacy and the secrecy of the sample," 2009, available at: https://adamdsmith.wordpress.com/2009/09/02/sample-secrecy/.

[10] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Proc. Adv. Cryptology (EUROCRYPT)*, 2006.

[11] D. Kifer and A. Machanavajjhala, "No free lunch in data privacy," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (MOD)*, 2011.

[12] H. H. Arcolezi, J.-F. Couchot, B. Al Bouna, and X. Xiao, "Random sampling plus fake data: Multidimensional frequency estimates with local differential privacy," in *Proc. ACM Int. Conf. Inform., Knowl. Manage. (CIKM)*, 2021.

Códigos Grupo LDOI con Matriz de Adyacencia Binaria

Fabián Molina
Universidad de Oviedo
molinafabian@uniovi.es

Consuelo Martínez
Universidad de Oviedo
cmartinez@uniovi.es

***Resumen*—En este trabajo construimos códigos grupo LDOI (Low-Density Orthogonal Idempotent) dando lugar a familias de códigos que se comportan de forma análoga a los códigos LDPC dentro del marco de los códigos grupo. A partir de las clases 2-ciclotómicas obtenemos idempotentes ortogonales de bajo peso, lo que permite definir códigos v-LDOI cuya matriz de adyacencia es binaria. Bajo ciertas condiciones aritméticas (en particular, la Propiedad (P) para algunos primos) mostramos que las columnas de su matriz de control intersectan a lo sumo en una posición, garantizando así que la matriz de adyacencia sea binaria y que la distancia mínima satisfaga $d = v + 1$. Esta estructura posibilita una descodificación eficiente mediante un algoritmo tipo Bit-Flipping. Además, se presentan construcciones explícitas, ejemplos y limitaciones, así como posibles líneas de trabajo futuro.**

***Index Terms*—códigos grupo, idempotentes ortogonales, clases 2-ciclotómicas, matriz de adyacencia binaria**

I. Introducción

La teoría de códigos desempeña un papel fundamental en el desarrollo de criptosistemas resistentes a la computación cuántica. La primera propuesta de este tipo fue el criptosistema de McEliece, introducido en 1978 ([1]), el cual permanece seguro hasta la fecha. Dicho esquema, basado en códigos de Goppa, presenta una alta eficiencia tanto en los procesos de cifrado como de descifrado; sin embargo, su principal desventaja radica en el gran tamaño de las claves públicas requeridas. A pesar de numerosos intentos por reducir este inconveniente mediante el uso de códigos más estructurados, tales propuestas han resultado vulnerables frente a distintos ataques criptográficos.

En particular, se han propuesto diversas alternativas con el objetivo de mitigar este problema sustituyendo los códigos de Goppa por códigos con mayor estructura algebraica, tales como los códigos Reed–Solomon generalizados o los códigos Reed–Muller binarios (véanse [2], [3], [4], [5], [6]). No obstante, estas propuestas fueron posteriormente demostradas como vulnerables frente a distintos ataques (véanse [7], [8], [9], [10], [11]). Asimismo, se han considerado variantes de los códigos de Goppa (ver [12], [13], [14]), pero muchas de ellas han sido parcial o totalmente vulneradas (véanse [15], [16]).

Los códigos grupo pueden describirse como ideales biláteros de un álgebra de grupo, y su estudio ha demostrado ser de amplio interés tanto desde el punto de vista algebraico como computacional (véanse, por ejemplo, [17], [18], [19]). En este contexto, resulta natural considerar el uso de códigos grupo en el diseño de criptosistemas post-cuánticos. No obstante, un aspecto esencial para su viabilidad práctica es la existencia de algoritmos de descodificación eficientes.

En [20] se propuso un algoritmo de descodificación por permutación parcial para ciertos códigos grupo abelianos en el caso en que el álgebra de grupo es semisimple. Posteriormente, en [21] se introdujo un algoritmo de descodificación basado en la búsqueda de t-subconjuntos del grupo G, diseñado para códigos grupo definidos sobre álgebras de grupo semisimples. En esta misma línea, en [22] se presentaron dos algoritmos de descodificación para códigos grupo: el primero constituye una mejora del algoritmo propuesto en [21] y se basa en el uso de un único $\mathbb{K}G$-síndrome; el segundo generaliza el algoritmo de descodificación de Meggitt, originalmente definido para códigos cíclicos, al contexto de los códigos grupo. Este último emplea una lista estructurada de representantes junto con sus correspondientes $\mathbb{K}G$-síndromes, cuya construcción resulta, sin embargo, computacionalmente costosa para determinados parámetros.

Más adelante, en [23] se estudió nuevamente el algoritmo de descodificación por permutación para códigos grupo asociados a álgebras de grupo semisimples. En dicho trabajo se explota la estructura semisimple para obtener de manera eficiente un conjunto de información y una matriz de control de paridad. Asimismo, se introduce el concepto de GPD-conjunto como una generalización del clásico PD-conjunto, lo que permite aplicar un algoritmo de descodificación por permutación generalizado a aquellos código grupo en los cuales no es posible aplicar descodificación por permutación clásico. Este marco resulta válido tanto para códigos grupo abelianos como no abelianos, ampliando significativamente el alcance de los métodos de descodificación disponibles. No obstante, el problema de determinar PD-conjuntos y GPD-conjuntos de forma eficiente permanece abierto.

Finalmente, en [24] se introdujo la familia de códigos grupo LDOI (*Low-Density Orthogonal Idempotent*), definida como aquella clase de códigos grupo asociados a ideales biláteros de álgebras de grupo semisimples cuyo idempotente ortogonal posee peso de Hamming bajo. Esta característica permite establecer una analogía natural con los códigos LDPC (véanse [25], [26], [27]), puesto que conduce a matrices de control de paridad de baja densidad y, en consecuencia, a algoritmos de descodificación de baja complejidad.

En dicho trabajo se formaliza esta analogía mediante

la introducción de la noción de matriz de adyacencia, herramienta fundamental para relacionar propiedades estructurales del idempotente ortogonal con parámetros del código, como la distancia mínima. Asimismo, se adapta al contexto de los códigos grupo un algoritmo iterativo de descodificación (G-BF) inspirado en el algoritmo de *Bit Flipping* (véanse [28], [29]), el cual opera directamente en el álgebra de grupo y presenta complejidad polinómica.

Uno de los resultados más relevantes de [24] muestra que, para ciertas familias de códigos grupo LDOI —en particular, aquellos cuya matriz de adyacencia es binaria—, una sola iteración del algoritmo es suficiente para corregir todos los errores permitidos por la capacidad correctora del código, alcanzando así un comportamiento óptimo. El objetivo del presente trabajo es establecer condiciones suficientes, de carácter algebraico y combinatorio, que permitan construir códigos grupo LDOI cuya matriz de adyacencia sea binaria. En particular, se proporcionan construcciones explícitas de familias de códigos grupo LDOI, en el caso abeliano, basadas en clases ciclotómicas.

II. Preliminares

De ahora en adelante, vamos a considerar el cuerpo binario $\mathbb{K} = \mathbb{F}_2$, un grupo finito $G = \{g_1 = 1_G, \ldots, g_n\}$ de orden impar y el álgebra de grupo $\mathbb{F}_2G$. También vamos a fijar la base $B = G$ de $\mathbb{F}_2G$, es decir, un orden para los elementos de G. Esto permite identificar elementos de $\mathbb{F}_2G$ con $n-$tuplas de elementos del cuerpo $\mathbb{F}_2$ y viceversa.

De acuerdo con lo anterior, todo código lineal sobre $\mathbb{F}_2$ de longitud n y dimensión k se puede ver como un subespacio vectorial de $\mathbb{F}_2G$. Se dice que $\mathfrak{C}$ es un $G-$código sobre $\mathbb{F}_2$ si puede ser identificado con un ideal bilátero de $\mathbb{F}_2G$. Si existe un grupo (abeliano) G tal que $\mathfrak{C}$ es $G-$código, se dice que $\mathfrak{C}$ es un *código grupo (abeliano)*. De ahora en adelante, asumiremos que $\mathfrak{C} \subseteq \mathbb{F}_2G$ es un código grupo con dimensión k, distancia mínima d y capacidad correctora t.

Como $\text{mcd}(n, 2) = 1$, el álgebra de grupo $\mathbb{F}_2G$ es semi-simple, y por lo tanto, $\mathfrak{C}$ es generado por un idempotente central que denotamos por $e_0 \in \mathbb{F}_2G$. Además, existe un ideal bilátero $\mathfrak{C}^+$, también generado por un idempotente central $e_0^+ \in \mathbb{F}_2G$, tal que $\mathbb{F}_2G = \mathfrak{C} \oplus \mathfrak{C}^+$. En consecuencia, $\mathbf{z} \in \mathfrak{C}$ si y solo si existe $\mathbf{z}' \in \mathbb{K}G$ tal que $\mathbf{z} = \mathbf{z}'e_0$. Lo anterior es equivalente a que $\mathbf{z}e_0^+ = 0$.

Definición II.1 ([22]). *El $\mathbb{F}_2G-$*síndrome *de $\mathbf{z} \in \mathbb{F}_2G$ es el elemento de $\mathbb{F}_2G$ dado por*

$$S^+(\mathbf{r}) := \mathbf{r}e_0^+. \tag{1}$$

Notemos que, si $\mathbf{r} = \mathbf{c} + \mathbf{e}$ es una palabra recibida, donde $\mathbf{c} \in \mathfrak{C}$ y $\mathbf{e} \in \mathbb{F}_2G$, entonces $S^+(\mathbf{r}) = \mathbf{e}e_0^+$.

Teorema II.2 ([22]). *Si $S^+(\mathbf{r})$ es el $\mathbb{F}_2G-$síndrome de una palabra recibida $\mathbf{r} \in \mathbb{F}G$, entonces la ecuación $\mathbf{x}e_0^+ = S^+(\mathbf{r})$ tiene a lo sumo una solución con peso $\leq t$.*

Así, descodificar por $\mathbb{F}_2G-$síndrome consiste en encontrar una solución de la ecuación $\mathbf{x}e_0^+ = S^+(\mathbf{r})$ con peso $\leq t$.

Sea $C_{g_i}^+ \in M_{n\times 1}(\mathbb{F}_2)$ es el vector columna de coeficientes de $g_ie_0^+$, $i \in \{1, \ldots, n\}$, con respecto a la base $B(= G)$. Si $\mathbf{r} \in \mathbb{F}_2G$ es la palabra recibida, denotamos por $V^+ \in M_{n\times 1}(\mathbb{F}_2)$ al vector columna de los coeficientes (con respecto a B) de su $\mathbb{F}_2G-$síndrome $S^+(\mathbf{r})$. Entonces, descodificar por $\mathbb{F}_2G-$síndrome es equivalente a encontrar una solución $X = (X_1, \ldots, X_n)$, con peso $w \leq t$, del sistema

$$X_1C_{g_1}^+ + \cdots + X_nC_{g_n}^+ = V^+. \tag{2}$$

En [24] se abordó este problema introduciendo los códigos grupo LDOI. Informalmente, se dice que un código grupo $\mathfrak{C}$ es *v-LDOI* si $v = \text{wt}(e_0^+) \ll n$ (ver [24]). Si $v = 1$, entonces $e_0^+ = 1$ y $\mathfrak{C} = \{0\}$, así que se impone la condición $v \neq 1$. De acuerdo con la noción anterior, si $\mathfrak{C}$ es un código grupo $v-$LDOI, la matriz

$$\mathcal{C}^+(g_1, \ldots, g_n) := \begin{pmatrix} C_{g_1}^+ & \ldots & C_{g_n}^+ \end{pmatrix}, \tag{3}$$

tiene el mismo número v de entradas no nulas en cada columna (ver [24]). Además como $\mathcal{C}^+(g_1, \ldots, g_n)$ se comporta como una matriz de control de $\mathfrak{C}$ (ver [23]), al suponer que v es mucho más pequeño que n, se tiene que $\mathfrak{C}$ se asemeja a un código LDPC.

Si d es la distancia mínima de $\mathfrak{C}$, entonces

$$d \leq \text{wt}(e_0) \leq \text{wt}\left(e_0^+\right) + \text{wt}\,(1) = v + 1. \tag{4}$$

En [24] se encontraron condiciones en códigos grupo $v-$LDOI que permiten obtener la igualdad $d = v + 1$, así como otras propiedades útiles para la descodificación. Tales propiedades están ligadas a algunas características de su matriz de adyacencia.

Definición II.3 ([24]). *La matriz de adyacencia de un código grupo $\mathfrak{C} \subseteq \mathbb{F}_2G$ es la matriz $\mathcal{A} \in M_{n\times n}(\mathbb{Z})$ cuyas entradas (i, j) vienen dadas por*

$$a_{i,j} := \begin{cases} 0 & \text{si } i = j, \\ \left|\text{Supp}(C_{g_i}^+) \cap \text{Supp}(C_{g_j}^+)\right| & \text{si } i \neq j. \end{cases} \tag{5}$$

Si $a_{i,j} \in \{0, 1\}$ para todo i, j, se dice que $\mathcal{A}$ es binaria.

Nótese que el cálculo de $\mathcal{A}$ tiene un orden de complejidad $\mathcal{O}(vn^2)$. Además la matriz $\mathcal{A}$ es simétrica, y cada fila es una permutación de la primera fila. En efecto, sea $d_{i,j}$ el elemento en la fila i y columna j de la matriz $\mathcal{D}$. Entonces, existen $\mathbf{z} \in \mathbb{F}_2G$ con peso $\text{wt}(\mathbf{z}) = d_{i,j}$ y $\mathbf{z}_i, \mathbf{z}_j \in \mathbb{F}_2G$ con soportes disjuntos tales que

$$g_ie_0^+ = \mathbf{z} + \mathbf{z}_i \text{ y } \text{Supp}(\mathbf{z}) \cap \text{Supp}(\mathbf{z}_i) = \emptyset.$$
$$g_je_0^+ = \mathbf{z} + \mathbf{z}_j \text{ y } \text{Supp}(\mathbf{z}) \cap \text{Supp}(\mathbf{z}_j) = \emptyset.$$

De lo anterior se obtiene que

$$e_0^+ = g_i^{-1}\mathbf{z} + g_i^{-1}\mathbf{z}_i \text{ y } \text{Supp}(g_i^{-1}\mathbf{z}) \cap \text{Supp}(g_i^{-1}\mathbf{z}_i) = \emptyset.$$
$$g_i^{-1}(g_je_0^+) = g_i^{-1}\mathbf{z} + g_i^{-1}\mathbf{z}_j \text{ y } \text{Supp}(g_i^{-1}\mathbf{z}) \cap \text{Supp}(g_i^{-1}\mathbf{z}_j) = \emptyset.$$

Además, $g_i^{-1}\mathbf{z}_i$ y $g_i^{-1}\mathbf{z}_j$ tienen soportes disjuntos. Si $h \in \{1, \ldots, n\}$ es el índice tal que $g_h = g_i^{-1}g_j$, entonces $d_{1,h} = \text{wt}(g_i^{-1}\mathbf{z}) = \text{wt}(\mathbf{z}) = d_{i,j}$. Otras propiedades se muestran a continuación:

Teorema II.4 ([24]). *Sean $\mathfrak{C}$ un código grupo $v-$LDOI y $\mathcal{A}$ su matriz de adyacencia. Supongamos que $\mathbf{z} \in \mathbb{F}_2 G$ tiene peso u. Entonces $\mathbf{z} \in \mathfrak{C}$ si y solo si*

$$\sum_{i<j} d_{i,j} \geq \frac{uv}{2}, \tag{6}$$

donde la suma recorre todos los índices i, j tales que $g_i, g_j \in Supp(\mathbf{z})$.

Teorema II.5 ([24]). *Sean $\mathfrak{C}$ un código grupo $v-$LDOI y $\mathcal{A}$ su matriz de adyacencia. Si $\mathcal{A}$ es binaria, entonces*

$$v(v-1) < n. \tag{7}$$

La cota anterior es una condición (necesaria) que debemos tener en cuenta para construir códigos grupo $v-$LDOI con matriz de adyacencia binaria. Más adelante haremos algunas construcciones explícitas de códigos grupo con tal característica.

Teorema II.6 ([24]). *Si $\mathfrak{C}$ es un código grupo v-LDOI con matriz de adyacencia binaria y distancia mínima d, entonces $d = v + 1$.*

El resultado anterior destaca la importancia de los códigos grupo LDOI con matriz de adyacencia binaria. Otra noción importante para la descodificación es la siguiente.

Definición II.7 ([24]). *El contador de la posición i de $\mathbf{z} \in \mathbb{F}_2 G$, denotado $\gamma_i(\mathbf{z})$, se define por*

$$\gamma_i(\mathbf{z}) := \left|\mathrm{Supp}(S^+(\mathbf{z})) \cap \mathrm{Supp}(g_i e_0^+)\right|. \tag{8}$$

De la definición anterior, se sigue que si $\mathbf{r} = \mathbf{c} + \mathbf{e}$ es una palabra recibida, donde $\mathbf{c} \in \mathfrak{C}$ y $\mathbf{e} \in \mathbb{F}_2 G$, entonces $\gamma_i(\mathbf{r}) = \gamma_i(\mathbf{e})$. Tal cantidad indica el número de veces que en el sistema aparece un "1" en la $i-$ésima posición de aquellas ecuaciones del sistema (2) en las que el término independiente es igual a 1.

Teorema II.8 ([24]). *Con la notación anterior, sean $\mathfrak{C}$ un código grupo v-LDOI con matriz de adyacencia binaria y*

$$b := \begin{cases} v/2 + 1, & \text{si } v \text{ es par}, \\ \lceil v/2 \rceil, & \text{si } v \text{ es impar}. \end{cases} \tag{9}$$

Entonces, $g_i \in \mathrm{Supp}(\mathbf{e})$ si y solo si $\gamma_i(\mathbf{r}) \geq b$.

Los resultados anteriores dan lugar al diseño de un algoritmo de descodificación eficiente para códigos grupo v-LDOI con matriz de adyacencia binaria (ver [24]).

III. Códigos grupo LDOI abelianos

Es conocido ([30]) que si

$$G = \langle h_1 \rangle \times \cdots \times \langle h_m \rangle \cong \mathbb{Z}_{n_1} \oplus \cdots \oplus \mathbb{Z}_{n_m} \tag{10}$$

es un grupo abeliano y $\mathbb{K}$ es un cuerpo finito cuya característica y el orden de G son relativamente primos, existe una correspondencia biyectiva entre los códigos grupo de $\mathbb{K}G$ y las uniones (disjuntas) de clases ciclotómicas en $\mathbb{Z}_{n_1} \times \cdots \times \mathbb{Z}_{n_m}$. En otras palabras, toda unión de clases ciclotómicas determina un código grupo abeliano y viceversa. A partir de esta observación, emplearemos la noción de clases 2-ciclotómicas para definir códigos grupo abelianos cuyo idempotente ortogonal presenta un peso pequeño (LDOI).

Definición III.1. *La* clase 2-ciclotómica *de $(z_1, \ldots, z_m) \in \mathbb{Z}_{n_1} \times \cdots \times \mathbb{Z}_{n_m}$, denotada $Q(z_1, \ldots, z_m)$, viene dada por*

$$Q(z_1, \ldots, z_m) := \{(z_1, \ldots, z_m), \ldots, (2^{s-1} z_1, \ldots, 2^{s-1} z_m)\},$$

donde s es el menor entero positivo tal que $2^s z_i \equiv z_i \pmod{n_i}$ para todo $i \in \{1, \ldots, m\}$.

Las clases 2-ciclotómicas determinan una partición de $\mathbb{Z}_{n_1} \times \cdots \times \mathbb{Z}_{n_m}$ ver [30]). Cuando trabajamos con un grupo abeliano G sobre el cuerpo binario $\mathbb{F}_2$, estas clases permiten construir los idempotentes centrales de $\mathbb{F}_2 G$. En efecto, si $f \in \mathbb{F}_2 G$ es un idempotente con soporte U, entonces

$$f = \sum_{g_j \in U} g_j,$$

y por tanto

$$f = f^2 = \Big(\sum_{g_j \in U} g_j \Big)^2 = \sum_{g_j \in U} g_j^2,$$

de donde se deduce que si $g_i \in U$ entonces $g_i^2 \in U$.

Proposición III.2. *Sea*

$$G = \langle h_1 \rangle \times \cdots \times \langle h_m \rangle \cong \mathbb{Z}_{n_1} \oplus \cdots \oplus \mathbb{Z}_{n_m},$$

donde cada n_i es impar. Si $\mathcal{F}$ es una unión disjunta de clases 2-ciclotómicas en $\mathbb{Z}_{n_1} \times \cdots \times \mathbb{Z}_{n_m}$ y se define

$$f = \sum_{(z_1, \ldots, z_m) \in \mathcal{F}} h_1^{z_1} \cdots h_m^{z_m}, \tag{11}$$

entonces f es un idempotente de $\mathbb{F}_2 G$ y la aplicación $\mathcal{F} \mapsto f$ es biyectiva.

Demostración: Notar que

$$f^2 = \Big(\sum_{(z_1, \ldots, z_m) \in \mathcal{F}} h_1^{z_1} \cdots h_m^{z_m} \Big)^2 = \sum_{(z_1, \ldots, z_m) \in \mathcal{F}} h_1^{2z_1} \cdots h_m^{2z_m}.$$

Como $(2z_1, \ldots, 2z_m)$ pertenece a la misma clase 2-ciclotómica que $(z_1, \ldots, z_m)$, la suma anterior coincide con f, por lo que $f^2 = f$.

Recíprocamente, si f es un idempotente y

$$\mathcal{F} = \{(z_1, \ldots, z_m) : h_1^{z_1} \cdots h_m^{z_m} \in \mathrm{Supp}(f)\},$$

entonces $(2^r z_1, \ldots, 2^r z_m) \in \mathcal{F}$ para todo $r \geq 0$, por lo que $\mathcal{F}$ es unión de clases 2-ciclotómicas. ∎

Con la notación anterior, toda unión de clases 2-ciclotómicas induce un idempotente de $\mathbb{F}_2 G$ y viceversa. La unión vacía corresponde al idempotente 0, y la clase nula $\{(0, \ldots, 0)\}$ corresponde al idempotente 1.

Dado $v < n$, nos interesa determinar cuándo existe un código grupo v-LDOI sobre $\mathbb{F}_2 G$ (con G abeliano). El idempotente ortogonal $e_0^+ \in \mathbb{F}_2 G$ asociado a dicho código está inducido por una unión no vacía $\mathcal{F}$ de clases 2-ciclotómicas de cardinal $|\mathcal{F}| = v$. Por tanto, la existencia de una clase 2-ciclotómica de tamaño menor o igual que v garantiza la existencia de un código v-LDOI. Si s es el menor entero positivo tal que $2^s z_i \equiv z_i \pmod{n_i}$ para todo i, entonces necesariamente $v \geq s$.

A continuación presentamos construcciones concretas que cumplen la condición anterior. Cabe recordar que el orden de 2 en $\mathbb{Z}_3^*$ es 2, por lo que en $\mathbb{Z}_3$ no existen dos clases 2-ciclotómicas distintas.

Definición III.3. *Sean $p > 3$ primo y s el orden de 2 en $\mathbb{Z}_p^*$. Decimos que p satisface la* Propiedad (P) *si para cualesquiera $x, y, x', y' \in \{0, \ldots, s-1\}$ la congruencia*

$$2^x + 2^y \equiv 2^{x'} + 2^{y'} \pmod{p}$$

implica $\{x, y\} = \{x', y'\}$.

El menor primo que satisface la Propiedad (P) es 7.

Teorema III.4. *Sean $p \geq 7$ primo y*

$$G = \langle h_1 \rangle \times \cdots \times \langle h_m \rangle \cong \mathbb{Z}_p^m.$$

Supongamos que $\mathfrak{C}$ es un código grupo de $\mathbb{F}_2 G$ cuyo idempotente ortogonal e_0^+ está inducido por la clase 2-ciclotómica de $z = (z_1, \ldots, z_m) \in \mathbb{Z}_p^m$. Si $z_l \neq 0$ para todo l y p satisface la Propiedad (P), entonces la matriz de adyacencia de $\mathfrak{C}$ es binaria.

Demostración: Sea s el orden de 2 en $\mathbb{Z}_p^*$. Por hipótesis

$$e_0^+ = \sum_{r=0}^{s-1} h_1^{2^r z_1} \cdots h_m^{2^r z_m}. \tag{12}$$

Si $g = h_1^{i_1} \cdots h_m^{i_m}$ y $g' = h_1^{j_1} \cdots h_m^{j_m}$ son elementos distintos de G, y los soportes de ge_0^+ y $g'e_0^+$ se intersecan en alguna posición, entonces para cada coordenada l existen $x(l), x'(l) \in \{0, \ldots, s-1\}$ tales que

$$2^{x(l)} z_l + i_l \equiv 2^{x'(l)} z_l + j_l \pmod{p}.$$

Si tomamos ℓ con $i_\ell \neq j_\ell$, se obtiene

$$z_\ell (2^{x'} - 2^x) \equiv i_\ell - j_\ell \not\equiv 0 \pmod{p},$$

por lo que $x \neq x'$. Si existiera una segunda posición de intersección existirían pares $(y, y') \neq (x, x')$ tales que

$$2^y - 2^{y'} \equiv 2^{x'} - 2^x \pmod{p}. \tag{13}$$

Esto contradice con la Propiedad (P). Así, la intersección tiene a lo sumo un elemento, de modo que la matriz de adyacencia es binaria. ∎

Ejemplo III.5. *Supongamos que $G = C_{31} = \langle g \rangle$. La clase ciclotómica de $z = 3$ es*

$$Q(3) = \{3, 6, 12, 17, 24\}.$$

En el álgebra de grupo $\mathbb{F}_2 G$, podemos considerar el código grupo $\mathfrak{C}$ cuyo idempotente ortogonal es

$$e_0^+ = g^3 + g^6 + g^{12} + g^{17} + g^{24}.$$

Entonces $\mathfrak{C}$ es 5-LDOI, y su matriz de adyacencia es binaria. Esto implica que $d = 6$ y que $\mathfrak{C}$ descodifica todos los errores de peso ≤ 2 usando los contadores definidos previamente.

A continuación se presentan generalizaciones del resultado anterior.

Teorema III.6. *Sean $p \geq 7$ primo y $G = \mathbb{Z}_p^m$. Suponga que e_0^+ está inducido por la unión de dos clases 2-ciclotómicas distintas con representantes $z = (z_1, \ldots, z_m)$ y $w = (w_1, \ldots, w_m)$. Si todas las componentes de z y w son no nulas, $Q(z_l) = Q(w_l)$ para cada l, y p satisface la Propiedad (P), entonces la matriz de adyacencia de $\mathfrak{C}$ es binaria.*

Demostración: Supongamos que s es el orden de 2 en el grupo multiplicativo $\mathbb{Z}_p^*$. Por hipótesis,

$$e_0^+ = \left(\sum_{r=0}^{s-1} h_1^{2^r z_1} \cdots h_m^{2^r z_m} \right) + \left(\sum_{r=0}^{s-1} h_1^{2^r w_1} \cdots h_m^{2^r w_m} \right).$$

Si $g = h_1^{i_1} \cdots h_m^{i_m}$ y $g' = h_1^{j_1} \cdots h_m^{j_m}$ son dos elementos distintos de G, entonces existe $\ell \in \{1, \ldots, m\}$ tal que $i_\ell \neq j_\ell$. Notemos que

$$ge_0^+ = \left(\sum_{r=0}^{s-1} h_1^{2^r z_1 + i_1} \cdots h_m^{2^r z_m + i_m} \right) + \left(\sum_{r=0}^{s-1} h_1^{2^r w_1 + i_1} \cdots h_m^{2^r w_m + i_m} \right),$$

$$g'e_0^+ = \left(\sum_{r=0}^{s-1} h_1^{2^r z_1 + j_1} \cdots h_m^{2^r z_m + j_m} \right) + \left(\sum_{r=0}^{s-1} h_1^{2^r w_1 + j_1} \cdots h_m^{2^r w_m + j_m} \right).$$

Si los soportes de ge_0^+ y $g'e_0^+$ se intersecan en una posición, entonces para cada $l \in \{1, \ldots, m\}$, existe un par $x, x' \in \{0, 1, \ldots, s-1\}$ tal que ocurre uno de los siguientes casos:

1. $2^x z_l + i_l = 2^{x'} z_l + j_l \mod p$.
2. $2^x w_l + i_l = 2^{x'} w_l + j_l \mod p$.
3. $2^x z_l + i_l = 2^{x'} w_l + j_l \mod p$.
4. $2^x w_l + i_l = 2^{x'} z_l + j_l \mod p$.

Sin pérdida de generalidad, vamos a considerar sólo los casos 1 y 3. Como $Q(z_1, \ldots, z_m) \neq Q(w_1, \ldots, w_m)$ y $Q(z_l) = Q(w_l)$ para todo $l \in \{1, \ldots, m\}$, luego existe $r_l \in \{1, \ldots, s-1\}$ tal que $w_l = 2^{r_l} z_l$. Para $l = \ell$, ocurre uno de los siguientes casos:

i. $z_\ell (2^{x'} - 2^x) = i_\ell - j_\ell \neq 0 \mod p$ y así $x \neq x'$.
ii. $z_\ell (2^{x' + r_\ell} - 2^x) = i_\ell - j_\ell \neq 0 \mod p$, y así $x \neq x' + r_\ell$. Además, existe $x'' \in \{0, 1, \ldots, s-1\}$ tal que $2^{x''} = 2^{x' + r_\ell}$. Esto implica que $z_\ell (2^{x''} - 2^x) = i_\ell - j_\ell \neq 0 \mod p$, y por consiguiente, $x \neq x''$.

Si los soportes de ge_0^+ y $g'e_0^+$ se intersecan en una segunda posición, ocurre uno de los siguientes casos:

- Si ocurre i, existe un par $(y, y') \neq (x, x')$, con $y, y' \in \{0, 1, \ldots, s-1\}$ distintos, tal que $z_\ell (2^{y'} - 2^y) = i_\ell - j_\ell \neq 0 \mod p$, y por tanto,

$$2^{y'} - 2^y = 2^{x'} - 2^x \mod p.$$

- Si ocurre ii, existe un par $(y, y'') \neq (x, x'')$, con $y, y'' \in \{0, 1, \ldots, s-1\}$ distintos, tal que $z_\ell (2^{y''} - 2^y) = i_\ell - j_\ell \neq 0 \mod p$, y por tanto,

$$2^{y''} - 2^y = 2^{x''} - 2^x \mod p.$$

En los dos casos, se contradice la Propiedad (P). En consecuencia, los soportes de ge_0^+ y $g'e_0^+$ se intersecan como máximo en una posición. ∎

Ejemplo III.7. *Supongamos que $G = C_7 \times C_7 = \langle g, h \rangle$. Las clases ciclotómicas de $(5, 2)$ y $(6, 4)$, son*

$$Q(5,2) = \{(5,2),(3,4),(6,1)\}$$

y

$$Q(6,4) = \{(6,4),(5,1),(3,2)\},$$

$$Q(5) = Q(6) = \{3,5,6\} \quad y \quad Q(2) = Q(4) = \{2,4,6\}.$$

Entonces, el código de grupo $\mathfrak{C}$ con idempotente ortogonal

$$e_0^+ = g^5h^2 + g^3h^4 + g^6h + g^6h^4 + g^5h + g^3h^2,$$

es 6-LDOI, y su matriz de adyacencia es binaria.

Corolario III.8. *Sean $p \geq 7$ primo y $G = \mathbb{Z}_p^m$. Supongamos que e_0^+ está inducido por la unión de r clases 2-ciclotómicas distintas con representantes $z^{(1)}, \ldots, z^{(r)} \in \mathbb{Z}_p^m$, cuyas componentes son no nulas y*

$$Q(z_l^{(1)}) = \cdots = Q(z_l^{(r)}) \quad \textit{para todo } l.$$

Si p satisface la Propiedad (P), entonces la matriz de adyacencia de $\mathfrak{C}$ es binaria.

En particular, cuando e_0^+ es inducido por la unión de clases no nulas, se tiene $1 \notin \mathrm{Supp}(e_0^+)$. Este hecho, junto con las hipótesis anteriores, implican que la matriz de adyacencia de $\mathbb{C}$ sea binaria.

Teorema III.9. *Sean $p \geq 7$ primo y*

$$G = \langle h_1 \rangle \times \cdots \times \langle h_m \rangle \cong \mathbb{Z}_{n_1} \oplus \cdots \oplus \mathbb{Z}_{n_m},$$

con $n_1, \ldots, n_m$ impares. Definamos $L = \{l \in \{1, \ldots, m\} : n_l = p\}$ y supongamos $L \neq \emptyset$. Si e_0^+ está inducido por la clase 2-ciclotómica de un elemento no nulo $z = (z_1, \ldots, z_m) \in \prod_i \mathbb{Z}_{n_i}$ con $\mathrm{Supp}(z) \subseteq L$ y p satisface la Propiedad (P), entonces la matriz de adyacencia de $\mathfrak{C}$ es binaria.

Demostración: Supongamos que $\mathrm{Supp}(z) = \{l_1, \ldots, l_\eta\}$ y que s es el orden de 2 en el grupo multiplicativo $\mathbb{Z}_p^*$. Por hipótesis,

$$e_0^+ = \sum_{r=0}^{s-1} h_{l_1}^{2^r z_{l_1}} \cdots h_{l_\eta}^{2^r z_{l_\eta}}.$$

Si

$$g = h_1^{i_1} \cdots h_{l_1}^{i_{l_1}} \cdots h_{l_\eta}^{i_{l_\eta}} \cdots h_m^{i_m}$$

y

$$g' = h_1^{j_1} \cdots h_{l_1}^{j_{l_1}} \cdots h_{l_\eta}^{j_{l_\eta}} \cdots h_m^{j_m}$$

son dos elementos distintos de G, entonces existe $\ell \in \{1, \ldots, m\}$ tal que $i_\ell \neq j_\ell$. Como

$$ge_0^+ = \sum_{r=0}^{s-1} h_1^{i_1} \cdots h_{l_1}^{2^r z_{l_1} + i_{l_1}} \cdots h_{l_\eta}^{2^r z_{l_\eta} + i_{l_\eta}} \cdots h_m^{i_m},$$

y

$$g'e_0^+ = \sum_{r=0}^{s-1} h_1^{j_1} \cdots h_{l_1}^{2^r z_{l_1} + j_{l_1}} \cdots h_{l_\eta}^{2^r z_{l_\eta} + j_{l_\eta}} \cdots h_m^{j_m},$$

hay dos casos a considerar:

1. Si $\ell \in \mathrm{Supp}(z)$, entonces los soportes de ge_0^+ y $g'e_0^+$ se intersecan como máximo en una posición. La prueba de ello es análoga al Teorema III.4.
2. Si $\ell \notin \mathrm{Supp}(z)$, entonces los soportes de ge_0^+ y $g'e_0^+$ no se intersecan en ninguna posición.

Con lo anterior, la matriz de adyacencia de $\mathfrak{C}$ es binaria. ■

Corolario III.10. *Sean $p \geq 7$ un número primo,*

$$G = \langle h_1 \rangle \times \cdots \times \langle h_m \rangle \cong \mathbb{Z}_{n_1} \oplus \cdots \oplus \mathbb{Z}_{n_m},$$

donde $n_1, \ldots, n_m$ son números impares, y $L = \{l \in \{1, \ldots, m\} : n_l = p\}$. Supongamos que $L \neq \emptyset$ y que $\mathfrak{C}$ es un código grupo de $\mathbb{F}_2 G$ cuyo idempotente ortogonal e_0^+ está inducido por la unión de r clases $2-$ciclotómicas distintas, cuyos representantes son

$$z^{(1)} = \left(z_1^{(1)}, \ldots, z_m^{(1)}\right), \ldots, z^{(r)} = \left(z_1^{(r)}, \ldots, z_m^{(r)}\right).$$

Si

$$\mathit{Supp}\left(z^{(1)}\right) = \cdots = \mathit{Supp}\left(z^{(r)}\right) = L' \subseteq L,$$

$$Q\left(z_l^{(1)}\right) = \cdots = Q\left(z_l^{(r)}\right)$$

para cada $l \in L'$, y p satisface la Propiedad (P), entonces la matriz de adyacencia de $\mathfrak{C}$ es binaria.

El siguiente ejemplo muestra que las condiciones de los resultados anteriores son suficientes pero no necesarias. Esto deja abierta la posibilidad de encontrar muchos más códigos grupo LDOI con matriz de adyacencia binaria.

Ejemplo III.11. *Supongamos que $C_7 \times C_7 = \langle x, y \rangle$ y $C_3 = \langle z \rangle$. El grupo C_3 actúa (como grupo de automorfismos) sobre $C_7 \times C_7$ mediante*

$$zxz^{-1} = x^2 \text{ y } zyz^{-1} = y^2.$$

Sean $G = (C_7 \times C_7) \rtimes C_3$ y $\mathfrak{C}$ el código grupo de $\mathbb{F}_2 G$ que es ortogonal a

$$e_0^+ = x^3 + x^6 + x^5 + x^5y + x^3y^2 + x^6y^4.$$

Entonces $\mathfrak{C}$ es $6-$LDOI, y además su matriz de adyacencia de $\mathfrak{C}$ es binaria.

Finalmente, es importante señalar que no es posible generalizar estos resultados al producto cartesiano de grupos cíclicos cuyos órdenes son primos distintos que satisfacen la propiedad (P).

Ejemplo III.12. *Todos los códigos grupo construidos sobre $\mathbb{F}_2(C_7 \times C_{31})$ poseen una matriz de adyacencia no binaria.*

IV. Conclusiones

1. Se han establecido condiciones aritméticas —en particular, la Propiedad (P) para primos $p \geq 7$— que garantizan que los códigos LDOI construidos poseen matriz de adyacencia binaria.
2. Aunque las condiciones encontradas son suficientes pero no necesarias, permiten generar familias amplias de códigos grupo LDOI con estructura favorable para la descodificación.

Agradecimientos

Este trabajo fue financiado por la Unión Europea a través del Plan de Recuperación, Transformación y Resiliencia - NextGeneration EU dentro del marco de la Agencia España Digital 2026 (proyecto MRR-MAETD:24-INCIBE-001).

REFERENCIAS

[1] J. McEliece, "A public-key cryptosystem based on algebraic coding theory," in *DSN Progress Report, Jet Propulsion Laboratory, California Institute of Technology*, pp. 114-116, 1978.

[2] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, and D. Schipani, "Enhanced public key security for the McEliece cryptosystem," *Journal of Cryptology*, vol. 29, pp. 1–27, 2016.

[3] T. Berger and P. Loidreau, "How to mask the structure of codes for a cryptographic use," *Designs, Codes and Cryptography*, vol. 35, pp. 63–79, 2005.

[4] T. Berger, P. Cayrel, P. Gaborit, and A. Otmani, "Reducing key length of the McEliece cryptosystem," in *Lecture Notes in Computer Science*, vol. 5580, pp. 77–97, 2009.

[5] H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," *Problems of Control and Information Theory*, vol. 15, no. 2, pp. 157–166, 1986.

[6] V. Sidelnikov, "A public-key cryptosystem based on binary Reed–Muller codes," *Discrete Mathematics and Applications*, vol. 4, no. 3, 1994.

[7] A. Couvreur, P. Gaborit, V. Gauthier, A. Otmani, and J. Tillich, "Distinguisher-based attacks on public-key cryptosystems using Reed–Solomon codes," *Designs, Codes and Cryptography*, vol. 73, no. 2, pp. 641–666, 2014.

[8] L. Minder and A. Shokrollahi, "Cryptanalysis of the Sidelnikov cryptosystem," in *Lecture Notes in Computer Science*, vol. 4515, pp. 347–360, 2007.

[9] S. Shestakov and V. Sidelnikov, "On insecurity of cryptosystems based on generalized Reed–Solomon codes," *Discrete Mathematics and Applications*, vol. 2, no. 4, pp. 439–444, 1992.

[10] C. Wieschebrink, "Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes," in *Lecture Notes in Computer Science*, vol. 6061, pp. 61–72, 2010.

[11] C. Wieschebrink, "An attack on the modified Niederreiter encryption scheme," in *Lecture Notes in Computer Science*, vol. 3958, pp. 14–26, 2006.

[12] H. Janwa and O. Moreno, "McEliece public cryptosystem using algebraic–geometric codes," *Designs, Codes and Cryptography*, vol. 8, pp. 293–307, 1996.

[13] Z. Li, C. Xing, and S. Ling, "Reducing the key size of McEliece cryptosystem from automorphism-induced Goppa codes via permutations," in *Lecture Notes in Computer Science*, vol. 11443, pp. 599–617, 2019.

[14] E. Persichetti, "Compact McEliece keys based on quasi-dyadic Srivastava codes," *Journal of Mathematical Cryptology*, vol. 6, no. 2, pp. 149–169, 2012.

[15] A. Couvreur, I. Márquez-Corbella, and R. Pellikaan, "A polynomial time attack against algebraic geometry code based public key cryptosystem," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, pp. 1446–1450, 2014.

[16] J.-C. Faugère, L. Perret, and F. de Portzamparc, "Algebraic attack against variants of McEliece with Goppa polynomial of a special form," in *Lecture Notes in Computer Science*, vol. 8873, pp. 21–41, 2014.

[17] C. García, S. González, V. Markov, C. Martínez, and A. Nechaev, "Group codes over non-abelian groups," in *Journal of Algebra and its Applications*, vol. 12, no. 7, 2013.

[18] C. García, S. González, V. Markov, C. Martínez, and A. Nechaev, "New examples of non-abelian group codes," in *Advances in Mathematics of Communications*, vol. 10, no. 1, pp. 1-10, 2016.

[19] C. García, S. González, V. Markov, and C. Martínez, "Non-abelian group codes over an arbitrary finite field," in *Journal of Mathematical Sciences*, vol. 223, no. 5, pp. 504-507, 2017.

[20] J. Bernal, Á. del Río, and J. Simón, "Partial Permutation Decoding for Abelian Codes," in *IEEE Transactions on Information Theory*, pp. 274-278, 2012.

[21] C. Matínez and F. Molina, "The syndromes decoding algorithm in group codes," in *Finite Fields and their Applications*, vol. 89, 2023.

[22] C. Martínez, F. Molina, and A. Piñera-Nicolás, "Decoding algorithms in group codes" in *Finite Fields and their Applications*, vol. 108, 2025.

[23] C. Martínez and F. Molina, "Descodificación por permutación en códigos grupo," in *Libro de Actas XVIII Reunión Española sobre Criptología y Seguridad de la Información*, to appear.

[24] F. Molina, P. Santini, and M. Baldi, "Group Codes with Low-Density Orthogonal Idempotent," in *2024 IEEE International Symposium on Information Theory (ISIT)*, pp. 446-451, 2024.

[25] R. Gallager, *Low-density parity-check codes*, in *IRE Transactions on Information Theory*, vol. 8, no. 1, 1962, pp. 21-28.

[26] D. MacKay, *Good error-correcting codes based on very sparse matrices*, in *IEEE Transactions on Information Theory*, vol. 45, no. 2, 1999, pp. 399-431.

[27] R. Tanner, *A recursive approach to low complexity codes*, in *IEEE Transactions on Information Theory*, vol. 27, no. 5, 1981, pp. 533-547.

[28] P. Santini, M. Battaglioni, M. Baldi, and F. Chiaraluce, *Hard-decision iterative decoding of LDPC codes with bounded error rate*, in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-6.

[29] P. Santini, M. Battaglioni, M. Baldi, and F. Chiaraluce, *Analysis of the error correction capability of LDPC and MDPC codes under parallel Bit-Flipping decoding and application to cryptography*, in *IEEE Transactions on Communications*, vol. 68, no. 8, 2020, pp. 4648-4660.

[30] J.J. Bernal and J.J. Simón, *Information sets from defining sets in abelian codes*, in *IEEE Transactions on Information Theory*, vol. 57, no. 12, 2011, pp. 7990-7999.

Hacia un indicador de privacidad para aplicaciones móviles basado en OWASP MAS

Amador Aparicio, M. Mercedes Martínez-González, Alejandro Pérez-Fuente, Pablo-Abel Criado Lozano
Grupo de Investigación en Ingeniería de la Privacidad
Universidad de Valladolid
gi.ingpriv@uva.es

***Resumen*—La privacidad en las aplicaciones móviles sigue siendo difícil de evaluar, incluso con marcos de seguridad consolidados como OWASP MAS. Este trabajo propone un indicador de privacidad sencillo y reproducible, derivado de las pruebas de privacidad definidas en OWASP MASTG, que cuantifica el grado de cumplimiento de dichas verificaciones y, al basarse en pruebas concretas, permite identificar qué comprobaciones se cumplen y cuáles no. Además, se presenta una metodología general para construir indicadores de privacidad a partir de marcos de pruebas de seguridad. Como prueba de concepto, el indicador se aplica a diez aplicaciones reales y se compara con una referencia externa basada en evidencias declarativas (permisos y política de privacidad), mostrando coincidencias parciales y discrepancias que sugieren señales complementarias. Hasta donde se conoce, no se han propuesto indicadores cuantitativos de privacidad derivados directamente de las pruebas de OWASP MAS/MASTG.**

***Index Terms*—Privacidad, Indicadores de Privacidad, Seguridad, Aplicaciones Móviles, OWASP MAS, MASVS, MASTG, MASWE.**

I. Introducción

En el ecosistema de las aplicaciones móviles, la privacidad del usuario está directamente ligada a la seguridad. Sin medidas como el cifrado, la autenticación o la protección del almacenamiento, los datos personales no pueden mantenerse privados [1]. Por ello, este trabajo considera ambas dimensiones de forma conjunta: la seguridad como el medio que lo hace posible y la privacidad como el resultado que se busca medir.

En este contexto, *OWASP Mobile Application Security (OWASP MAS)* [2] se ha consolidado como uno de los marcos de referencia más utilizados para evaluar la seguridad de las aplicaciones móviles [3]. El marco se estructura en tres componentes complementarios: (i) *MASVS*, que define los requisitos de seguridad [4]; (ii) *MASTG*, que especifica las pruebas técnicas necesarias para verificarlos e incluye tanto pruebas de seguridad como verificaciones específicas de privacidad [5]; y (iii) *MASWE*, incorporado recientemente, que enumera debilidades y vulnerabilidades [6].

A pesar de la amplitud de este marco, su uso en herramientas de análisis no refleja la dimensión de la privacidad. Las principales herramientas empleadas para analizar la seguridad de las aplicaciones móviles [7], [8] suelen ignorar esta dimensión, pese a la relación directa entre ambas: la seguridad actúa como soporte esencial para proteger los datos personales de los usuarios [9]. Estas herramientas proporcionan indicadores que permiten estimar lo segura que es una aplicación, pero no informan sobre hasta qué punto respeta la privacidad de los datos personales de sus usuarios. A partir de esta limitación surge la siguiente pregunta de investigación:

RQ. ***¿Es posible construir un indicador de privacidad, basado en marcos de pruebas de seguridad móviles?***

Para responder a esta pregunta, se propone un indicador de privacidad sencillo, fácilmente interpretable y reproducible, construido a partir de las pruebas de privacidad de *OWASP MAS*, que cuantifica hasta qué punto una aplicación móvil respeta los datos personales de sus usuarios. Al estar basado en pruebas de privacidad concretas, también permite identificar qué aspectos de la aplicación deberían mejorarse para que el tratamiento de los datos personales sea más respetuoso. Por otro lado, se introduce una metodología de investigación que permite construir indicadores de privacidad a partir de un marco de análisis de seguridad para aplicaciones móviles. Las fases de validación e integración se dejan para trabajos posteriores, tal y como se discute en la Sección VI y se retoma en la Sección VII.

El resto del trabajo se organiza del siguiente modo. La Sección II presenta el trabajo relacionado. La Sección III describe la metodología propuesta. En la Sección IV se construye y formaliza el indicador de privacidad. La Sección V ofrece una prueba de concepto que ilustra cómo aplicar dicho indicador. En la Sección VI se analizan las implicaciones, ventajas y limitaciones de la propuesta. Por último, la Sección VII recoge las conclusiones y el trabajo futuro.

II. Trabajo relacionado

Marcos de referencia para pruebas de aplicaciones móviles. Diversos trabajos han aplicado y analizado OWASP MAS como marco técnico de referencia para la evaluación de la seguridad de aplicaciones móviles, especialmente en contextos Android y de banca o pagos digitales [10], [11], [12], [13], [14], [15], [16]. *MASVS/MASTG* proporciona requisitos detallados y listas de verificación para arquitectura, almacenamiento de datos, privacidad, criptografía, comunicación de red, interacción con la plataforma, calidad de código y resiliencia. *MASVS* define niveles de verificación y *MASTG* guía pruebas técnicas específicas, cubriendo tanto Android como iOS.

Otros trabajos analizan marcos alternativos, como NIST 800-163, ISO/IEC 27034-1, las guías de ENISA y las propuestas de *vetting* de la NSA [11], [17], [18], [16]. Estos marcos se centran principalmente en la gestión de riesgos, los requisitos organizativos y los procesos de evaluación. Sin embargo, resultan menos específicos en cuanto a casos de prueba técnicos y escenarios concretos de vulnerabilidades móviles, por lo que a menudo se apoyan en *OWASP MASVS/MASTG* como referencia técnica para la verificación detallada de la seguridad en aplicaciones móviles.

Métricas de privacidad para apps móviles. Trabajos previos han evaluado la privacidad en aplicaciones Android mediante propuestas de métricas y revisiones sistemáticas basadas principalmente en los permisos declarados y en factores contextuales como la categoría, la popularidad o el número de instalaciones [19], [20], [21], [22], [23]. Estos trabajos calculan puntuaciones de privacidad o niveles de riesgo a partir de patrones de permisos por categoría, metadatos de la aplicación y, en algunos casos, modelos de *machine learning* que predicen el nivel de privacidad a partir de dichas variables.

Sin embargo, estos enfoques presentan varias limitaciones. La dependencia de información declarativa (permisos, categoría, descripción, número de instalaciones) no garantiza que el modelo refleje el comportamiento real de la aplicación, ya que los permisos pueden estar sobredeclarados, infrautilizados o utilizarse de forma distinta a lo esperado. Además, no incorporan análisis técnico del código ni del comportamiento en tiempo de ejecución, lo que limita la detección de accesos no declarados a datos sensibles o de abusos de los permisos. Sin embargo, no siempre se integran en herramientas automáticas ni en auditorías, lo que dificulta su uso en la práctica.

Seguridad y privacidad en apps móviles. Trabajos previos han analizado la relación entre seguridad y privacidad en aplicaciones móviles desde diferentes ángulos: incoherencias entre políticas de privacidad y comportamiento real de las apps, evaluación de aplicaciones que tratan datos especialmente sensibles (como salud mental) y estudio del papel de la privacidad en la percepción de seguridad del usuario [1], [24], [25], [26], [27]. En conjunto, estos trabajos ponen de manifiesto que la privacidad depende de la existencia de controles de seguridad adecuados, pero no llegan a formalizar un indicador que conecte los resultados de las pruebas de seguridad con el impacto sobre la privacidad de los datos de los usuarios ni su integración en herramientas de auditoría automatizada.

Herramientas de auditorías de aplicaciones móviles. Trabajos previos han revisado herramientas de análisis estático y dinámico para aplicaciones Android, centrándose en la detección de fugas de información, el uso indebido de permisos, la identificación de comportamientos maliciosos y otras vulnerabilidades técnicas mediante el análisis del código y la monitorización en tiempo de ejecución [7], [8]. En conjunto, estos trabajos ofrecen una visión amplia del ecosistema de *security testing* en Android y muestran que las herramientas actuales se orientan principalmente a la seguridad del código y del comportamiento de las aplicaciones.

Sin embargo, la privacidad no se incorpora como una dimensión de evaluación explícita: no se analiza cómo afectan esas vulnerabilidades a los datos personales del usuario ni se definen indicadores que conecten los resultados de las pruebas de seguridad con el grado de exposición de la información. Sigue faltando, por tanto, una medida cuantitativa de privacidad, integrable en herramientas de auditoría automatizada, que ayude a interpretar de manera sencilla los hallazgos de seguridad desde la perspectiva de la protección de los datos de los usuarios.

III. Metodología

Se propone una metodología de investigación inspirada en [3] que estructura la elaboración, validación e integración del indicador en una herramienta de análisis de aplicaciones móviles:

1. **Selección del marco de referencia.** Se comparan estándares de evaluación de seguridad y privacidad en aplicaciones móviles y se selecciona un marco que incluya controles de privacidad verificables, además de los de seguridad.
2. **Identificación de las pruebas.** A partir del marco seleccionado, se recopila el conjunto de pruebas definido, que servirá como base para la construcción del indicador.
3. **Filtrado de pruebas de privacidad.** Del conjunto inicial, se seleccionan las pruebas explícitamente relacionadas con privacidad; este subconjunto determina el alcance del indicador.
4. **Propuesta del indicador.** Con las pruebas de privacidad filtradas, se define una medida cuantitativa que estime el grado de protección de la información personal ofrecida por la aplicación.
5. **Validación del indicador.** Se aplica el indicador a una muestra de aplicaciones y se contrastan los resultados con otros indicadores o fuentes de referencia para evaluar su validez y detectar posibles mejoras.
6. **Integración del indicador.** Una vez validado, el indicador se integra en una herramienta de análisis de aplicaciones móviles para permitir su cálculo automático y reproducible, manteniendo coherencia con el marco seleccionado.

Alcance del trabajo. Aunque la metodología contempla seis pasos (incluida la validación del indicador y su integración en una herramienta), en este artículo se implementan y describen en detalle los pasos 1–4. Asimismo, se presenta una prueba de concepto como *validación preliminar* (paso 5), aplicada a una muestra reducida y contrastada con una fuente de referencia. La validación completa y la integración automatizada del indicador en una herramienta (paso 6) se abordarán en trabajos futuros.

IV. Definición del indicador de privacidad

A partir de la metodología descrita en la Sección III, se detalla cada uno de los pasos necesarios para obtener la definición del indicador de privacidad.

IV-A. Selección del marco de referencia

Se analizan los principales marcos de referencia utilizados en la auditoría de aplicaciones móviles con el objetivo de identificar cuál de ellos ofrece una base más adecuada para incorporar la dimensión de privacidad al análisis de las aplicaciones móviles.

La Tabla I compara los principales marcos de referencia utilizados en las pruebas de seguridad de las aplicaciones móviles [28], atendiendo a su alcance, nivel técnico y grado de atención a la privacidad. Su objetivo es identificar el más adecuado para incorporar la dimensión de privacidad al análisis de las apps móviles.

De los marcos analizados en la Tabla I, se selecciona *OWASP MAS*[1] porque es el único que, además de abordar la seguridad en aplicaciones móviles, incorpora explícitamente la dimensión de la privacidad. Este marco no solo define qué

[1]OWASP Mobile Application Security: https://mas.owasp.org/

Tabla I: Comparativa de marcos de referencia para la auditoría de aplicaciones móviles.

Marco	Enfoque móvil	Cobertura seg.	Cobertura priv.	Observaciones
NIST SP 800-163 (2015)	Parcial	Alta	Nula	Guía de validación sin controles verificables; no permite identificar riesgos que afecten a los datos personales del usuario.
ENISA Mobile App Security (2020)	Sí	Media	Baja	Buenas prácticas generales; poca capacidad para detectar debilidades reales de privacidad en apps móviles.
ISO/IEC 27034-1	No	Media	Nula	Marco genérico de seguridad del software; no analiza impactos directos sobre la privacidad del usuario.
Mobile App Vetting (NSA, 2019)	Sí	Alta	Nula	Orientado a entornos gubernamentales; acceso limitado y sin un enfoque específico en privacidad.
OWASP MAS (MASVS + MASTG)	**Sí**	**Sí**	**Sí**	**Ecosistema abierto y verificable. MASVS define los requisitos, mientras que MASTG proporciona pruebas tanto de seguridad como de privacidad.**

debe cumplir una aplicación, sino que también ofrece cómo comprobarlo. En concreto, *MASVS*[2] establece los requisitos de seguridad y privacidad, mientras que *MASTG*[3] proporciona las pruebas técnicas necesarias para verificarlos. En conjunto, forman un marco completo, abierto y verificable, que permite evaluar de manera sólida tanto las debilidades de seguridad como las de privacidad.

IV-B. Identificación de las pruebas

Las pruebas del marco de referencia *OWASP MAS* se localizan en el catálogo oficial de *MASTG Tests*[4], que constituye la base para identificar y seleccionar las verificaciones relacionadas con la privacidad. La versión analizada de *OWASP MASTG* incluye 189 pruebas; la Tabla II muestra una selección de este conjunto. Cada prueba incorpora un identificador (ID), un título descriptivo, la plataforma aplicable (en este trabajo, Android) y el nivel de seguridad asociado (L1, L2 o R). Además, *MASTG* marca explícitamente qué pruebas están relacionadas con privacidad (columna P).

IV-C. Filtrado de las pruebas de privacidad.

Del conjunto total de pruebas recopiladas en el paso anterior, se seleccionan aquellas relacionadas con la privacidad (columna P). La Tabla III recoge todas las pruebas de privacidad definidas en *OWASP MASTG*.

Se determina que, de las 189 pruebas de *OWASP MASTG*, 12 están directamente relacionadas con la privacidad, de las cuales 6 son estrictamente de privacidad y 6 combinan aspectos de privacidad y seguridad.

IV-D. Propuesta del indicador de privacidad

A partir del conjunto de pruebas de privacidad identificado en el paso anterior, se construye un indicador sencillo que evalúa hasta qué punto una aplicación móvil respeta los datos personales de sus usuarios. El indicador se basa en las pruebas de privacidad definidas en *OWASP MASTG* y mide qué proporción de estas pruebas cumple realmente la aplicación. El indicador es el siguiente:

$$\mathcal{I}_{\text{priv}} = \frac{|T_{TG}^{\checkmark}|}{|T_{TG}|}$$

donde:

- $\mathcal{I}_{\text{priv}}$ es el indicador global de privacidad de la aplicación, con valores entre 0 y 1.
- $|T_{TG}^{\checkmark}|$ es el número de pruebas de privacidad de MASTG que la aplicación cumple.
- $|T_{TG}|$ es el número total de pruebas de privacidad de MASTG consideradas para la aplicación analizada.

El valor del indicador oscila entre 0 y 1. Los valores próximos a 1 indican que la aplicación supera una proporción alta de las verificaciones de privacidad de ambos marcos, mientras que los valores cercanos a 0 reflejan un mayor riesgo para los datos personales del usuario.

V. Prueba de concepto

Tomamos como referencia el trabajo de Chang et al. [23], que estima una puntuación cuantitativa de riesgo de privacidad por aplicación mediante un enfoque distinto al nuestro (permisos del *AndroidManifest.xml* y análisis del texto de la política de privacidad para inferir el tratamiento de datos personales). Con esta referencia externa, aplicamos I_{priv} a las mismas diez aplicaciones del estudio. La Tabla IV recoge, para cada app, su categoría, rango de descargas en *Google Play* y el *Score* normalizado reportado en [23]. Nótese que ambas métricas se interpretan en sentido inverso: valores más altos del *Score* indican mayor riesgo/impacto potencial (apps potencialmente más intrusivas), mientras que valores más altos de I_{priv} reflejan mayor cumplimiento de pruebas de privacidad de *MASTG* (apps potencialmente más respetuosas).

Para el cálculo, consideramos únicamente las pruebas de privacidad aplicables a Android definidas en *OWASP MASTG*. En particular, empleamos las nueve pruebas de la Tabla V (MASTG-TEST-0231, 0262, 0203, 0216, 0206, 0256, 0255, 0254 y 0257), de modo que el denominador del indicador es

[2]OWASP Mobile Application Security Verification Standard: https://mas.owasp.org/MASVS/

[3]OWASP Mobile Application Security Testing Guide: https://mas.owasp.org/MASTG/

[4]https://mas.owasp.org/MASTG/tests/

Tabla II: Algunas pruebas definidas en OWASP MASTG.

ID	Título	Plataforma	L1	L2	R	P
MASTG-TEST-0231	Referencias a APIs de registro	Android	✓	✓	–	✓
MASTG-TEST-0207	Almacenamiento en tiempo de ejecución de datos sin cifrar en el *sandbox* de la aplicación	Android	✓	✓	–	–
MASTG-TEST-0200	Archivos escritos en el almacenamiento externo	Android	✓	✓	–	–
MASTG-TEST-0287	Datos sensibles almacenados sin cifrar mediante la API `SharedPreferences` en el *sandbox* de la aplicación	Android	✓	✓	–	–
MASTG-TEST-0004	Determinar si los datos sensibles se comparten con terceros mediante servicios embebidos	Android	✓	✓	–	–
MASTG-TEST-0262	Referencias a configuraciones de copia de seguridad que no excluyen datos sensibles	Android	✓	✓	–	✓
MASTG-TEST-0201	Uso en tiempo de ejecución de APIs para acceder al almacenamiento externo	Android	✓	✓	–	–
MASTG-TEST-0005	Determinar si los datos sensibles se comparten con terceros mediante notificaciones	Android	✓	✓	–	–
MASTG-TEST-0203	Uso en tiempo de ejecución de APIs de registro	Android	✓	✓	–	✓
...						

Tabla III: Pruebas de privacidad definidas en OWASP MASTG.

ID	Título	Plataforma	L1	L2	R	P
MASTG-TEST-0231	Referencias a las API de registro	Android	✓	✓	–	✓
MASTG-TEST-0262	Referencias a configuraciones de copia de seguridad que no excluyen datos confidenciales	Android	✓	✓	–	✓
MASTG-TEST-0203	Uso en tiempo de ejecución de las API de registro	Android	✓	✓	–	✓
MASTG-TEST-0216	Los datos confidenciales no se excluyen de la copia de seguridad	Android	✓	✓	–	✓
MASTG-TEST-0206	Información personal identificable no declarada en la captura de tráfico de red	Android	–	–	–	✓
MASTG-TEST-0256	Justificación de la falta de permiso	Android	–	–	–	✓
MASTG-TEST-0255	Solicitudes de permiso no minimizadas	Android	–	–	–	✓
MASTG-TEST-0254	Permisos peligrosos de la aplicación	Android	–	–	–	✓
MASTG-TEST-0257	No se restablecen los permisos no utilizados	Android	–	–	–	✓
MASTG-TEST-0215	Datos confidenciales no marcados para exclusión de copia de seguridad	iOS	✓	✓	–	✓
MASTG-TEST-0298	Supervisión en tiempo real de los archivos aptos para la copia de seguridad	iOS	✓	✓	–	✓
MASTG-TEST-0281	Dominios de seguimiento conocidos no declarados	iOS	–	–	–	✓

Tabla IV: Aplicaciones seleccionadas como caso de uso.

Aplicación	Categoría	Descargas	Score
OpenVPN	Comunicación	10M+	0.6892
Firefox Focus	Comunicación	10M+	0.4799
DuckDuckGo	Productividad	50M+	0.6739
Ted	Productividad	5M+	0.7182
Signal	Comunicación	100M+	0.6932
Blockchain.com	Finanzas	10M+	0.7367
Kodi	Reproductores y editores de vídeo	50M+	0.4879
Wikipedia	Libros y referencia	50M+	0.4363
OsmAnd	Viajes	10M+	0.5451
Telegram	Comunicación	1B+	0.7399

$|T_{TG}| = 9$. Las pruebas específicas de iOS se excluyen del análisis.

Tras aplicar el indicador I_{priv} al conjunto de 10 aplicaciones de la Tabla IV, observamos en la Tabla V que los valores de I_{priv} oscilan entre 0,22 y 0,56, lo que sugiere un cumplimiento moderado-bajo incluso en aplicaciones populares y de código abierto.

La Figura 1 compara I_{priv} y el *Score* de [23] en la escala $[0, 1]$. La relación no es uniforme: en algunas apps los valores quedan relativamente próximos (p. ej., *Firefox Focus*), mientras que en otras la discrepancia es clara (p. ej., *Telegram* y *OsmAnd* con I_{priv} bajo y *Score* alto). Esto sugiere que ambos enfoques capturan señales complementarias: I_{priv} resume verificaciones técnicas de privacidad, mientras que el *Score* se apoya en permisos declarados y en el contenido de la política de privacidad.

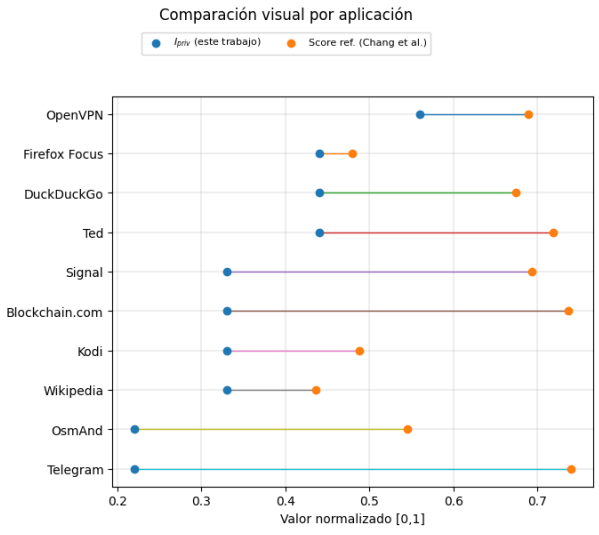


Figura 1: Comparación por aplicación entre I_{priv} (valores más altos indican mayor respeto por la privacidad) y el Score de referencia normalizado de [23] (valores más altos indican mayor riesgo/impacto potencial sobre la privacidad).

VI. Discusión

Aunque el indicador propuesto permite evaluar cuantitativamente la privacidad en aplicaciones móviles, presenta limi-

Tabla V: Tabla de indicadores de privacidad para las diez aplicaciones analizadas.

Aplicación	I_{priv}	Score [23]	Cumple MASTG-TEST	No cumple MASTG-TEST
OpenVPN	**0,56**	0,6892	0206, 0254, 0255, 0256, 0257	0231, 0203, 0262, 0216
Firefox Focus	**0,44**	0,4799	0262, 0216, 0256, 0257	0231, 0203, 0206, 0254, 0255
DuckDuckGo	**0,44**	0,6739	0262, 0216, 0256, 0257	0231, 0203, 0206, 0254, 0255
Ted Editor	**0,44**	0,7182	0206, 0255, 0256, 0257	0231, 0203, 0262, 0216, 0254
Signal	**0,33**	0,6932	0206, 0256, 0257	0231, 0203, 0262, 0216, 0254, 0255
Blockchain	**0,33**	0,7367	0206, 0256, 0257	0231, 0203, 0262, 0216, 0254, 0255
Kodi	**0,33**	0,4879	0206, 0256, 0257	0231, 0203, 0262, 0216, 0254, 0255
Wikipedia	**0,33**	0,4363	0206, 0256, 0257	0231, 0203, 0262, 0216, 0254, 0255
OsmAnd	**0,22**	0,5451	0256, 0257	0231, 0203, 0262, 0216, 0206, 0254, 0255
Telegram	**0,22**	0,7399	0256, 0257	0231, 0203, 0262, 0216, 0206, 0254, 0255

taciones que conviene considerar. En primer lugar, el conjunto de pruebas explícitamente orientadas a privacidad en *OWASP MASTG* es reducido (12 de 189), lo que restringe su cobertura y hace que el resultado dependa de pocas verificaciones. Además, estas pruebas se centran principalmente en aspectos técnicos (p. ej., *logging*, copias de seguridad, permisos o tráfico de red), por lo que el indicador debe interpretarse como una señal parcial del riesgo de privacidad. Por ello, no se espera necesariamente una correlación alta con puntuaciones externas basadas en evidencias distintas (p. ej., permisos o textos declarativos), ya que el indicador refleja únicamente el cumplimiento de verificaciones técnicas y captura una señal complementaria.

En segundo lugar, en esta versión no se ha incorporado *OWASP MASWE*[5], ya que el marco se encuentra en fase beta y su integración podría introducir solapamientos con *MASTG*. Por ejemplo, `MASWE-0001` se relaciona con verificaciones de *logging* ya presentes en *MASTG* (p. ej., `MASTG-TEST-0231` y `MASTG-TEST-0203`). Sin un mapeo y una deduplicación explícitos, combinar ambas fuentes podría distorsionar el valor del indicador.

Por otra parte, aunque la prueba de concepto se basa en una muestra reducida ($n = 10$) y no permite extraer conclusiones generales ni estadísticamente robustas, sugiere que el tipo de aplicación puede influir en qué verificaciones resultan más difíciles de cumplir: en la muestra analizada, las diferencias de I_{priv} se explican principalmente por pruebas de copias de seguridad, permisos y red/PII, relacionadas con cómo está diseñada la app y cómo gestiona los datos. Esto refuerza que las comparaciones entre aplicaciones deben interpretarse junto con el detalle de pruebas incumplidas.

Finalmente, el indicador no distingue dimensiones de privacidad y, en esta versión, asigna el mismo peso a todas las pruebas, ya que *OWASP MASTG* no define niveles de riesgo específicos para privacidad comparables a L1/L2/R. Estas limitaciones justifican una validación posterior a mayor escala con fuentes externas independientes, orientada a evaluar la estabilidad del indicador, su capacidad discriminativa y las discrepancias respecto a otras aproximaciones.

VII. Conclusiones y Trabajo Futuro

Este trabajo muestra que es posible construir un indicador cuantitativo de privacidad a partir de un marco de pruebas de seguridad para aplicaciones móviles, con el fin de estimar, desde un punto de vista técnico, el riesgo asociado al tratamiento de datos personales. El indicador propuesto se calcula a partir del cumplimiento de pruebas orientadas a la privacidad y, al derivarse directamente de *OWASP MASTG*, es trazable e interpretable: permite identificar qué comprobaciones se cumplen y cuáles no. Esto ofrece un apoyo práctico tanto a auditores como a desarrolladores, al facilitar la detección de carencias concretas y orientar acciones de mitigación antes de desplegar la aplicación a usuarios finales.

Tabla VI: Relación de *OWASP MASTG* con las pruebas de privacidad.

Métrica	Valor
Análisis global de MASTG	
Pruebas totales en MASTG	189
Pruebas orientadas a privacidad	12 (6,35 %)
Análisis del subconjunto de pruebas de privacidad	
Pruebas de privacidad exclusivas	6 (50,00 %)
Pruebas de privacidad ligadas a seguridad	6 (50,00 %)
Pruebas de privacidad aplicables a Android	9 (75,00 %)
Pruebas de privacidad aplicables a iOS	3 (25,00 %)

Además, como se aprecia en la Tabla VI, *OWASP MASTG* dedica un espacio reducido a la privacidad: solo 12 de sus 189 pruebas (6,35 %) están explícitamente orientadas a este aspecto. Dentro de este subconjunto, seis pruebas son exclusivas de privacidad (50 %) y seis están ligadas también a controles de seguridad (50 %). Por último, la cobertura no es homogénea entre plataformas: nueve pruebas son aplicables a Android (75 %), mientras que tres corresponden a iOS (25 %).

La prueba de concepto muestra, además, que al comparar I_{priv} con una referencia externa (Tabla V y Figura 1) aparecen coincidencias parciales y discrepancias entre aplicaciones, lo que sugiere que ambos enfoques capturan señales distintas. En conjunto, estos resultados refuerzan que en *OWASP MASTG* la privacidad no se trata de forma aislada, sino estrechamente ligada a la seguridad: junto a pruebas centradas en el uso de datos personales, se incluyen verificaciones que evalúan cómo se protege técnicamente la aplicación. Asimismo, la mayor parte de estas pruebas está disponible en Android, mientras que iOS cuenta con una cobertura sensiblemente menor.

Como línea de trabajo futuro, el indicador podrá ampliarse gracias a su diseño modular, incorporando nuevas pruebas de privacidad si *OWASP MAS* introduce verificaciones adicionales en el futuro y actualizando su definición en función de la

[5]https://mas.owasp.org/MASWE/#

evolución de estos marcos.

Asimismo, se realizará una validación estadística del indicador aplicándolo a una muestra más amplia y diversa de aplicaciones y comparándolo con fuentes externas de referencia para contrastar sus resultados y evaluar su estabilidad (p.,ej., mediante análisis de correlación).

Además, se estudiará una versión más granular, alineada con dimensiones de privacidad (minimización, almacenamiento/copia local, divulgación/rastreo y gestión de permisos), que permita descomponer el indicador en subpuntuaciones interpretables por tipo de verificación.

Finalmente, el indicador se integrará en una herramienta de análisis de aplicaciones móviles, lo que permitirá evaluar su funcionamiento en un entorno real y valorar su utilidad práctica para desarrolladores, auditores y responsables de privacidad.

AGRADECIMIENTOS

Este trabajo se enmarca dentro de las actividades del proyecto estratégico de Ciberseguridad "App-PI (Impacto en la Privacidad de las Aplicaciones): Un ecosistema para la evaluación del impacto de las aplicaciones móviles en la privacidad y seguridad de sus usuarios", que se desarrolla en el marco de un convenio de colaboración entre la Universidad de Valladolid y el Instituto Nacional de Ciberseguridad (INCIBE) para la promoción de proyectos estratégicos de Ciberseguridad en España, dentro del marco de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiado por la Unión Europea (Next Generation).

REFERENCIAS

[1] Sebastian Zimmeck, Ziqi Wang, Lieyong Zou, Roger Iyengar, Bin Liu, Florian Schaub, Shomir Wilson, Norman M. Sadeh, Steven M. Bellovin, and Joel R. Reidenberg. Automated analysis of privacy requirements for mobile apps. In *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017*. The Internet Society, 2017.

[2] Owasp mobile application security (mas). https://mas.owasp.org/, 2025. OWASP Foundation.

[3] Joao Marono, Catarina Silva, João Paulo Barraca, Vitor A. Cunha, and Paulo Salvador. Assessing mobile application privacy: A quantitative framework for privacy measurement. *CoRR*, abs/2311.00066, 2023.

[4] Owasp mobile application security verification standard (masvs). https://mas.owasp.org/MASVS/, 2025. OWASP Mobile Application Security Project.

[5] Owasp mobile application security testing guide (mastg). https://mas.owasp.org/MASTG/, 2025. OWASP Mobile Application Security Project.

[6] Owasp mobile application security weakness enumeration (maswe). https://mas.owasp.org/MASWE/, 2025. OWASP Mobile Application Security Project.

[7] Li Li, Tegawendé F. Bissyandé, Mike Papadakis, Siegfried Rasthofer, Alexandre Bartel, Damien Octeau, Jacques Klein, and Yves Le Traon. Static analysis of android apps: A systematic literature review. *Inf. Softw. Technol.*, 88:67–95, 2017.

[8] Thomas Sutter, Timo Kehrer, Marc Rennhard, Bernhard Tellenbach, and Jacques Klein. Dynamic security analysis on android: A systematic literature review. *IEEE Access*, 12:57261–57287, 2024.

[9] Ali Balapour, Hamid Reza Nikkhah, and Rajiv Sabherwal. Mobile application security: Role of perceived privacy as the predictor of security perceptions. *Int. J. Inf. Manag.*, 52:102063, 2020.

[10] Trevor Henry Chiboora, Lenah Chacha, Theoneste Byagutangaza, and A. Gueye. Evaluating mobile banking application security posture using the owasp's masvs framework. *Proceedings of the 6th ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies*, 2023.

[11] Mykhailo Antonishyn. Mobile applications vulnerabilities testing model. 8:49–57, 2020.

[12] Muhammad Saifulhakim and A. Fajar. Security assessment for digital wallet payment partner applications using the owasp method: A case study in indonesia. *Journal of System and Management Sciences*, 2024.

[13] Grace Friscilla, Margaretha Karo-Karo, Susila Windarta, and Ira Rosianal Hikmah. Evaluating compliance of of the xyz ministry's android messaging applications with owasp masvs: A comprehensive case study. *2024 IEEE 2nd International Conference on Electrical Engineering, Computer and Information Technology (ICEECIT)*, pages 35–40, 2024.

[14] Mykhailo Antonishyn. The usage of dependency graphs to test the security of mobile software applications. *COMPUTER AND INFORMATION SYSTEMS AND TECHNOLOGIES*, 2020.

[15] Aide Alanda, D. Satria, H. Mooduto, and B. Kurniawan. Mobile application security penetration testing based on owasp. *IOP Conference Series: Materials Science and Engineering*, 846, 2020.

[16] Michael A. Ogata, Joshua Franklin, J. Voas, Vincent Sritapan, and S. Quirolgico. Vetting the security of mobile applications. 2015.

[17] Olha Mykhaylova, T. Fedynyshyn, Anastasiia Datsiuk, Bohdan Fihol, and Hennadii Hulak. Mobile application as a critical infrastructure cyberattack surface. pages 29–43, 2023.

[18] Mykhailo Antonishyn and O. Misnik. Analysis of testing approaches to android mobile application vulnerabilities. pages 270–280, 2019.

[19] Erza Gashi and Zhilbert Tafa. Permission-based privacy analysis for android applications. 2017.

[20] Bin Pan, Hongxia Guo, Xing You, and Li Xu. Privacy rating of mobile applications based on crowdsourcing and machine learning. *J. Glob. Inf. Manag.*, 30:1–15, 2022.

[21] Jose M. Del Alamo, Danny S. Guamán, Belen Balmori, and Ana Diez. Privacy assessment in android apps: A systematic mapping study. *Electronics*, 2021.

[22] Gulshan Shrivastava, Prabhat Kumar, Deepak Gupta, and Joel J.P.C. Rodrigues. Privacy issues of android application permissions: A literature review. *Transactions on Emerging Telecommunications Technologies*, 31, 2019.

[23] Kai Chih Chang, Razieh Nokhbeh Zaeem, and K. Suzanne Barber. A framework for estimating privacy risk scores of mobile apps. In Willy Susilo, Robert H. Deng, Fuchun Guo, Yannan Li, and Rolly Intan, editors, *Information Security - 23rd International Conference, ISC 2020, Bali, Indonesia, December 16-18, 2020, Proceedings*, volume 12472 of *Lecture Notes in Computer Science*, pages 217–233. Springer, 2020.

[24] Kristen O'Loughlin, Martha Neary, Elizabeth C Adkins, and Stephen Matthew Schueller. Reviewing the data security and privacy policies of mobile apps for depression. *Internet Interventions*, 15:110 – 115, 2018.

[25] Hamid Reza Nikkhah, Ali Balapour, and Rajiv Sabherwal. Mobile applications security: Role of privacy. In *24th Americas Conference on Information Systems, AMCIS 2018, New Orleans, LA, USA, August 16-18, 2018*. Association for Information Systems, 2018.

[26] Saqib Saeed. Usable privacy and security in mobile applications: Perception of mobile end users in saudi arabia. *Big Data Cogn. Comput.*, 8(11):162, 2024.

[27] Felipe Faustino, Jéssyka Vilela, Carla Silva, and Mariana Maia Peixoto. A systematic mapping study in security software testing for mobile devices. In *International Conference on Internet of Things, Big Data and Security*, 2024.

[28] Mykhailo Antonishyn and Oleksii Misnik. Analysis of testing approaches to android mobile application vulnerabilities. In Aleksandr G. Dodonov, Dmitry V. Lande, Nikolai T. Stoianov, Vitaliy V. Tsyganok, Andrei A. Snarskii, and Oleg Chertov, editors, *Selected Papers of the XIX International Scientific and Practical Conference Ïnformation Technologies and Security"(ITS 2019), Kyiv, Ukraine, November 28, 2019*, volume 2577 of *CEUR Workshop Proceedings*, pages 270–280. CEUR-WS.org, 2019.

Towards an integrated risk analysis security framework according to a systematic analysis of existing proposals

Antonio SANTOS-OLMO
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
Antonio.Santosolmo@uclm.es

Luis Enrique SÁNCHEZ
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
Luise.Sanchez@uclm.es

David G. ROSADO
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
David.GRosado@uclm.es

Carlos BLANCO
Grupo de investigación ISTR
Universidad de Cantabria
Carlos.Blanco@unican.es

Haralambos MOUTADIDIS
Institute for Analytics and Data Science
University of Essex, UK
H.Mouratidis@essex.ac.uk

Eduardo FERNÁNDEZ-MEDINA
Grupo de Seguridad y Auditoría–GSyA
Universidad de Castilla-La Mancha
Eduardo.Fdezmedina@uclm.es

***Resumen*—Este es un resumen extendido del artículo publicado en [1]. La ciberseguridad es esencial para organizaciones que dependen de sistemas de información cada vez más expuestos a amenazas crecientes. Sin embargo, las metodologías de análisis y gestión de riesgos no se han adaptado al ritmo tecnológico, a la colaboración entre empresas ni a la aparición de riesgos dinámicos, asociativos y jerárquicos. Para identificar estas carencias, se realizó una revisión sistemática de la literatura, seleccionando 30 estudios entre 2011 y 2022. El análisis evidenció limitaciones comunes: ausencia de dinamismo, escaso soporte para la reutilización de conocimiento, poca consideración de relaciones jerárquicas, falta de herramientas de apoyo, alta complejidad y baja aplicabilidad en pymes. Como respuesta, se propone MARISMA, un marco integral basado en meta-patrones, procesos dinámicos, una base de conocimiento y una herramienta que automatiza y actualiza el análisis en tiempo real. MARISMA aborda las debilidades detectadas mediante catálogos adaptativos, métricas dinámicas, aprendizaje continuo y soporte colaborativo. En conclusión, la revisión confirma la necesidad de marcos más flexibles y evolutivos, y MARISMA constituye un avance en esta dirección.**

***Index Terms*—information security management, security system, security risk assessment and management**

I. Introducción

La creciente digitalización de empresas y administraciones públicas ha intensificado su dependencia de los sistemas de información, al mismo tiempo que el número, variedad e impacto de las amenazas aumenta de forma constante. En este contexto, la seguridad de la información y, especialmente, los procesos de análisis y gestión de riesgos se han convertido en elementos fundamentales para garantizar la continuidad operativa, la protección de activos críticos y la toma de decisiones estratégicas.

Sin embargo, la evolución tecnológica ha transformado profundamente los escenarios de riesgo. La adopción masiva de paradigmas como Cloud Computing, Internet of Things (IoT), Big Data o los sistemas ciberfísicos propios de la Industria 4.0 ha generado entornos altamente dinámicos, distribuidos y dependientes de múltiples actores. Esto ha introducido riesgos asociativos, jerárquicos y cambiantes que los métodos tradicionales no fueron diseñados para gestionar. Además, las metodologías actuales suelen ser complejas, costosas y difíciles de aplicar, especialmente para pequeñas y medianas empresas que carecen de recursos especializados y herramientas de apoyo.

Ante este panorama, el artículo presenta una revisión sistemática de la literatura orientada a identificar las limitaciones de los enfoques existentes y determinar qué características deberían reunir los nuevos modelos de análisis de riesgos. Como resultado, se plantea MARISMA, un framework integral que combina meta-patrones, procesos dinámicos, mecanismos de aprendizaje y una herramienta software para superar las deficiencias detectadas.

II. Metodología de revisión sistemática

La revisión se realizó siguiendo el protocolo de revisiones sistemáticas propuesto por Kitchenham, adaptado al ámbito de los sistemas de información. La pregunta de investigación que guió el estudio fue:

¿Qué trabajos han desarrollado sistemas de análisis de riesgos considerando riesgos jerárquicos, asociativos, dinámicos y de bajo coste?

II-A. Estrategia de búsqueda y fuentes

Se emplearon repositorios académicos de prestigio (ACM, IEEE, Springer, Elsevier, Wiley, entre otros) y el portal institucional de la Universidad de Castilla-La Mancha. Se utilizaron cadenas de búsqueda basadas en los conceptos de análisis, gestión y evaluación de riesgos, riesgos jerárquicos, riesgos asociativos, dinamismo y reutilización del conocimiento.

La búsqueda se limitó a publicaciones entre 2011 y 2022.

II-B. Criterios de selección

A partir de 6.635 resultados iniciales, se aplicaron criterios de inclusión y exclusión sobre títulos, resúmenes y posteriormente textos completos. Finalmente se seleccionaron 30 estudios pertinentes.

Estos fueron clasificados en cinco categorías según su naturaleza:

- Procesos (4)

- Frameworks (9)
- Modelos (9)
- Metodologías (6)
- Otros (2)

Para cada estudio se registró su alcance, técnicas empleadas, contribuciones y relación con las características consideradas necesarias para una metodología de análisis de riesgos moderna.

III. Resultados del análisis

Los estudios analizados muestran una actividad científica significativa en materia de gestión y análisis de riesgos. Sin embargo, tras el análisis comparativo, se identificaron diez carencias principales presentes de forma transversal:

1. Falta de catálogos adaptativos. Los métodos existentes no permiten actualizar amenazas, vulnerabilidades o controles sin alterar la estructura completa del modelo, dificultando la adaptación a nuevos entornos tecnológicos.
2. Consideración insuficiente de riesgos jerárquicos y asociativos. Aunque varios trabajos reconocen la importancia de interdependencias, ninguno ofrece una solución completa para abordar relaciones entre activos, servicios compartidos o colaboraciones interempresariales.
3. Escasa reutilización del conocimiento y aprendizaje. Pocos enfoques contemplan mecanismos para aprovechar información derivada de análisis previos, lo que limita la madurez y evolución de los sistemas de seguridad.
4. Falta de dinamismo real. La mayoría de las propuestas generan imágenes estáticas del riesgo y no proporcionan mecanismos para recalcular el nivel de riesgo ante incidentes o cambios en el entorno.
5. Ausencia de capacidades colaborativas. Ningún trabajo aborda la gestión de riesgos en ecosistemas compuestos por múltiples organizaciones que comparten activos o procesos.
6. Valoración limitada de elementos. No todas las propuestas incluyen mecanismos completos para valorar activos, impactos o amenazas, lo que dificulta estimaciones de riesgo precisas.
7. Métricas poco dinámicas. Las fórmulas de cálculo suelen ser rígidas y no permiten ajustar los resultados a cambios en el contexto o en los datos disponibles.
8. Alto nivel de subjetividad. La dependencia del juicio experto provoca variabilidad en los resultados y dificulta la comparación entre análisis realizados por distintos consultores.
9. Complejidad y alto coste de aplicación. La mayoría de los métodos son poco accesibles para pymes debido a su complejidad operativa y a la falta de herramientas de apoyo.
10. Ausencia de soporte automatizado y escasez de casos prácticos. Aunque algunos trabajos ofrecen herramientas parciales, pocas propuestas incluyen software completo y validado en entornos reales.

IV. Propuesta: el marco MARISMA

El framework MARISMA (ver Figura 1) surge como respuesta directa a las carencias identificadas en la revisión sistemática, integrando elementos metodológicos, estructurales y tecnológicos con el fin de proporcionar un enfoque completo, adaptable y dinámico para el análisis y la gestión de riesgos en sistemas de información (ver Tabla I).

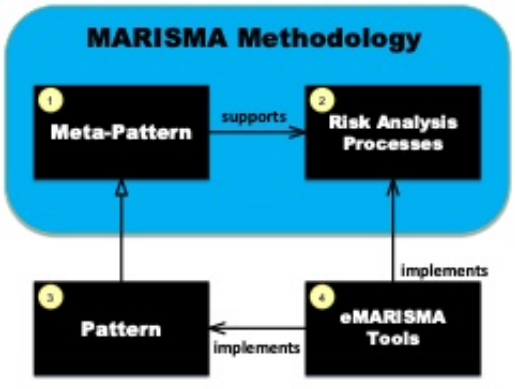


Figura 1. Metodología MARISMA (de [1])

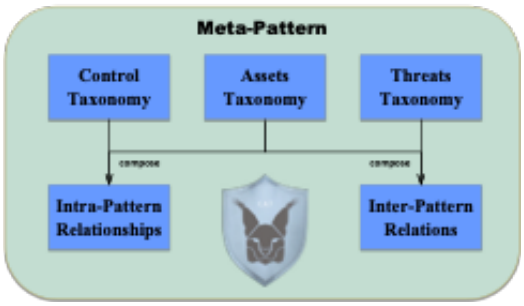


Figura 2. Meta-patrón CAT (de [1])

Su diseño parte de la premisa de que los modelos tradicionales no pueden seguir el ritmo de evolución tecnológica ni las crecientes interdependencias entre organizaciones y activos, por lo que introduce diversos mecanismos destinados a dotar al proceso de flexibilidad, aprendizaje continuo y soporte automatizado.

El núcleo conceptual de MARISMA es el meta-patrón CAT (Control–Activo–Amenaza), una estructura que permite modelar de manera uniforme los elementos esenciales del análisis de riesgos y las relaciones que existen entre ellos (ver Figura 2). A diferencia de los métodos clásicos, donde controles y salvaguardas se incorporan solo en la fase final, el meta-patrón integra estos elementos desde el inicio, permitiendo trazar dependencias directas entre controles implementados, activos protegidos y amenazas relevantes. Esta integración estructural facilita la creación de catálogos adaptativos, capaces de evolucionar sin necesidad de rediseñar la metodología en cada actualización.

Complementando el meta-patrón, MARISMA define tres procesos fundamentales que cubren todas las fases del ciclo de vida del análisis de riesgos (ver Figura 3).

1. RPG (Risk Pattern Generator), se encarga de generar y mantener patrones normativos que actúan como plantillas reutilizables. Estos patrones incorporan conocimiento adquirido a través de múltiples implementaciones y permiten que las organizaciones compartan visiones comunes del riesgo, facilitando un enfoque colaborativo y coherente.
2. RAMG (Risk Analysis and Management Generator),

Tabla I
CARENCIAS IDENTIFICADAS Y SOLUCIONES APORTADAS POR MARISMA

Carencia	Solución de MARISMA
Catálogos adaptativos	Meta-patrón dinámico y patrones actualizables.
Jerarquía y asociatividad	Matrices que relacionan controles, activos y amenazas.
Reutilización de conocimiento	Base de conocimiento y herencia entre patrones.
Dinamismo	Procesos RAMG y DRM con recálculo continuo.
Capacidad colaborativa	Patrones compartidos entre organizaciones.
Valoración precisa	Métricas definidas según necesidades del análisis.
Métricas dinámicas	Fórmulas configurables y ajustables en tiempo real.
Baja subjetividad	Pre-auditorías y sistemas expertos con aprendizaje.
Simplicidad y bajo coste	Automatización mediante la herramienta eMARISMA.
Herramientas y casos reales	Aplicado en múltiples organizaciones de distintos sectores.

realiza la instanciación de dichos patrones adaptándolos al contexto específico de cada organización. Este proceso permite definir métricas dinámicas y fórmulas de cálculo ajustables, resolviendo las limitaciones relacionadas con la valoración de elementos y la rigidez de los modelos tradicionales.

3. DRM (Dynamic Risk Management), introduce mecanismos de mantenimiento continuo del riesgo, basados en eventos, incidentes y sugerencias generadas mediante un sistema experto. Este proceso dota al modelo de un verdadero dinamismo, permitiendo que los riesgos evolucionen de forma automática conforme cambian las condiciones del sistema.

Un aspecto crucial del framework es su base de conocimiento, donde se almacenan patrones, relaciones y aprendizajes procedentes de instancias reales. Esta base facilita la herencia entre patrones, la transferencia de experiencias y la retroalimentación constante entre implementaciones. De este modo, MARISMA no solo gestiona riesgos, sino que aprende y mejora con cada ciclo, proporcionando una aproximación madura y evolutiva.

Finalmente, el framework se materializa en la herramienta eMARISMA, que automatiza procesos, simplifica la aplicación práctica y permite que tanto grandes organizaciones como pymes se beneficien del enfoque. Su arquitectura modular (ver Figura 4), basada en tecnologías abiertas, posibilita la escalabilidad y la integración en distintos entornos, reforzando así su aplicabilidad real. El uso continuado en empresas de diversos sectores y países evidencia su capacidad para adaptarse a contextos heterogéneos, confirmando su eficacia como solución integral para la gestión moderna de riesgos.

V. DISCUSIÓN

El análisis comparativo realizado permite observar que, aunque existe un amplio número de propuestas orientadas a mejorar el análisis y la gestión de riesgos, la mayoría se centran únicamente en aspectos parciales del problema.

Algunos trabajos profundizan en técnicas para reducir la incertidumbre, otros en mecanismos para valorar activos, y otros en estructuras conceptuales destinadas a organizar procesos. Sin embargo, muy pocas propuestas integran de forma simultánea flexibilidad, dinamismo, aprendizaje y aplicabilidad práctica. Otro aspecto relevante es que varios estudios destacan la necesidad de adaptar los modelos de riesgo a contextos emergentes como IoT, Cloud Computing o entornos industriales conectados. No obstante, las soluciones presentadas suelen estar limitadas a ámbitos muy concretos,

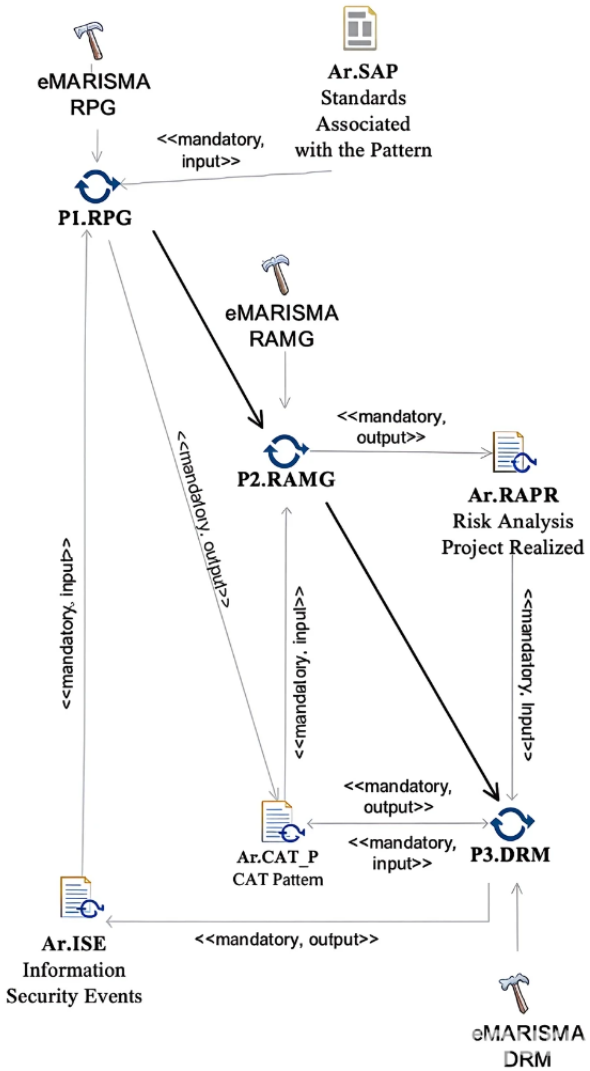


Figura 3. Procesos de MARISMA en SPEM (de [1])

dificultando su aplicación generalizada. A esto se suma la falta de herramientas de soporte que faciliten la implantación, el mantenimiento y la actualización de los análisis de riesgo.

Además, se evidencia un desfase entre los avances propuestos en la literatura y la realidad empresarial, donde predominan los recursos limitados, la carencia de especialistas y la necesidad de herramientas intuitivas. La ausencia de mecanismos para reutilizar conocimiento provoca que cada análisis se aborde como un ejercicio aislado, elevando costes

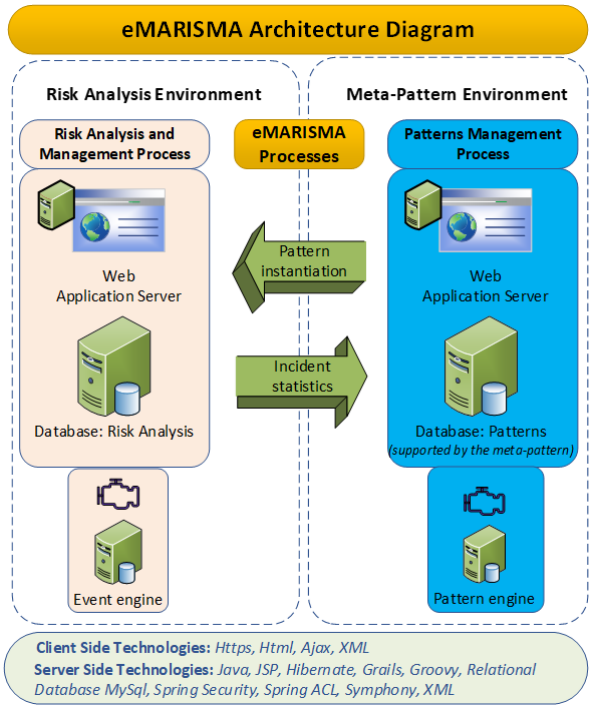


Figura 4. Diagrama de la Arquitectura eMARISMA (de [1])

y reduciendo la eficiencia.

En este contexto, la propuesta MARISMA adquiere especial relevancia, ya que incorpora mecanismos de evolución continua, estructuras reutilizables y una herramienta capaz de dar soporte operativo. Su enfoque integral contrasta con las aproximaciones fragmentadas identificadas en la literatura, posicionándolo como un marco sólido para afrontar los desafíos actuales del análisis de riesgos.

VI. Conclusiones

La revisión sistemática demuestra que, aunque existe una amplia variedad de marcos, modelos y metodologías, ninguno cubre completamente las necesidades actuales de las organizaciones en materia de análisis y gestión de riesgos. Persisten carencias relevantes en dinamismo, reutilización de conocimiento, simplicidad, automatización y tratamiento de riesgos asociados y jerárquicos.

MARISMA surge como una propuesta que integra tanto perspectivas metodológicas como tecnológicas para ofrecer un enfoque adaptable, dinámico y orientado a todo tipo de organizaciones. Su meta-patrón, sus procesos y su herramienta software permiten abordar de forma integral las debilidades detectadas y avanzar hacia un modelo de análisis de riesgos capaz de evolucionar con el entorno organizativo y tecnológico.

Agradecimientos

This work has been developed within the AETHER-UCLM (PID2020-112540RB-C42) funded by MCIN/AEI/10.13039/501100011033, Spain, ALBA-UCLM (TED2021-130355B-C31), ALBA-UC (TED2021-130355A-C33) financed by CIN/AEI/10.13039/501100011033/ Unión Europea NextGenerationEU/PRTR, and supported by the European Union's Horizon 2020 Project "CyberSANE" under Grant Agreement No. 833683.

Referencias

[1] Santos-Olmo, A., Sánchez, L.E., Rosado, D.G., Serrano, M.A., Blanco C., Mouratidis H., Fernández-Medina E. Towards an integrated risk analysis security framework according to a systematic analysis of existing proposals. *Frontiers of Computer Science*. 18, 183808 (2024). https://doi.org/10.1007/s11704-023-1582-6. JCR Impact Factor: 4.6. Q1 - Computer Science-Software Engineering, 17/128.

Security analysis of multilevel secret sharing schemes based on autoparatopisms of Latin squares

Manuel González-Regadera
Universidad de Sevilla
Dpto. Matemática Aplicada I
mgonzalez34@us.es

Raúl M. Falcón
Universidad de Sevilla
Dpto. Matemática Aplicada I
rafalgan@us.es

María Dolores Frau
Universidad de Sevilla
Dpto. Matemática Aplicada I
mdfrau@us.es

***Abstract*—This paper delves into the security analysis of multilevel secret sharing schemes that have a Latin square L with a given autoparatopism Θ as the key, and its entries as pieces of information. This kind of cryptographic protocol does not depend on the existence of absent but indispensable participants. In this paper, we introduce the concept of minimal breaking set (MBS) of Θ-orbits in L as a minimal subset of pieces of information that break the unique Θ-completability of an authorized group. Determining all these minimal breaking sets plays a relevant role in designing the scheme and studying its security level. On the one hand, every minimal authorized set describes a Θ-critical set of L that must intersect every MBS. On the other hand, MBSs enable one to study the resilience, vulnerability and robustness of the scheme. We prove that the set of MBSs only depends on the main class of the Latin square and the conjugacy class of the autoparatopism under consideration. Some illustrative examples are described.**

Keywords: Secret sharing scheme, minimal authorized set, Latin square, autoparatopism, critical set.

I. Introduction

The notion of a *secret sharing scheme* was independently introduced by Blakley [2] and Shamir [20]. This is a cryptographic protocol in which a trusted dealer divides a key into pieces of information that are distributed among a group of m participants, who can collaborate among themselves to recover the secret by sharing their respective pieces of information. The scheme depends on a threshold value $k \leq m$ so that, if there is a subset of participants that share at least k pieces of information, then they form an *authorized set* and recover the secret. Otherwise, they form an *unauthorized set* and the secret still remains unknown. An authorized set is *minimal* if the removal of one participant from the set makes it unauthorized. A multilevel version of this cryptographic protocol arises when the participants are organized into different levels so that each level has a different access threshold.

Recall also here that a *partial Latin square* of order n is an $n \times n$ partial array with non-empty entries in a set of n distinct symbols such that each symbol appears at most once per row and at most once per column. This is a *Latin square* if there are no empty entries in the array. In 1994, Cooper, Donovan and Seberry proposed [5] a secret sharing scheme that has a given Latin square L as the key, and each of its entries as a piece of information. The set of pieces of information of any authorized set of the scheme coincides with the set of non-empty entries of a partial Latin square that is uniquely completable to L. In particular, an authorized set is minimal if and only if it describes a *critical set* of L. That is, a partial Latin square that is uniquely completable to L, in which the removal of any of its entries makes it non-uniquely completable to L. The problem of deciding whether a partial Latin square is uniquely completable is NP-complete [4]. A greedy algorithm to find critical sets in Latin squares was described in [15]. We refer to [3] for a comprehensive survey on critical sets of Latin squares.

From now on, we respectively denote by $\mathcal{PL}(n)$ and $\mathcal{L}(n)$ the set of partial Latin squares of order n with non-empty entries in the set $[n] := \{1, \dots, n\}$, and its subset of Latin squares. Every partial Latin square $P := (P[i,j]) \in \mathcal{PL}(n)$ is uniquely identified with its set of non-empty entries

$$\mathrm{Ent}(P) := \{(i, j, P[i,j])\colon\ i, j, P[i,j] \in [n]\}.$$

The cardinality of this set is the *size* of P. It is known that the size of any critical set of $L \in \mathcal{L}(n)$ depends only on the paratopism class of L (see [6]). Recall here that, if $\mathcal{S}_n$ denotes the symmetric group on the set $[n]$, then a *paratopism* is any pair $\Theta := (\pi; f)$ formed by a permutation $\pi \in S_3$ and a triple $f := (f_1, f_2, f_3) \in S_n \times S_n \times S_n$. Then, a partial Latin square $P \in \mathcal{PL}(n)$ is *paratopic* to $P^\Theta \in \mathcal{PL}(n)$ if

$$\mathrm{Ent}(P^\Theta) := \{\Theta(e)\colon\ e \in \mathrm{Ent}(P)\},$$

where

$$\Theta(e) := (f_{\pi(1)}(e_{\pi(1)}),\ f_{\pi(2)}(e_{\pi(2)}),\ f_{\pi(3)}(e_{\pi(3)}))$$

for all $e := (e_1, e_2, e_3) \in \mathrm{Ent}(P)$. If π is the trivial permutation in S_3, then Θ is called an *isotopism*. If $P^\Theta = P$, then Θ is an *autoparatopism* of P (an *autotopism* if π is trivial). The set $\mathrm{Apar}(P)$ of autoparatopisms of P is a subgroup of the semidirect product

$$S_3 \wr S_n := S_3 \times (S_n \times S_n \times S_n),$$

which is a group under the composition of paratopisms

$$(\rho; g)(\pi; f) := (\pi\rho; g^\pi f)\,,$$

where

$$g^\pi := \left(g_{\pi^{-1}(1)}, g_{\pi^{-1}(2)}, g_{\pi^{-1}(3)}\right).$$

To be paratopic or isotopic are equivalence relations among (partial) Latin squares whose respective equivalence classes are called *main classes* and *isotopic classes*. The distribution of Latin squares into these classes is known for order up to 11 (see [16]–[18]), and that of partial Latin squares is known for order up to six (see [9]–[11], [13]). The distribution into main classes of all critical sets of Latin squares of order up to six is also known [1]. An example of a critical set is known for each main class of Latin squares of order seven [7].

The notion of a critical set of a Latin square has recently been extended [14] to that of a critical set based on a given autoparatopism. (This extension constitutes indeed a natural generalization of the notion of a critical set based on a given autotopism, which was previously introduced in [9].) More specifically, a partial Latin square $P \in \mathcal{PL}(n)$ is a critical set of a Latin square $L \in \mathcal{L}(n)$ with respect to an autoparatopism $\Theta \in \mathrm{Apar}(L)$ (or, simply, a Θ-*critical set* of L) if the following two conditions hold.

1) It is *uniquely* Θ-*completable*. That is, $\mathrm{Ent}(P) \subseteq \mathrm{Ent}(L)$ and there is no other Latin square $L' \in \mathcal{L}(n)$ such that $\mathrm{Ent}(P) \subseteq \mathrm{Ent}(L')$ and $\Theta \in \mathrm{Apar}(L')$.
2) The previous condition does not hold after the removal of any of the entries in P.

If Θ is the trivial paratopism, then this definition coincides with the classical notion of a critical set of a Latin square.

Critical sets based on autoparatopisms can be implemented to design new multilevel secret sharing schemes that do not depend on the existence of absent but indispensable participants. First, the dealer considers a pair $(L, \Theta) \in \mathcal{L}(n) \times \mathrm{Apar}(L)$. The paratopism Θ is made public. Then, similarly to the scheme proposed in [5], the dealer distributes to each participant a distinct entry of P, so that all of them must collaborate to recover the Latin square L, which is the key. If the dealer consider more than one Θ-critical set, some of them even of distinct sizes, then one gets a multilevel scheme. Every minimal authorized group in the access structure of this scheme is formed by participants whose pieces of information describe a Θ-critical set of L. The removal of participants in these groups implies a lack of unique Θ-completion, which in turn means that any attempt to reconstruct the secret would lead to ambiguity. That is, this removal gives rise to unauthorized groups.

For Latin squares of order up to five, a census of critical sets based on autotopisms was described in [12], while a census based on autoparatopisms was recently described in [14]. In order to deal with higher orders, and keeping in mind the mentioned implementation into secret sharing schemes, this paper proposes to deal with the following problems concerning a Latin square $L \in \mathcal{L}(n)$ having a given paratopism $\Theta \in S_3 \wr S_n$ in its autoparatopism group.

Problem 1: Study the minimal sets of entries that we have to remove from L so that the resulting partial Latin squares are not uniquely Θ-completable to L. Which is the minimum cardinality of these minimal sets?

Problem 2: Study the minimal hitting sets that intersect any of the minimal sets described in Problem 1.

All the resulting partial Latin squares described in Problem 1 are maximal unauthorized sets in the secret sharing scheme described by L and Θ. Thus, the minimum cardinality therein mentioned corresponds to the maximum cardinality of these maximal unauthorized sets. Furthermore, every minimal authorized set in the scheme must contain at least one of the minimal hitting sets described in Problem 2.

This paper deals with both Problems 1 and 2 and their subsequent implementation into the computation of critical sets based on autoparatopisms. It is organized as follows. Section II deals with some preliminary concepts and results on critical sets based on autoparatopisms that are used throughout the paper. Then, Section III delves into Problems 1 and 2. An illustrative example of secret sharing schemes based on an autoparatopism is described in Section IV.

II. Preliminaries

Let $L \in \mathcal{L}(n)$ and $\Theta \in \mathrm{Apar}(L)$. The *order* of Θ is the smallest positive integer $o(\Theta)$ such that $\Theta^{o(\Theta)}$ is trivial. The Θ-*orbit* of an entry $e \in \mathrm{Ent}(L)$ is the set

$$\mathrm{Orb}_\Theta(e) := \left\{\Theta^k(e) \colon\ 0 \le k \le o(\Theta)\right\} \subseteq \mathrm{Ent}(L). \quad (1)$$

To be in the same Θ-orbit is an equivalence relation among the entries of L, which we denote by $\sim_\Theta$. The set $\mathrm{Orb}_\Theta(L)$ that is formed by all Θ-orbits in $\mathrm{Ent}(L)$ constitutes a partition of $\mathrm{Ent}(L)$ and hence of the cells of L. In particular, $|\mathrm{Orb}_\Theta(L)| = |\mathrm{Ent}(L)/\sim_\Theta|$. This partition is represented by a Θ-*coloring*, where two cells of L are colored with the same color if and only if they are in the same Θ-orbit. Thus, for example, we show here a Θ-coloring of a Latin square having the paratopism

$$\Theta := ((12); (\mathrm{Id}_4, (1234), (12)(34)) \in S_3 \wr S_4$$

as an autoparatopism.

1	3	4	2
2	1	3	4
4	2	1	3
3	4	2	1

Since every Θ-orbit in L is uniquely determined by any of its entries, every Θ-critical set of L has at most one entry in each Θ-orbit of L. Thus, the largest size of every Θ-critical set of L is less than or equal to the number of Θ-orbits of L.

In the secret sharing scheme based on L and Θ, those participants who have entries associated with longer orbits have more information about the key than those with entries in smaller orbits. In this regard, those orbits with a larger number of entries can be assigned to a higher level.

III. Minimal breaking sets

Again, let $L \in \mathcal{L}(n)$ and $\Theta \in \mathrm{Apar}(L)$. In order to deal with Problem 1, we introduce here the following notion.

Definition 1: A *minimal breaking set* (MBS) of Θ-orbits in L is a subset of representative entries $S \subseteq \mathrm{Ent}(L)/\sim_\Theta$ such that the partial Latin square $L_S \in \mathcal{PL}(n)$, which is defined so that

$$\mathrm{Ent}(L_S) := \mathrm{Ent}(L) \setminus \bigcup_{e \in S} \mathrm{Orb}_\Theta(e),$$

is no longer uniquely Θ-completable to L, and no proper subset of S satisfies this condition.

Every MBS associated to the trivial paratopism is a *Latin trade* [3]. Every Θ-critical set of L intersects every MBS of Θ-orbits in L. In the context of a secret sharing scheme based on L and Θ, the following aspects concerning MBSs are remarkable.

- If the set of pieces of information shared by a group of participants does not intersect any MBS, then they will not be able to unequivocally reconstruct the secret. That is, they constitute an unauthorized group of the scheme.

- The distribution of MBSs according to their size can be used to study the *resilience* of the scheme. That is, the capacity to resist the loss or unavailability of some of the distributed pieces of information without compromising the capacity to recover the original secret. More specifically, the following staments hold.
  - The smaller MBSs are, the more vulnerable the scheme is, because very little information would need to be removed to compromise the uniqueness of the secret.
  - The larger MBSs are, the more robust the scheme is against either the loss of small portions of information or unauthorized access to them.

  As a consequence, a distribution of MBSs in which the majority of them are large describes a robust scheme. A concentration of small MBSs is a sign of vulnerability, because it indicates the existence of many pieces of information such that if they were not shared within a group of participants, then this group would be unauthorized. A lower number of MRBs indicates that the scheme could be vulnerable if they were easily accessible.
- The way in which MBSs overlap or are distributed across all Θ-orbits can also be relevant. In this regard, if all MBSs share a core set of Θ-orbits, this core set becomes critical in case of an attack or lack of information. Protecting those orbits would be a priority. Furthermore, a scheme in which MBSs are highly diverse with little overlap is more robust because an attacker would have to identify and eliminate a specific set of orbits from a broader range. That is, the scheme were more resilient to attacks targeting specific orbits.
- MBSs allow the dealer to analyze the security of the scheme, because they are useful to understand the critical collaboration among participants, whose loss or manipulation could lead to ambiguity in reconstructing the secret, or even to the impossibility of reconstructing it.

From now on, we denote by $\mathrm{MBS}(\Theta, L)$ the set of MBSs of Θ-orbits in L. Algorithm 1 describes a procedure that enables us to compute this set. It plays therefore a relevant role to study the resilience of the secret sharing scheme based on L and Θ.

Algorithm 1 Find minimal breaking sets.

```
1:  procedure MinimalBreakingSets(L, Θ)
2:      MBS:= Empty set                          ▷ Set of MBSs.
3:      for t ← 1, |Ent(L)/ ∼_Θ | do
4:          Candidates = {t-subsets in Ent(L)/ ∼_Θ}
5:          for S ∈ Candidates do
6:              if T ⊆ S for some T ∈ MBS then
7:                  break
8:              else
9:                  if L_S is not uniquely Θ-completable then
10:                     MBS← MBS ∪ {S}
11:                 end if
12:             end if
13:         end for
14:     end for
15:     return MBS
16: end procedure
```

In the worst case, if $L \in \mathcal{L}(n)$, then the number of candidates in Algorithm 1 is 2^{n^2}, which arises whenever Θ is trivial. Also in the worst case, the number of distinct minimal breaking sets is $\binom{n^2}{n^2/2}$, so that the pruning described in Line 6 could have a computational complexity $O\left(\binom{n^2}{n^2/2} \cdot n^2\right)$. In any case, the most expensive part of the algorithm concerns the verification of being uniquely Θ-completable. This is known [4] to be NP-complete when Θ is trivial.

In practice, it is not necessary to implement Algorithm 1 for all $L \in \mathcal{L}(n)$ and $\Theta \in \mathrm{Apar}(L)$, but only for a representative Latin square of each main class, and a representative paratopism of each conjugacy class. To prove it, we need the following preliminary lemmas.

Lemma 1: Let $L \in \mathcal{L}(n)$ and $\Theta \in \mathrm{Apar}(L)$. If $\Theta' \in S_3 \wr S_n$, then $\Theta'\Theta\Theta'^{-1} \in \mathrm{Apar}(L^{\Theta'})$.

Proof. The result follows readily from the fact that

$$\left(L^{\Theta'}\right)^{\Theta'\Theta\Theta'^{-1}} = \left(\left(\left(L^{\Theta'}\right)^{\Theta'^{-1}}\right)^{\Theta}\right)^{\Theta'} = L^{\Theta'}.$$

□

Lemma 2: Under the assumptions of Lemma 1, let $e_1, e_2 \in \mathrm{Ent}(L)$. Then,

$$e_1 \sim_\Theta e_2 \Leftrightarrow \Theta'(e_1) \sim_{\Theta'\Theta\Theta'^{-1}} \Theta'(e_2).$$

Proof. First, note that the equivalence relation $\sim_{\Theta'\Theta\Theta'^{-1}}$ is well-defined because of Lemma 2. Now, we have from (1) that $e_1 \sim_\Theta e_2$ if and only if there is a positive integer k such that $e_2 = \Theta^k(e_1)$. Then,

$$\left(\Theta'\Theta\Theta'^{-1}\right)^k (\Theta'(e_1)) = \Theta'\Theta^k\Theta'^{-1} (\Theta'(e_1)) = \Theta'(e_2)$$

and hence, $\Theta'(e_1) \sim_{\Theta'\Theta\Theta'^{-1}} \Theta'(e_2)$. The reciprocal follows similarly. □

Lemma 3: Under the assumptions of Lemma 1, a partial Latin square $P \in \mathcal{PL}(n)$ is (uniquely) Θ-completable to L if and only if $P^{\Theta'}$ is (uniquely) $\Theta'\Theta\Theta'^{-1}$-completable to $L^{\Theta'}$.

Proof. The equivalence concerning the completability follows readily from Lemma 1 and the fact that $\mathrm{Ent}(P) \subseteq \mathrm{Ent}(L)$ if and only if $\mathrm{Ent}\left(P^{\Theta'}\right) \subseteq \mathrm{Ent}\left(L^{\Theta'}\right)$. Furthermore, if P is uniquely Θ-completable, but $\mathrm{Ent}\left(P^{\Theta'}\right) \subseteq \mathrm{Ent}(L')$ for some $L' \in \mathcal{L}(n) \setminus \{L^{\Theta'}\}$ such that $\Theta'\Theta\Theta'^{-1} \in \mathrm{Apar}(L')$, then $\mathrm{Ent}(P) \subseteq \mathrm{Ent}\left(L'^{\Theta'^{-1}}\right)$ and $\Theta \in \mathrm{Apar}\left(L'^{\Theta'^{-1}}\right)$. Since P is uniquely Θ-completable, it must be $L = L'^{\Theta'^{-1}}$ and hence $L' = L^{\Theta'}$, which is a contradiction. That is to say, $P^{\Theta'}$ is uniquely $\Theta'\Theta\Theta'^{-1}$-completable to $L^{\Theta'}$. The reciprocal condition follows similarly. □

The following result holds readily from the previous lemmas.

Proposition 1: If $\Theta' \in S_3 \wr S_n$, then the map

$$\begin{array}{ccc} \mathrm{MBS}(L, \Theta) & \to & \mathrm{MBS}\left(L^{\Theta'}, \Theta'\Theta\Theta'^{-1}\right) \\ S & \mapsto & S^{\Theta'} := \{\Theta'(e) \colon e \in S\} \end{array}$$

is a one-to-one correspondence.

As an illustrative example, and based on the previous proposition, we have implemented Algorithm 1 to determine the distribution of MBSs (and hence the resilience of the corresponding secret sharing scheme) with respect to a representative Latin square of each main class in $\mathcal{L}(n)$ for $n \in \{3, 4\}$, and a representative paratopism of each conjugacy class. These are the representative Latin squares that we have used.

1	2	3
2	3	1
3	1	2

L_3

1	2	3	4
2	1	4	3
3	4	1	2
4	3	2	1

$L_{4.1}$

1	2	3	4
2	1	4	3
3	4	2	1
4	3	1	2

$L_{4.2}$

To avoid the enumeration of specific paratopisms, we carry out our study according to their cycle structures. Recall here that the *cycle structure* of a permutation $\pi \in S_n$ is the expression $z_\pi := n^{\lambda_n^\pi} \dots 1^{\lambda_1^\pi}$, where λ_ℓ^π is the number of cycles of length ℓ in the unique decomposition of π as a product of disjoint cycles. In practice, zero factors are omitted, and every factor of the form ℓ^1 is written ℓ. Thus, for example, $z_{(123)(456)(7)} = 3^2 1$. Then, the cycle structure of a paratopism $\Theta = (\pi; (f_1, f_2, f_3)) \in S_3 \wr S_n$ is the tuple

$$z_\Theta := (z_\pi; (z_{f_1}, z_{f_2}, z_{f_3})).$$

Mendis and Wanless [19] proved that every paratopism is conjugate to an isotopism or a paratopism of one of the following two types.

- **Type I**: $((12); (\mathrm{Id}_n, f_2, f_3)) \in S_3 \wr S_n$.
- **Type II**: $((123); (\mathrm{Id}_n, \mathrm{Id}_n, f_3)) \in S_3 \wr S_n$.

Even more, they determined the cycle structures of all autoparatopisms of types I and II of Latin squares of order up to 17. Those of all autotopisms of Latin squares of the same orders were already known [8], [21].

As an illustrative example, we have made use of Algorithm 1 to show in Tables I, II and III the distribution of MBSs according to their size with respect to each main class of Latin square of order $n \in \{3, 4\}$ and each conjugacy class. We have also indicated the distribution of minimal hitting sets of these MBSs according to their size. This last distribution focuses on the overlaps of MBSs. It is also interesting in the security analysis of the scheme because it establishes the core set of Θ-orbits that becomes critical in case of an attack. Each distribution is represented as an expression $m^{\rho_m} \dots 1^{\rho_1}$, where ρ_k is the number of MBSs of size k. Similarly to cycle structures, zero factors are omitted, and every factor of the form k^1 is written k.

TABLE I
DISTRIBUTION OF MBSS AND THEIR MINIMAL HITTING SETS BY SIZE: ISOTOPISMS.

L	z	MBSs	Minimal hitting sets
L_3	$(1^3, 1^3, 1^3)$	6^9	$3^{34}2^9$
	$(21, 21, 21)$	3^3	$2^3 1$
	$(3, 3, 1^3)$	2^3	2^3
	$(3, 3, 3)$	3	1^3
$L_{4.1}$	$(1^4, 1^4, 1^4)$	4^{12}	$7^{135}6^{511}5^{21}4^8$
	$(2^2, 2^2, 1^4)$	2^4	4^{16}
	$(21^2, 21^2, 21^2)$	42^3	4^{32}
	$(2^2, 2^2, 2^2)$	4^6	$4^2 3^{12} 2^4$
	$(31, 31, 31)$	4^5	2^{10}
	$(4, 4, 21^2)$	2^2	2^4
$L_{4.2}$	$(1^4, 1^4, 1^4)$	4^4	4^{256}
	$(2^2, 2^2, 1^4)$	2^4	4^{16}
	$(2^2, 2^2, 21^2)$	4^6	$3^{18}2^4$
	$(21^2, 21^2, 21^2)$	42^3	4^{32}
	$(4, 4, 2^2)$	2^2	2^4
	$(4, 4, 1^4)$	2^6	3^4

TABLE II
DISTRIBUTION OF MBSS AND MINIMAL HITTING SETS BY SIZE: TYPE I.

L	z	MBSs	Minimal hitting sets
L_3	$(1^3, 1^3, 1^3)$	4^3	2^{12}
	$(1^3, 1^3, 21)$	3	1^3
	$(1^3, 3, 21)$	1	1
	$(1^3, 3, 3)$	2	1^2
$L_{4.1}$	$(1^4, 1^4, 1^4)$	3^6	$6^2 5^4 8^3 3^{10}$
	$(1^4, 1^4, 21^2)$	43^2	3^{36}
	$(1^4, 2^2, 21^2)$	21^2	3^2
	$(1^4, 2^2, 2^2)$	21^2	3^2
	$(1^4, 2^2, 4)$	3^4	2^6
	$(1^4, 31, 31)$	2	1^2
$L_{4.2}$	$(1^4, 1^4, 1^4)$	43^2	3^{36}
	$(1^4, 1^4, 21^2)$	43^2	3^{36}
	$(1^4, 2^2, 21^2)$	21^2	3^2
	$(1^4, 2^2, 2^2)$	21^2	3^2
	$(1^4, 4, 2^2)$	1^2	2
	$(1^4, 4, 4)$	2	1^2

IV. AN EXAMPLE

We finish our study with an example that illustrates the design of a secret sharing scheme based on an autoparatopism of Latin squares.

Example 1: In a bank, there are four employees and one director. To open the safe, the requirement is that there must always be two employees and the director present. Therefore, they decide to divide the safe's key using a secret sharing scheme. The protocol determines that critical sets of three entries will be sought, one for the director and two for two of the employees, who will be the main participants of the scheme. The remaining employees will be considered substitutes for the participating employees, so redundant entries from the orbits used with the other employees will be assigned to them.

As an illustrative example, we can consider the autoparatopism

$$\Theta = (1^3; ((13)(45), (25)(34), (13)(45)))$$

of the following Θ-colored Latin square.

$L \equiv$

1	2	3	4	5
2	1	4	5	3
3	4	5	1	2
4	5	2	3	1
5	3	1	2	4

A Θ-critical set of L of size three is formed by the entries $\{(1, 4, 4), (4, 2, 5), (4, 4, 3)\}$. The entry $(1, 4, 4)$ is given to the director, while each one of the other two entries are respectively given to one of the other two participants. In addition, the third and fourth employees, who are not main participants of the scheme, receive, respectively, the entry $(5, 5, 4)$ and the entry $(5, 3, 1)$. In this way, the director is indispensable in the scheme, but the first employee can be replaced by the third one, and the second one by the fourth one. ◁

TABLE III
DISTRIBUTION OF MBSS AND MINIMAL HITTING SETS BY SIZE: TYPE II.

L	z	MBSs	Minimal hitting sets
L_3	$(1^3, 1^3, 1^3)$	5	1^5
		3	1^3
	$(1^3, 1^3, 21)$	0	0
$L_{4.1}$	$(1^4, 1^4, 1^4)$	$4^5 2^3$	3^{11}
		$6^{16}4$	$3^{38}2^6$
	$(1^4, 1^4, 21^2)$	2^2	2^4
	$(1^4, 1^4, 2^2)$	2	1^2
$L_{4.2}$	$(1^4, 1^4, 1^4)$	42	2^8
	$(1^4, 1^4, 21^2)$	2^2	2^4

V. DISCUSSION AND FURTHER WORK

Latin squares are useful for designing secret sharing schemes with one of them as the key and its entries as shares. Reconstructing the secret may involve pattern searches, which is often less efficient than the polynomial interpolation over a finite field described in [20]. When a sufficient number of shares are gathered, they can reveal enough information to significantly reduce the key space. However, the huge number of Latin squares is a main advantage because they often face interesting scalability and flexibility challenges.

Secret sharing schemes based on Latin squares become enriched once the dealer makes use of a specific symmetry described by a given autoparatopism of the Latin square under consideration to design multilevel schemes that do not depend on the existence of absent but indispensable participants. If the secret is encoded in a Latin square and the scheme is designed so that the recovery of the secret is invariant under a given autoparatopism, this can add a layer of robustness. A successful reconstruction that is not consistent with this symmetry could indicate an error or manipulation. Thus, for instance, if after gathering a subset of shares, they give rise to a partial Latin square that violates the mandatory symmetry, this could indicate that one or more shares have been corrupted or falsified. An attacker would not only need to solve the combinatorial problem of completing the square, but also do so in a way that is consistent with the symmetry.

In order to analyze the vulnerability and robustness of these schemes, we have introduced the notion of a minimal breaking set (MBS) that breaks the unique completability referred to the mentioned symmetry. The computational complexity to determine all MBSs is so costly in general that it is expensive for an attacker to identify minimal information losses that break the uniqueness of the reconstruction. However, this cost can previously be assumed by the dealer to design a robust scheme. Algorithm 1, which we have implemented in this paper to deal with small Latin squares of order up to four, is a first approach in this regard, but its computational complexity is too high to deal with all cases for higher orders. More precisely, the run time for computing all MBSs related to trivial autotopisms becomes excessive or even infeasible. In order to deal with higher orders, we propose here to delve into the following alternative approach as a further work.

Let L and L' be two Latin squares of order n having the same autotopism Θ as a common symmetry. Then, every MBS related to L and Θ must cover the set $D(L, L')$ of cells in which both L and L' differ. So, the problem of computing all MBSs related to L and Θ is equivalent to the set cover problem related to the universe that is formed by all sets $D(L, L'')$, with L'' having Θ as an autotopism. The calculation of all these Latin squares L'' and the solving of this set cover problem are two main questions to deal with. To this end, one can delve into the implementation of branch-and-bound methods, minimal traversal hypergraph algorithms, and/or greedy backtracking heuristics. All of them are well-studied and are more robust and scalable than blindly testing all subsets, as Algorithm 1 does.

ACKNOWLEDGMENT

The authors want to express their gratitude to the referees for their valuable comments which have helped improve this work. This paper has partially been supported by the Research and Innovation Project PPIT-FEDER 2023 *"Modeling small-world, scale-free networks from combinatorial designs based on quasigroup digraphs"*, co-financed by the EU - Ministry of Finance and Public Administration - European Funds - Andalusian Regional Government - Ministry of University, Research and Innovation.

REFERENCES

[1] Adams, P., Bean, R., Khodkar, A.: "A census of critical sets in the Latin squares of order at most six", in: *Ars Combin.*, vol. 68, pp. 203–223, 2003.

[2] Blakley, G.R.: "Safeguarding cryptographic keys", in *1979 International Workshop on Managing Requirements Knowledge (MARK)*, New York, NY, USA, 1979, pp. 313–318.

[3] Cavenagh, N.J.: "The theory and application of Latin bitrades: a survey", in *Math. Slovaca*, vol. 58, n. 6, pp. 691–718, 2008

[4] Colbourn, C.J., Colbourn, M.J., Stinson, D.R.: "The computational complexity of recognizing critical sets", in *Lect. Notes Math.*, vol. 1073, pp. 248–253, 1984.

[5] Cooper, J., Donovan, D., Seberry, J.: "Secret sharing schemes arising from Latin squares", in *Bull. Inst. Combin. Appl.*, vol. 12, pp. 33–43, 1994.

[6] Donovan, D., Cooper, J., Nott, D.J., Seberry, J.: "Latin squares: critical sets and their lower bounds", in *Ars Combin.*, vol. 39, pp. 33–48, 1995.

[7] Donovan, D., Howse, A.: "Critical sets for Latin squares of order 7", in *J. Combin. Math. Combin. Comput.*, vol. 28, pp. 113–123, 1998.

[8] Falcón, R.M.: "Cycle structures of autotopisms of the Latin squares of order up to 11", in *Ars Combin.* , vol. 103, pp. 239–256, 2012.

[9] Falcón, R.M.: "The set of autotopisms of partial Latin squares", in *Discrete Math.*, vol. 313, n. 11, pp. 1150–1161, 2013.

[10] Falcón, R.M.: "Enumeration and classification of self-orthogonal partial Latin rectangles by using the polynomial method", in *European J. Combin.*, vol. 48, pp. 215–223, 2015.

[11] Falcón, R.M., Falcón, O.J., Núñez, J.: "Counting and enumerating partial Latin rectangles by means of computer algebra systems and CSP solvers", in *Math. Methods Appl. Sci.*, vol. 41, n. 17, pp. 7236–7262, 2018.

[12] Falcón, R.M., Johnson, L., Perkins, S.: "A census of critical sets based on non-trivial autotopisms of Latin squares of order up to five", in *AIMS Math.*, vol. 6, no. 1, pp. 261–295, 2021.

[13] Falcón, R.M., Stones, R.J.: "Enumerating partial Latin rectangles", in *Electron. J. Combin.*, vol. 27, n. 2, Paper 2.47, 41 pp. (2020)

[14] González-Regadera, M., Falcón, R.M., Frau, M.D.: "Critical sets of Latin squares based on autoparatopisms". Submitted, 2025.

[15] Hamalainen, C.: "New 2-critical sets in the abelian 2-group", in *J. Combin. Math. Combin. Comput.*, vol. 61, pp. 193–219, 2007.

[16] Hulpke, A., Kaski, P., Östergård, P.R.J.: "The number of Latin squares of order 11", in *Math. Comp.*, vol. 80, n. 274, pp. 1197–1219, 2011.

[17] Kolesova, G., Lam, C.W.H., Thiel, L.: "On the number of 8×8 Latin squares", in *J. Combin. Theory Ser. A*, vol. 54, n. 1, pp. 143–148, 1990.

[18] McKay, B.D., Meynert, A., Myrvold, W.: "Small Latin squares, quasigroups, and loops", in *J. Combin. Des.*, vol. 15, n. 2, pp. 98–119, 2007.

[19] Mendis, M.J.L., Wanless, I.M.: "Autoparatopisms of quasigroups and Latin squares", in *J. Combin. Des.*, vol. 25, n. 2, pp. 51–74, 2017.

[20] Shamir, A.: "How to share a secret", in *Commun. ACM*, vol. 22, n. 11, pp. 612–613, 1979.

[21] Stones, D.S., Vojtěchovský, P., Wanless, I.M.: "Cycle structure of autotopisms of quasigroups and Latin squares", in *J. Combin. Des.*, vol. 20, n. 5, pp. 227–263, 2012.

Slice Property Inference in B5G Networks

Albert Garcia-Llagostera
Universitat Oberta de Catalunya (UOC)
UOC-TECH Research Center, Estudios de Informática, Multimedia y Telecomunicación
agarciall@uoc.edu
CYBERCAT - Center for Cybersecurity Research of Catalonia

Helena Rifà-Pous
Universitat Oberta de Catalunya (UOC)
UOC-TECH Research Center, Estudios de Informática, Multimedia y Telecomunicación
hrifa@uoc.edu
CYBERCAT - Center for Cybersecurity Research of Catalonia

Julián Salas
Universitat Autonoma de Barcelona (UAB)
Departament d'Enginyeria de la Informació i de les Comunicacions
julian.salas@uab.cat
CYBERCAT - Center for Cybersecurity Research of Catalonia

***Abstract*—Federated Learning (FL) enables privacy-preserving collaborative model training by sharing training outputs and keeping data local to clients. However, even without direct access to raw data, model updates can leak sensitive information, making it essential to reassess the privacy guarantees of FL-based applications and to develop effective defenses against such inference risks. In Beyond 5G (B5G) systems, network slice identity may reveal a client's service class, resource profile or operational role, making it a sensitive attribute whose exposure can widen the attack surface for slice-tailored intrusions. This work studies network slice as a novel property-inference attack (PIA) target, showing that an honest-but-curious FL server can infer a client's slice using the model update patterns of a B5G FL-IDS training. Deep Gradient Compression (DGC) is studied as a countermeasure to PIA for the first time and its defensive performance is benchmarked against Differential Privacy (DP), enabling clear comparison of their respective privacy-utility trade-offs under identical experimental conditions. Our study shows client's slice PIA is highly effective (88.62% of accuracy), and success significantly increases as the number of observed rounds grows. DGC protection is directly proportional to compression, while DP reduces attack accuracy on strong privacy budgets, revealing significant privacy-utility tradeoffs.**

***Index Terms*—Federated Learning, Privacy, B5G, Property Inference**

I. Introduction

Federated Learning (FL) has emerged as a revolutionary paradigm for distributed machine learning since its formal introduction [1]. This approach enables multiple clients to collaboratively train a shared global model while retaining their raw data locally. By doing so, FL effectively mitigates fundamental privacy concerns associated with centralized data aggregation. In the standard FL workflow, clients train the model locally, using their respective datasets and fully or partially transmit trained model features (e.g., gradients or weights) to a central aggregation server. The server aggregates and synthesizes these contributions through algorithms such as Federated Averaging (FedAvg) [1] to update the global model, which is then redistributed to participants for subsequent training rounds. Despite this initial promise of privacy preservation, FL's architecture introduces new privacy and security vulnerabilities. Crucially, sharing model updates instead of raw data does not inherently guarantee privacy.

A wide body of literature describes theoretical privacy threats and empirically proves FL inference attack vectors [2][3][4]. Among these, Property Inference Attacks (PIAs) allow an adversarial participant in collaborative learning to infer certain attributes from the training data of an individual or group that are not intended to be revealed and do not explicitly appear on the training data.

The massive and distributed datasets generated within IoT environments, spanning industrial sensors, healthcare monitors or smart city infrastructure, often leave traditional centralized Machine Learning processing infeasible due to privacy regulations, bandwidth limitations, and scalability requirements [5]. FL addresses these constraints by enabling edge devices to learn collaboratively without transferring raw training data, making it especially valuable for security-sensitive applications like Intrusion Detection Systems (IDS) in 5G networks [6]. Moreover, research has shown that FL applied to IoT intrusion detection often outperforms or compares favorably to purely local or centralized methods [7], [8].

One of core B5G key pillars is network slicing, which enables the creation of logically isolated virtual networks on a shared physical infrastructure, delivering unprecedented efficiency and flexibility [9] [10]. The AI-driven, disaggregated architecture of B5G networks introduces new security and privacy vulnerabilities [11]. Each slice generates unique traffic patterns and security requirements, needing specialized monitoring approaches. Knowing which network slice a client belongs to (slice identity) is a sensitive piece of contextual metadata: slices are explicitly provisioned to separate tenants, use-cases and service classes (e.g., eMBB, URLLC, mMTC), and slice inference therefore reveals operational role, service requirements, and user activities, offering significant value to an adversary. The U.S. government ESF analysis of slicing [12] stresses that slices are intended to provide isolation and that "improper network slice management may allow malicious actors to access data from different network slices" and expose slice configurations and users through eavesdropping or misconfiguration.

Despite advances in general PIA literature, slice property inference remains an unexplored vulnerability. This knowledge gap exposes a novel and critical target for PIA against FL applications in B5G IoT, allowing adversaries to deduce critical information about slice configurations, operational parameters, or security postures. Such inferences could facilitate targeted attacks against specific slice types, thereby shattering

the fundamental security promises of network slicing architectures. FL's vulnerability to gradient exploitation, the critical isolation requirements of 5G slicing, and the absence of slice-aware privacy studies on FL creates a critical gap this study aims to resolve.

To address this, our work makes several key contributions. First, to the best of the author's knowledge, this is the first study to systematically evaluate DGC as a privacy-enhancing mechanism against PIA in FL, providing empirical evidence on how gradient sparsification affects property leakage. Second, the empirical demonstration that network slice identity constitutes a vulnerable and learnable property: an honest-but-curious server can successfully infer which B5G slice each client belongs to during FL-IDS training. Third, to contextualize the privacy guarantees and utility impact of DGC, its defensive performance is compared against well-established mechanism - DP, enabling clear comparison of their respective privacy-utility trade-offs under identical experimental conditions.

II. State of the Art

Following the formal presentation of collaborative learning by [1], a discussion about its assumption of improved privacy began. The first FL privacy attack, [10] demonstrated a GAN-based attack targeting classifier Convolution Neuronal Networks (CNNs) on a collaborative learning scenario. On its proposal, a malicious active participant trains a local generator using each round's aggregated global server's model as a discriminator, producing fake class-representative images that influences its local, hence collaborative training. Once the model reaches a threshold accuracy, the fake class representatives may effectively provide private information about such a class to the malicious participant. As well, after the development of membership inference attacks on centralized Machine Learning [13], [2] developed its implementation to collaborative learning, demonstrating both active and passive adversaries can be successful on inferring whether a specific data point was used in a client's training, by observing gradient differences. Later, [4] proposed Deep Leakage of Gradients, proving that gradient updates of cross-entropy models deterministically leak the true label. These early attacks illustrated that even naive gradient sharing in FL can enable deep data reconstruction, raising fundamental privacy concerns in distributed learning.

Back to [2], this relevant work also lay the foundation for PIAs in Federated Learning (FL), where the goal is to infer aggregated or hidden attributes from a client's datasets. This work defines an honest-but-curious client able to exploit gradient updating to infer whether a property was participating on a specific collaborative training round or not. It is also the first work applying shadow model training technique [13] to a collaborative learning privacy attack. Shadow training is widely employed to perform inference attacks, as it allows an adversary to mimic the behavior of the target model in a hidden and controlled environment, extracting valuable information like gradients. More recently, [14] introduced a poisoning-assisted property inference attack (PAPI) in which a malicious client deliberately inverts its local labels to accentuate target attributes' influence on gradients - This approach outperformed state of the art property inference attacks against FL, exploiting the creation of discriminative patterns in successive global updates by deliberately inverting labels. At the same time, [15] proposed the Label Distribution Inference Attack (LDIA), where an FL honest-but-curious server analyses the evolution of output-layer gradients in each client to infer their relative abundance of training classes. Both PAPI and LDIA use shadow training to mimic the behavior of the model with pre-setted class or property abundances, which requires an auxiliary dataset – a representation of data alike or ground-truth the original.

While most attacks have focused on vision or general-purpose datasets, similar privacy threats apply to IoT and future B5G scenarios. PIAs are viable in FL-based IoT systems, even model updates from edge devices can unintentionally reveal hidden attributes, despite blockchain-enhanced privacy protections [16]. However, none of the existing works consider 5G or B5G-specific metadata such as network slice identifiers as a secret property. There is a gap in the literature on whether a curious server could use FL gradients to determine which network slice a client's data belong to. Our work addresses this unexplored vulnerability, investigating how gradient-based property inference might compromise the confidentiality of B5G slice information in federated learning.

III. Materials and Methods

A. Dataset, Preprocessing and Client's Data Distribution

The DoS/DDoS Attack Dataset for 5G Network Slicing [17] is based on a virtualized RAN/UE environment based on UERANSIM [1] and a standalone 5G core implemented with free5GC[2]. A shared control plane (AMF, NRF, and SMFs) and multiple user-plane functions (UPFs) that operate two separate network slices to create the testbed. Using a gNB emulator, user equipments (UEs) are connected to the slices, with various UEs and servers mapped to UPFs . In this configuration, the dataset depict realistic B5G slicing behavior, in which slices have distinct data-plane paths but share core control functions and both have normal and attack traffic [Table I].

For the Benign traffic generation, multiple UEs performs ICMP ping, file transfers, email exchanges and low-rate TCP/UDP across both the slices. The hping3 tool is used to generate high-rate volumetric floods, such as UDP flooding and various TCP-based DoS/DDoS variants (such as SYN, PUSH, FIN, and reset-flag floods), that originate from multiple UEs and are directed towards a designated victim server via the slice UPF. The traffic is captured in pcap format using Wireshark.

The CICFLOWMeter Flow analyser [18] is used to transform raw data into the bidirectional flows. The tool provides about 84 statistical flow features, adding on a slice identifier and binary labelling to distinguish the traffic class, and the final dataset is extracted in CSV format.

To study FL-IDS privacy breaches when training with B5G data and DGC's privacy enhancing and utility trade offs, different training and attacking scenarios were designed. Using the same non-IID distribution of training data among the clients, scenarios are categorized as Control, DGC and

[1]https://github.com/aligungr/UERANSIM?tab=readme-ov-file
[2]https://free5gc.org/

TABLE I
TRAFFIC CLASS DISTRIBUTION ACROSS THE TWO DATASET SLICES.

Slice	Normal Traffic (%)	Attack Traffic (%)
1	88.78	11.22
2	88.88	11.12

DP. DGC group contains four top-k sparsity configurations to study, top *10%*, *top 1%*, top *0.1%* and *top 0.01%*. DP group as well includes five identical configurations differing on privacy budget ε value; $\varepsilon = 0.3$, $\varepsilon = 0.5$, $\varepsilon = 1$, $\varepsilon = 10$ and $\varepsilon = 20$. Control group only contains one configuration, a plain FL training with no privacy enhancement. Shadow training and attacker training configuration is shared between all cases of study, what makes them differ is the way FL training is held. Each of these scenarios, ten in total, are repeated 16 times with different random seeds for robustness purposes. All models were trained with learning-rate(lr) = 0.001, Binary Cross-Entropy (BCE) as loss criteria and batch size = 64. Target IDS model used SGD optimizer, while attacker model used Adam optimizer. Number of training rounds for IDS model (R) is 100, with 2 local training epochs on each client each round before sending trained params. Attacker server observes and analyses IDS training for first B=10,20 or 30 observing rounds.

Prior to any model training or client partitioning, labels were binarized and rows containing NaN or null values were removed. Categorical features were transformed using one-hot encoding. Moreover, features - Source and Destination IPs, Source and Destination Ports and Protocol - leaking private information related to the class type, were deleted from training and slice type feature was hidden from training. Using a Dirichlet distribution $\alpha = 0.2$ for label class imbalance, 50 clients were created, 25 comprising data from slice 1 and 25 comprising data from slice 2. Each client was randomly assigned 2,000 to 5,000 samples, and data was splitted into train and validation data in a 80-20 ratio. In our study, all clients participate in every training round.

For the IDS, we opt for a Multi-Layer Perceptron (MLP), a simple yet effective architecture, consisting of 5 feed-forward layers, each followed by a ReLU activation function and a dropout layer (dropping ratio = 40%). The input shape corresponds with the 76 remaining independent features after preprocessing of the original dataset. The output layer uses a sigmoid activation function, typical for binary classification, yielding a value in the range (0, 1). To evaluate IDS model's performance, the accuracy metric (True Positives/Total Samples) is used. This accuracy metric is calculated each round using training subset and validation subset, generating Target Training Accuracy and Target Validation Accuracy for 100 rounds. After training, the model is evaluated using a test dataset, generating Target Test Accuracy.

B. Threat model

We adopt the widely established honest-but-curious threat model, where the adversary is the Central Aggregation Server. This entity strictly adheres to the prescribed FL protocol, executing operations like gradient aggregation (e.g., FedAvg) without introducing malicious alterations or performing data poisoning. However, the server is "curious" and actively seeks to extract sensitive information from the legitimate communications it processes. To perform such a task, the server have access to gradient or parameter updates, the number of participating clients and target IDS model. The server is assumed to possess general statistical knowledge regarding the training data distribution to protect the aggregation from poisoning attacks [19]. In our study, the server collects 1,000 samples from the same distribution as users' data in advance, as the server's "root" dataset. The attack objective is to infer the private network slice type of the participating clients, which constitutes a critical privacy leakage that could enable targeted attacks based on the slice's characteristics.

The PIA attack is executed using a two-stage process: an initial data generation phase and a classification phase. First, the server generates a large, labeled training dataset by employing shadow training: it trains multiple shadow models using partitioned datasets derived from the root dataset, where the desired property (the slice type) is known and varied for each model. Second, the server trains an attacker classifier on the resulting labeled dataset of shadow model update vectors. Finally, the trained classifier is used to infer the slice type of real clients by analyzing the sequential parameter updates they submit.

Shadow models setup: Using the server's root dataset, $Q = 500$ training shadow datasets (D_q^S) are generated and used to train $Q = 500$ shadow models (M_q^S), mimicking target model's clients behavior. Shadow datasets have a size ranging from 800 to 5,000 samples and follow a Dirichlet $\alpha = 0.6$ distribution, simulating Non-IID data partitions. Specifically, each shadow dataset exclusively contains traffic instances (both attack and normal) from either slice 1 or slice 2. To generate shadow datasets exceeding 1,000 instances with a 1,000-instances root dataset, root data is amplified through duplication. For the $Q = 500$ generated shadow datasets and models, D_q^S is the shadow dataset and M_q^S is the corresponding shadow model, where $q \in \{1, \ldots, Q\}$.

Attack procedure: This attack is based on the assumption that when a model is trained, the changes in the parameters of the output layer are influenced by the training data. By passively analysing these changes and performing shadow training, an attacker can infer the slice property of each client.

Each training round consists of a local training and a global aggregation phase. Having N number of participants to the federated learning and R number of training rounds, we call $M_{n,r}^L$ the model trained locally by client n at round r. We call M_r^G the model aggregated on the server on round r. Therefore, to address only the output layer weights of a model the letter "O" is used; $O_{n,r}^L$ for output layer weights of a locally trained model by client n at round r and O_r^G for output layer weights of an aggregated global model at round r.The update vector $\Delta O_{n,r}$ is defined as the difference between the local output layer weights ($O_{n,r}^L$) and the global output layer weights received at the start of the round (O_{r-1}^G). $\Delta O_{n,r}$ is then flattened as a $1 \times N_{ow}$ (Number of output layer weights) vector.

At the beginning of each round, the server sends its M_r^G, starting with M_0^G, to each of the clients. The clients train the received parameters using their own private data, generating

$M^L_{n,r}$, starting with $M^L_{n,1}$. The trained local models are then sent back to the global server to aggregate using an aggregation protocol. In this step, a normal global server would just aggregate the parameters or gradients and send the aggregated model M^G_1, or gradients back to the clients, repeating the cycle until convergence is achieved. The adversary server described here uses an already existing technique, the shadow training, and the storage of flattened $\Delta O_{n,r}$, starting with $\Delta O_{n,1}$, to extract private information from the clients. The observed $\Delta O_{n,r}$ for a client and range of rounds, for example $\Delta O_{1,1}$ - $\Delta O_{1,20}$ are the test samples used to later infer the slice property with a trained attacker model, shaped as B (B= n of attacker's observing rounds)x N_{ow} , i.e. 20xN_{ow} . Shadow training is used to generate training samples so the attacker model can learn to differentiate between Slice 1 and Slice 2 clients $\Delta O_{n,r}$ updates.

Meanwhile models are on local training phase of round r, training M^G_r into $M^L_{n,r+1}$, starting with r=0, the Server trains M^G_0 with all its D^S into $M^S_{q,r+1}$, generating Q shadow models. When the clients send their $M^L_{n,r+1}$ to the server to start the aggregation phase, on the one side, the server aggregates all the N clients local models as normal, generating M^G_{r+1}. On the other side, the server also aggregates all the participants' $M^L_{n,r+1}$ with every $M^S_{q,r+1}$ by separate, mimicking each $M^S_{q,r+1}$ as an additional client, generating Q shadow global models for that round. $\Delta O^S_{q,r}$ is the difference between a $M^S_{q,r+1}$ output layer weights - $O^S_{q,r+1}$ - and its subsequent $O^G S_{q,r+1}$. This will generate Q B xN_{ow} shaped training shadow samples. The attacker CNN will use these Q samples and it's known membership to slice 1 (label=0) or slice 2 (label = 1) as a label to train. If the searched property, in this case slice, is influencing the output layer weights and client's shared gradients are not protected in some way, once trained, the attacker should be able to infer the slice property from the real clients' parameter updating.

The attacker model is a 2D CNN tailored to process gradient update sequences. It first applies two convolutional layers (input $\rightarrow$ 16 and 16 $\rightarrow$ 32 channels, kernel size 3) to extract local structural patterns. The resulting feature map is flattened and passed through a 32-unit fully connected layer with ReLU activation and 0.3 dropout, followed by a final linear layer and a sigmoid function to output a binary probability representing the inferred client property. To evaluate attacker model performance, accuracy and loss metrics are used. Attacker's training data is split into training and validation with a 90-10 ratio. Each of the training rounds, training loss and accuracy and validation train and accuracy is calculated. We've observed that using lowest validation loss on the last 100 round model significantly improves test accuracy in some scenarios, while not affecting the remainder scenarios. After testing the attacker model, we obtain the attacker test accuracy.

C. Privacy Techniques

Deep Gradient Compression (DGC) is a communication-efficient optimization technique designed to mitigate the bandwidth bottleneck inherent in distributed and federated learning. Introduced by [20], DGC reduces the volume of gradient information transmitted between clients and the central server during model synchronization by exploiting the inherent sparsity and redundancy in gradient updates. Clients integrate DGC into its local training by applying top-k sparsification with error-feedback. After each backward pass, the raw gradients (g) are accumulated with a persistent residual buffer, and only the largest $k = \text{sparsity} \cdot |g|$ entries are transmitted. Non-selected components are stored as residuals and reinjected in subsequent iterations, following the DGC error-feedback rule. Momentum masking is applied to ensure consistency between the sparsified gradient and the optimizer state. This mechanism substantially reduces the amount of gradient information sent to the server while preserving convergence behaviour.

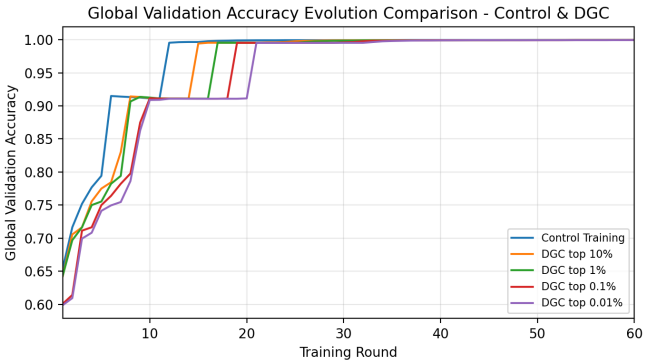


Fig. 1. Comparison of the evolution of global training accuracy among training with no privacy (Control) and training using DGC at different spar values.

Differential privacy is performed by adding calibrated noise into model updates, following the paradigm of additive noise mechanisms. For this purpose, the Opacus library was used. The mechanism was configured with $\varepsilon \in \{0.3, 0.5, 1, 10, 20\}$, target $\delta = 1e-3$ and max clipping $(C) = 10.0$, as we observed that this value of C is suitable for both utility and strong privacy, maintaining a fraction clipping of 5-25%. This corresponds to an implementation of DP-SGD: local gradients computed by each client are clipped to limit their L2 norm to 10.0, after which Gaussian noise is added to the model updates in accordance with the desired (ε, δ) privacy parameters.

IV. Results

A. Training of IDS Model

As a first assessment, it is noticeable how although using a fair volume of data (201,912 instances distributed across 50 clients on a non-IID setting), all control and DGC trainings reached 99.9% of validation accuracy before round 22, see Figure1. This points out that such training is a fairly easy task for our IDS MLP. Control scenario has the fastest learning model, reaching the 99% at round 12. Next, experiments where clients are using DGC with an sparsity of top 10% achieve the second better learning results, with a 99% of accuracy at round 17. Following, experiments where clients used DGC with an sparsity of top 1% and top 0.1% reached plateau validation accuracy at round 19 and round 21 respectively. It is relevant to acknowledge that there's no perceivable difference on final IDS test accuracy on control and DGC experiments all of them achieving 99.98% test accuracy .

Regarding the IDS training of the IDS using DP, there is a clear difference with control training, see Figure 2. While all

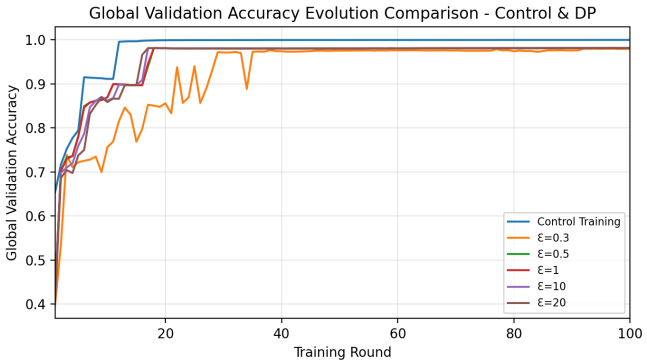


Fig. 2. Comparison of the evolution of global training accuracy among training with no privacy (Control) and training using Differential Privacy (DP) at different ε values.

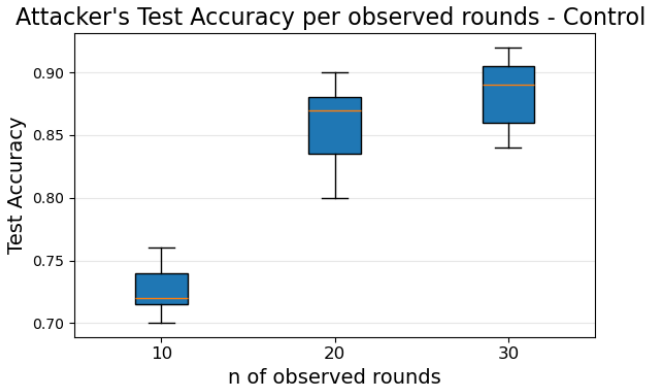


Fig. 3. Box-and-whisker plot of the attacker's test accuracy for 10, 20, and 30 observed rounds. Each box shows the interquartile range (Q1–Q3), the orange line denotes the median (Q2), and the whiskers extend to the most extreme values within 1.5×IQR of the quartiles.

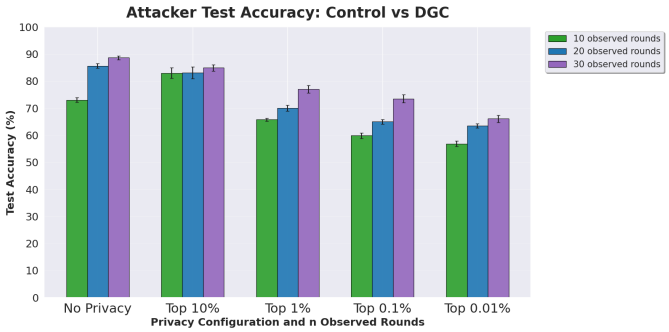


Fig. 4. Average Attack Test Accuracy on different DGC configurations and control for 10, 20 and 30 observed rounds. Spar values for DGC are shown on x axis, while observed rounds for each privacy configuration are color coded columns (green for obv=10, blue for obv=20 and magenta for obv = 30).

Opacus configurations reach a high global validation accuracy ($> 97\%$), these are the only experiments in our study that do not reach 99,9% for this metric after 100 rounds of training. DP-Opacus experiments with ε=0.5 and ε=1 reach its plateau at round 18, while ε=10 and ε=20 reach it at round 19. The most notorious difference comes with ε=0.3 reaching its plateau at round 37. It is worth noting that all experiments, regardless of the ε value, reached practically the same final test accuracy, being 97.2% for ε=0.3, 97.8% for ε=0.5 and ε=1 and 98.3% for ε=10 and ε=20.

B. Attack Performance

1) Control Scenario: After training target IDS models, SDIA attack was performed using 16 different random seeds to ensure robustness. First of all, regarding control training, the attacker successfully inferred the client's slices [Figure 3], reaching an average test accuracy of 88.62%, Standard Error (σ) =2.70 when observing 30 rounds of the target IDS training. When observing 20 rounds, the attack model obtains an average test accuracy of 85.63%, σ = 3.36, and when observing for 10 rounds it obtains an average test accuracy of 73.00%, σ=3.50. To verify the significance of the effect of "observing rounds" setting in the average test accuracy, a Mann-Whitney test was performed to make pairwise comparisons. All comparisons resulted in $p < 0.05$, which shows a significant direct relation between the n of observed rounds and the accuracy of the attacker.

2) Privacy Scenario: After proving the need of the implementation of extra privacy-preserving techniques into the FL training of an IDS to protect client's slice property, this part show attack mitigation results of DGC and DP. As seen on 4, DGC has proven to reduce attack test accuracy when compared to control in all situations except top 10%. There's also an indirect correlation between the compression and the attack accuracy; the higher the compression, the lower the attack accuracy. For obv=30, the average attack test accuracy is 84.87%, 77.00%, 73.50% and 66.13% for spar top 10%, 1%, 0.1% and 0.01% respectively. Furthermore, there's also a direct correlation between n of observed rounds and test accuracy, the later improving a mean of 6.71% ± 2.43 and 7.33% ± 2.54 when comparing obv=10 with n= 20 and obv=20 with obv=30 results respectively. The most privacy-preserving result belongs to top 0.01% obv = 10, with average attack test accuracy of 56.88%, σ=3.90.

After performing the Slice inference attack to federated IDS training using Opacus module, results are plotted on 5. All the DP configurations have shown to improve privacy against Slice Inference Attack, when compared to its homologous with control. For clients using DP with $\varepsilon \leq 10$, all attack test accuracies are below below 53,00%. In the lowest privacy budget scenario, ε=20, the accuracy of the attack increases in a significant way, with 62,33%, 69.25% and 73.37% of test accuracy for 10,20 and 30 observed rounds respectively. Moreover, at ε=20, when privacy budget seems to start being too low, the direct correlation between the number of observed rounds and the attack accuracy appears once again, just like in the DGC scenarios.

V. Conclusions

The conducted experiments demonstrate that the federated training of an intrusion detection system (IDS) can achieve high performance and stable convergence even under heterogeneous data distributions. The global model rapidly reached near-perfect validation accuracy across all scenarios, indicating that the underlying classification task was well suited for the proposed MLP architecture. When privacy-preserving mechanisms were introduced, privacy-efficiency trade-offs emerged in proportion to the difficulty of the task.

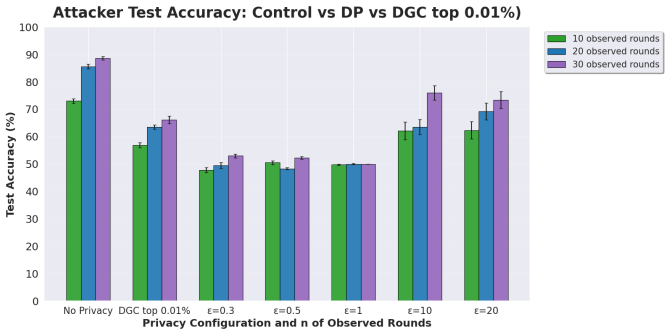


Fig. 5. Average Attack Test Accuracy for best DGC configuration, all DP configurations and control for 10, 20 and 30 observed rounds. Spar values for DGC and ε values for DP are shown on x axis, while observed rounds for each privacy configuration are color coded columns (green for obv=10, blue for obv=20 and magenta for obv = 30).

The use of Deep Gradient Compression (DGC) and Differential Privacy (DP) both slowed down convergence and slightly reduced the final validation accuracy, with the degradation being more pronounced under stronger privacy conditions. In particular, DP-Opacus introduced notable accuracy loss (2%) for small ε values.

Regarding the slice inference attack, the results confirmed that, in the absence of privacy defenses, the adversarial server can successfully infer sensitive client's slice property with high accuracy, achieving test accuracy of 88.62% σ =2.70 at 30 observed rounds Fig. 3. The attack's success increased significantly with the number of observed communication rounds, underlining the cumulative privacy risk inherent to prolonged federated training.

When privacy mechanisms were applied, DGC effectively reduced the leakage of client information, showing a consistent decrease in attack accuracy as the compression level increased 4. DP also reduced the attacker's success rate across all ε values, improving privacy enhancement compared to the best DGC scenario, regardless of the observing rounds configuration. Overall, the study reveals that while DGC and DP mechanisms can mitigate property inference risks, their protective effect depends on parameter tuning and adversarial observation of training rounds.

Acknowledgments

This work was supported by the Spanish Ministry of Science and Innovation through the projects PID2021-125962OB-C31 "SECURING/CYBER", PID2024-156914OB-C41 "SAFE/CYBER", PID2021-125962OB-C33 "SECURING/NET" and PID2024-156914OB-C43 "SAFE-BLOCKCHAIN". Additional funding was provided by the ARTEMISA International Chair of Cybersecurity (C057/23) and the DANGER Strategic Project of Cybersecurity (C062/23), both funded by the Spanish National Institute of Cybersecurity through the European Union NextGenerationEU and the Recovery, Transformation, and Resilience Plan; and the Catalan Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) grant SGR2021-00643.

References

[1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," *20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017*, 2017.

[2] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE Symposium on Security and Privacy (SP)*, May 2019, pp. 691–706.

[3] A. Blika, S. Palmos, G. Doukas, V. Lamprou, S. Pelekis, M. Kontoulis, C. Ntanos, and D. Askounis, "Federated learning for enhanced cybersecurity and trustworthiness in 5G and 6G networks: A comprehensive survey," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 3094–3130, 2025.

[4] L. Zhu and S. Han, "Deep leakage from gradients," *33rd Annual Conference on Neural Information Processing Systems, NeurIPS 2019* 2020, pp. 17–31.

[5] E. Dritsas and M. Trigka, "Federated learning for IoT: A survey of techniques, challenges, and applications," *Journal of Sensor and Actuator Networks*, vol. 14, no. 1, 2025.

[6] T. E. T. Djaidja, B. Brik, A. Boualouache, S. M. Senouci, and Y. Ghamri-Doudane, "Federated learning for 5G and beyond: A blessing and a curse—An experimental study on intrusion detection systems," *Computers & Security*, vol. 139, p. 103707, 2024. [Online].

[7] S. A. Rahman, H. Tout, C. Talhi, and A. Mourad, "Internet of things intrusion detection: Centralized, on-device, or federated learning?" *IEEE Network*, vol. 34, no. 6, pp. 310–317, 2020.

[8] M. M. Rashid, S. U. Khan, F. Eusufzai, M. A. Redwan, S. R. Sabuj, and M. Elsharief, "A federated learning-based approach for improving intrusion detection in industrial internet of things networks," *Network*, vol. 3, no. 1, pp. 158–179, 2023.

[9] J. Dias, P. Pinto, R. Santos, and S. Malta, "5G network slicing: Security challenges, attack vectors, and mitigation approaches," *Sensors*, vol. 25, no. 13, 2025.

[10] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the GAN: Information leakage from collaborative deep learning," *Proceedings of the ACM Conference on Computer and Communications Security*, 2017, pp. 603–618.

[11] H. Rifà-Pous, V. Garcia-Font, C. Núñez-Gomez, J. Salas *et al.*, "Security, trust and privacy challenges in AI-driven 6G networks," in *CS & IT Conference Proceedings*, vol. 14, no. 14, 2024.

[12] U.S. Department of Defense, "Potential threats to 5G network slicing," Tech. Rep., Dec. 2022. [Online]. Available: https://media.defense.gov/2022/Dec/13/2003132073

[13] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *IEEE Symposium on Security and Privacy*, 2017, pp. 3–18. [Online].

[14] Z. Wang, Y. Huang, M. Song, L. Wu, F. Xue, and K. Ren, "Poisoning-assisted property inference attack against federated learning," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 4, pp. 3328–3340, 2023.

[15] Y. Gu and Y. Bai, "LDIA: Label distribution inference attack against federated learning in edge computing," *Journal of Information Security and Applications*, vol. 74, 2023. [Online].

[16] M. Shen, H. Wang, B. Zhang, L. Zhu, K. Xu, Q. Li, and X. Du, "Exploiting unintended property leakage in blockchain-assisted federated learning for intelligent edge computing," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2265–2275, 2021.

[17] M. S. Khan, B. Farzaneh, N. Shahriar, and M. M. Hasan, "DoS/DDoS attack dataset of 5G network slicing," *IEEEDataPort* 2023.

[18] A. Habibi Lashkari, "CICFlowMeter-v4.0: A network traffic biflow generator and analyser for anomaly detection," *GitHub Repo: https://github.com/ISCX/CICFlowMeter* Aug. 2018.

[19] X. Cao, M. Fang, J. Liu, and N. Gong, "FLTrust: Byzantine-robust federated learning via trust bootstrapping," *Network and Distributed System Security Symposium* Jan. 2021.

[20] Y. Lin, S. Han, H. Mao, Y. Wang, and W. Dally, "Deep gradient compression: Reducing the communication bandwidth for distributed training," *International Conference on Learning Representations, ICLR* 2018.

Comunicación segura de vehículos usando criptografía basada en identidades

Pablo Escapa Gordón
Grupo de Robótica. Instituto I4, Universidad de León.
Campus de Vegazana S/N, 24007 León, España
pescg@unileon.es

Vicente Matellán Olivera
Grupo de Robótica. Instituto I4, Universidad de León.
Campus de Vegazana S/N, 24007 León, España
vicente.matellan@unileon.es

Adriana Suárez Corona
Departamento de Matemáticas, Universidad de León.
Campus de Vegazana S/N, 24007 León, España
asuac@unileon.es

***Resumen*—Este es un resumen extendido de la publicación [1]. Establecer una comunicación segura entre vehículos es esencial para el desarrollo de automóviles autónomos y para hacer posibles nuevas funcionalidades, como la gestión inteligente del tráfico o los sistemas destinados a evitar o mitigar los accidentes. Para estos escenarios, en los que el despliegue de infraestructuras de clave pública (*Public Key Infrastructure* (PKI)) puede resultar difícil, se propone el uso de la criptografía basada en identidades como alternativa, puesto que simplifica el cifrado al permitir la generación dinámica de claves y la comunicación segura, sin necesidad del uso de certificados, lo que lo hace altamente escalable y eficiente. Se propone un protocolo de comunicación aplicable a las comunicaciones V2V (vehículo a vehículo), sustituyendo los esquemas basados en PKI por esquemas criptográficos basados en identidades que utilizan el VIN (*Vehicle Identification Number*) como identificador inequívoco del vehículo. Además, se describe una implementación prototipo en un vehículo convencional y las métricas de rendimiento del sistema. El sistema propuesto proporciona una identificación de vehículos segura y flexible, con múltiples aplicaciones prácticas.**

***Index Terms*—Criptografía basada en identidades, V2V, automoción**

I. Introducción

Los componentes principales de los vehículos modernos ya no son los mecánicos o los hidráulicos, sino los ordenadores y sistemas de comunicaciones, pareciéndose cada vez más a robots. Tecnologías como los sistemas avanzados de asistencia al conductor (ADAS) [2] y la conectividad externa mejoran la experiencia del conductor y los pasajeros, y también pueden permitir la conducción autónoma y la gestión inteligente del tráfico. En este sentido, se puede considerar que los coches son plataformas hiper-conectadas y con sensores avanzados, con tres objetivos para el transporte de personas: cero accidentes, cero contaminación y cero atascos.

Para alcanzar estos objetivos hay tres elementos fundamentales —conectividad, seguridad y protección— que son indispensables para el desarrollo y la implantación de los vehículos autónomos. En particular, la comunicación vehicular supone una gran revolución, ya que el intercambio de información entre vehículos (comunicaciones V2V), con la infraestructura (V2I) (vehículo a infraestructura) o, en general, con otros agentes del tráfico (V2X), podría ayudar a prevenir los 1,35 millones de muertes causadas por accidentes de tráfico en todo el mundo [3] y reducir los casi 7 millones de muertes causadas por la contaminación [4]. Sin embargo, la industria automovilística debe garantizar la seguridad de estas comunicaciones para evitar su uso indebido. La identificación errónea de los vehículos puede dar lugar a decisiones de conducción erróneas, lo que se traduce en riesgos de seguridad y posibles accidentes.

El objetivo del artículo aquí resumido es diseñar y evaluar un protocolo de comunicación seguro para redes vehiculares que sustituya los esquemas tradicionales basados en PKI por protocolos criptográficos basados en identidades, utilizando el número de identificación del vehículo (VIN) como identificador único. Las principales contribuciones incluyen un protocolo de comunicación basado en identidades genérico, el uso del VIN como identificador seguro y práctico para la generación dinámica de claves en redes vehiculares, una implementación de prototipo en el mundo real que demuestra la viabilidad del enfoque propuesto en un vehículo convencional y un análisis de métricas de rendimiento que destaca el impacto de las capacidades de procesamiento del hardware en las operaciones de cifrado y descifrado.

El uso de esquemas criptográficos en estos protocolos de comunicación permite la autenticación, la comunicación confidencial y evita que se modifique la información intercambiada. Algunos protocolos seguros de comunicaciones V2X tradicionales basados en PKI proponían descentralizar las tareas de la autoridad de certificación (CA) para reducir la sobrecarga de comunicación y computación, así como la dependencia de una CA centralizada.

La criptografía basada en identidades, introducida por Shamir [5], es una alternativa a la criptografía de clave pública que evita el uso de certificados y la gestión y actualización de las listas de revocación de certificados (CRL). En este caso, cadenas arbitrarias, denominadas identidades, pueden desempeñar el papel de claves públicas. Éstas deben identificar de forma inequívoca a los usuarios, como las direcciones de correo electrónico o los números de la seguridad social en el ámbito de las comunicaciones personales. En el contexto de la comunicación vehicular, el VIN o número de bastidor, que identifica de forma única a los vehículos, es un candidato ideal para construir estos identificadores ya que, mientras que el número de matrícula puede cambiar si el vehículo se vuelve a matricular, por ejemplo, en otro país, el número de

identificación del vehículo es único.

En la literatura se pueden encontrar numerosos sistemas criptográficos basados en identidades, incluyendo esquemas de cifrado, firma digital y protocolos de establecimiento de claves. Algunos protocolos de comunicaciones V2V utilizan este tipo de esquemas [7], [8], [9], [10]. Sin embargo, la mayoría de estas propuestas no especifican qué información se utiliza como identidad. Además, ninguno de ellos ha utilizado el VIN en formato digital como identificador de automóviles.

Andreica et. al [8] utilizan matrículas como identidades. Éstas son reconocidas primero por las cámaras de los teléfonos inteligentes ubicadas en los vehículos. Groza et al. [11] utilizan el VIN para el control de acceso de vehículos con un teléfono inteligente. Sin embargo, reconocen el número de identificación del vehículo utilizando un teléfono inteligente y no utilizan su versión digital [12].

En la sección II, presentamos el protocolo genérico propuesto y su instancia utilizando [6]. En la sección III describimos brevemente la parte experimental y presentamos los resultados obtenidos. El artículo concluye con la sección IV que presenta las conclusiones y trabajo futuro.

Todos los detalles se pueden consultar en [1].

II. Protocolo basado en identidades para comunicaciones V2V

En esta sección se describe un protocolo genérico de comunicación entre dos vehículos que garantiza la confidencialidad, autenticación e integridad de los datos intercambiados.

Este protocolo se basa en un esquema de establecimiento de claves basado en identidades de dos partes con una sola ronda de comunicación. El protocolo utiliza el número de identificación digital del vehículo como identidad. Además, se utilizará un esquema AEAD (cifrado autenticado con datos asociados) [13].

Antes de comenzar la comunicación, se realiza una fase de configuración previa, en la que se definen los parámetros públicos del sistema y la clave privada del Centro Generador de Claves (KGC). Este centro, a partir del número de bastidor (VIN) de cada vehículo y de su información secreta, calcula la clave privada correspondiente ($SK_{VIN_{veh}}$) y se la entrega a cada unidad vehicular.

A continuación, se produce un intercambio de mensajes entre los dos vehículos, correspondientes al protocolo de intercambio de claves de una ronda con autenticación basada en sus VIN digitales. El vehículo A envía el primer mensaje. A continuación, el vehículo B añade un código de autenticación de mensajes (MAC) al mensaje enviado, calculado utilizando la clave común generada a partir del mensaje enviado por el vehículo A. El intercambio concluye con el envío por parte del vehículo A al vehículo B del resultado del MAC calculado sobre todos los mensajes intercambiados, utilizando la clave común, actuando como confirmación de la clave.

El flujo de mensajes del esquema genérico puede verse en la Figura 2.

Una vez que los vehículos han compartido la clave común, emplean un esquema de cifrado autenticado (AEAD) para comunicarse de forma segura.

Este protocolo es seguro para cualquier esquema de establecimiento de clave basado en identidades de una sola ronda de comunicación entre dos partes. Dado un esquema de este tipo,

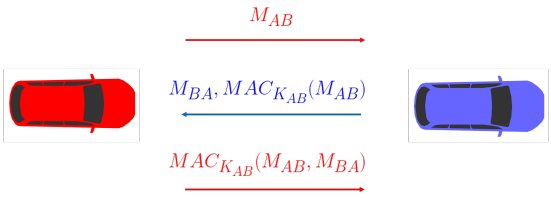


Figura 1. Flujo de mensajes del protocolo de comunicación V2V genérico.

el protocolo propuesto garantiza confidencialidad, integridad y autenticación.

En la implementación de la prueba de concepto, el protocolo se instancia utilizando el esquema de establecimiento de claves basado en identidades de Fiore y Gennaro [6]. Además, el MAC utilizado fue HMAC-SHA256.

El protocolo propuesto hereda la seguridad de estas construcciones; por tanto, su seguridad se basa en la Hipótesis de Diffie–Hellman Fuerte en el Modelo del Oráculo Aleatorio. Este protocolo es altamente eficiente, ya que no utiliza aplicaciones bilineales, lo que lo hace ideal para escenarios vehiculares.

Consta de los siguientes algoritmos:

Set up: El Centro Generador de Claves (KGC) selecciona un grupo finito G de orden primo q, un generador aleatorio del grupo $g \in G$, y un exponente $x \in Z_q$. Se seleccionan también dos funciones hash $H_1 : \{0,1\}^* \rightarrow Z_q$ y $H_2 : Z_q \times Z_q \rightarrow \{0,1\}^\ell$. Los parámetros públicos son $MPK = (G, q, g, y = g^x, H_1, H_2)$, y la clave maestra secreta será el exponente x.

Key Derivation: Para calcular la clave privada de un usuario con identidad ID, el KGC la calcula como la firma de Schnorr's de ID con la clave pública y, es decir, $r_{ID} = g^{k_{ID}}$, donde $k_{ID} \in Z_q$ y $s_{ID} = k_{ID} + xH1(ID; r_{ID}) \mod q$.

Key Agreement: A y B eligen exponentes privados efímeros $t_A, t_B \in Z_q$.

El flujo de comunicación del protocolo, instanciado con [6] y utilizando el VIN como identidad, se muestra en la Figura 2. Aquí se incluye, además, a la autoridad central que genera las claves privadas correspondientes a cada identidad.

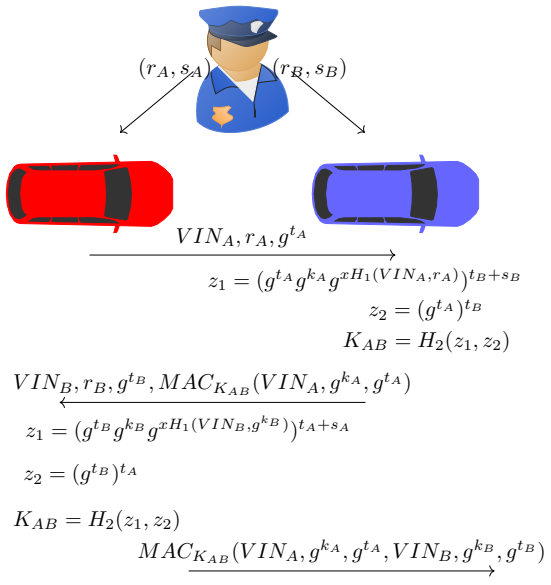


Figura 2. Protocolo de comunicación V2V basado en [6].

Posteriormente, la clave establecida se usa en un esquema AEAD, concretamente AES-256 en modo Galois Counter (AES-256 GCM), elegido por su alto rendimiento, bajo consumo de recursos, eficiencia y elevado nivel de seguridad.

III. Validación Experimental

En este apartado se describen dos escenarios diferentes en los que se ha implementado la comunicación vehicular. El vehículo 1 era un BMW X1 con VIN WBAVN71060VX14326 y equipado con una Raspberry Pi 3 y un módulo PiCAN. Este conjunto formaba la OBU integrada en el primer vehículo, dotándolo de aplicaciones adicionales como protocolos de cifrado y conectividad.

El segundo vehículo estaba equipado con un ordenador portátil Asus que conectaba el vehículo BMW 320D con número de bastidor WBAUT31010F150820 a través de la interfaz CANalyze.

Se han llevado a cabo los siguientes experimentos:

Primer experimento – Sistema anticolisión. Se utiliza la OBU (*On Board Unit*) del vehículo 1 para leer parámetros del CAN BUS, como la velocidad y el ángulo de dirección, y transmitirlos al vehículo 2, que se mueve en sentido contrario. El procesamiento de esta información podría utilizarse para desarrollar un sistema de prevención de colisiones.

Segundo experimento – Adelantamiento. El vehículo 2 solicita la posición y velocidad del vehículo 1 mediante la OBU, que obtiene la información por el GPS o directamente del CAN BUS. Aunque esos datos no son suficientes para decidir un adelantamiento, combinándolos con otros (potencia, caja de cambios, etc.) se podría determinar su viabilidad.

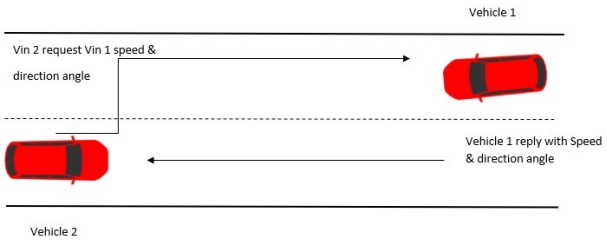


Figura 3. Primer experimento: sistema anticolisión.

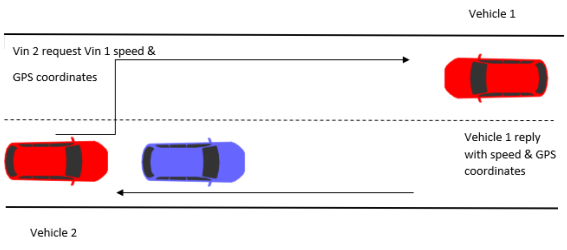


Figura 4. Segundo experimento: adelantamiento.

Probamos el protocolo propuesto en ambos escenarios. En el primero, ambos coches estaban parados y había cinco metros entre ellos, y en el segundo, el primer coche estaba parado y el segundo se movía a 5 km/h. Esto nos permitió medir la sobrecarga de cálculo al enviar varios mensajes (velocidad, temperatura, etc.).

El tiempo obtenido fue el tiempo total necesario para enviar información, como la velocidad, desde el primer vehículo al segundo. Esto incluyó el tiempo necesario para la ejecución del esquema criptográfico, el retraso en la comunicación y el tiempo de procesamiento en ambos vehículos.

En las Tablas I y II se muestra el tiempo de procesamiento de los vehículos en ambos experimentos para cada una de las ejecuciones. Como se esperaba, el portátil Asus con procesador de 3.2 GHz resultó ser mucho más rápido que la Raspberry Pi.

Vehículo que inicia	Vehículo que recibe	Conexión establecida	Tiempo de procesamiento de A en segundos	Tiempo de procesamiento de B en segundos	Iteración
A	B	OK	0,0982	0,2812	1
A	B	OK	0,0987	0,2814	2
A	B	OK	0,0989	0,2821	3
A	B	OK	0,0995	0,2799	4
A	B	OK	0,0988	0,2788	5
A	B	OK	0,0983	0,2822	6
A	B	OK	0,0981	0,2718	7
A	B	OK	0,0986	0,2819	8
A	B	OK	0,0982	0,2787	9
A	B	OK	0,0983	0,2813	10

Tabla I
Tiempo de ejecución a 0 Km/h.

Vehículo que inicia	Vehículo que recibe	Conexión establecida	Tiempo de procesamiento de A en segundos	Tiempo de procesamiento de B en segundos	Iteración
A	B	OK	0,0985	0,2842	1
A	B	OK	0,0991	0,2818	2
A	B	OK	0,0992	0,2832	3
A	B	OK	0,0996	0,2877	4
A	B	OK	0,1001	0,2901	5
A	B	OK	0,0982	0,2845	6
A	B	OK	0,0983	0,2721	7
A	B	OK	0,0988	0,2822	8
A	B	OK	0,0988	0,2791	9
A	B	OK	0,0987	0,2845	10

Tabla II
Tiempo de ejecución a 5 Km/h

Asimismo, se observó que la velocidad del vehículo influye en el tiempo de procesamiento del protocolo. Este incremento es causado por las características del canal de transmisión y de los fenómenos físicos que afectan a la propagación de las ondas electromagnéticas empleados por las tecnologías 5G y IEEE 802.11p. No obstante, el impacto de este efecto es limitado y puede mitigarse para cumplir con los requisitos de latencia especificados.

Las pruebas demostraron que el protocolo funciona correctamente. Las Tablas III y IV presentan el promedio, la varianza, la desviación estándar y el coeficiente de variación de las métricas medidas en ambos experimentos.

El tiempo obtenido corresponde al tiempo total necesario para enviar información (por ejemplo, la velocidad) desde el

Medida	A	B
Media	0.09856	0.27993
Varianza	1.684E-07	8.8281E-06
Desviación típica	0.000410366	0.002971212

Tabla III
TIEMPOS DE PROCESAMIENTO EN SEGUNDOS A 0 KM/H

Medida	A	B
Media	0.09893	0.28294
Varianza	3.121E-07	2.14544E-05
Desviación típica	0.000558659	0.004631889

Tabla IV
TIEMPOS DE PROCESAMIENTO EN SEGUNDOS A 5 KM/H

primer vehículo al segundo. Este incluye el tiempo necesario para la ejecución del protocolo criptográfico, el retardo de comunicación y el tiempo de procesamiento en ambos vehículos.

Los resultados muestran que los sistemas funcionaron correctamente en las diez ejecuciones realizadas. Se puede observar que el rendimiento promedio es bueno para aplicaciones V2V contando con una desviación estándar muy baja.

IV. CONCLUSIONES Y TRABAJO FUTURO

El empleo de la criptografía basada en identidades demuestra ser una solución idónea para las comunicaciones vehiculares eliminando la tediosa tarea que supone la creación de certificados de sistemas basados en criptografía de clave pública (PKI). El protocolo propuesto se basa en el VIN (Vehicle Identification Number) o bastidor digital. El uso de éste simplifica la autenticación y supone un avance logrando identificar a los vehículos inequívocamente al igual que su homólogo en formato tradicional. Esto proporciona y habilita diversas aplicaciones de gran utilidad.

Las pruebas de concepto han demostrado que los tiempos de procesamiento son bajos, requiriendo poca carga computacional, confirmando la idoneidad de uso para entornos vehiculares que requieren del manejo de datos en tiempo real. Los resultados demuestran que este método reduce la complejidad de la gestión de certificados sin sacrificar la seguridad ni el rendimiento.

El protocolo puede ser generalizado para su uso en múltiples ámbitos destacando la autenticación digital de los vehículos, la automatización del pago en peajes, gasolineras o kioscos sin hardware adicional, el intercambio de información para la gestión del tráfico y la seguridad vial o la transmisión de datos con talleres.

Sería interesante desarrollar mecanismos para evitar la falsificación del VIN digital, probar la capacidades el protocolo con nuevas tecnologías de red (como 5G y 802.11p) y adaptarlo a nuevos requerimientos de latencia.

AGRADECIMIENTOS

Adriana Suárez Corona ha sido financiada parcialmente por el proyecto de investigación PID2021-123461NBC22, financiado por el Ministerio de Ciencia, Innovación y Universidades de España. Parte del desarrollo de este trabajo fue financiado por INCIBE (Instituto Nacional de Ciberseguridad) en el marco del Reto 2 del proceso de Contratación Pública Precomercial CPP001/23, correspondiente a servicios de «I+D+i» en el ámbito de la ciberseguridad (acciones 1, 2, 3, 4, 5 y 7) en el marco del Mecanismo de Recuperación y Resiliencia, Componente 15, Inversión 7 (MRR C15.I7).

REFERENCIAS

[1] Escapa Gordón, P.; Matellán Olivera,V. y Suárez Corona, A.: "Vehicle-to-Vehicle Secure Communication Protocol Based on Digital Vehicle Identification Number", en *Sensors*, vol. 25, no. 19, pp. 5954, 2025.

[2] González, L.; Vaca, M.; Lattarulo, R.; Calvo, I.; Perez, J.; Ruiz, A. Análisis de riesgos de ciberseguridad en arquitectura de vehículos automatizados. XXXIX Jornadas Automática 2018, 5–7.

[3] OMS. 10 datos sobre la seguridad vial en el mundo. 2017. Available online: https://prevencionar.com/2018/03/15/10-datos-sobre-la-seguridad-vial-en-el-mundo/ (accedido octubre 2025).

[4] OPS/OMS. OMS estima que 7 millones de muertes ocurren cada año debido a la contaminación atmosférica. 2017. Available online: https://www.paho.org/hq/index.php?option=com_content&view=article&id=9406:2014-7-million-deaths-annually-linked-air-pollution&Itemid=135&lang=es (accedido octubre 2025).

[5] Shamir, A: Ïdentity-based cryptosystems and signature schemes", en *Workshop on the Theory and Application of Cryptographic Techniques*, vol. 3, pp. 47-53, 1984.

[6] Fiore, D.y Gennaro, R. : Ïdentity-based key exchange protocols without pairings", en *Transactions on Computational Science* pp. 42–77, 2010.

[7] Shuhaimi, N.I.; Juhana, T. Security: "Vehicular Ad-Hoc Network with Identity-Based Cryptography Approach: A survey", en *Proceedings of the 2012 7th International Conference on Telecommunication Systems, Services and Applications (TSSA)*; pp. 276–279, 2012.

[8] Andreica, T.; Groza, B. "Secure V2V Communication with Identity-based Cryptography from License Plate Recognition", en *Proceedings of the Sixth International Conference on Internet of Things, Management and Security (IOTSMS)*, pp. 366–373, 2019.

[9] Al-shareeda, M.A.; Anbar, M.; Manickam, S.; Hasbullah, I.H.: .An efficient identity-based conditional privacy-preserving authentication scheme for secure communication in a vehicular ad hoc network", en *Symmetry*, vol. 12, pp. 1687, 2020.

[10] Wasef, A.; Shen, X:" Efficient group signature scheme supporting batch verification for securing vehicular networks", en *Proceedings of the 2010 IEEE International Conference on Communications*, pp. 1–5, 2010.

[11] Groza, B.; Andreica, T.; Berdich, A.; Murvay, P.S.; Gurban, E.H: "Prestvo: Privacy enabled smartphone based access to vehicle on-board units", en *IEEE Access* vol. 8, pp. 119105–119122, 2020.

[12] Escapa Gordón, P. ES1249504 Número de bastidor/chasis de un automóvil en formato digital. 2021. Disponible online: https://patentscope.wipo.int/search/en/detail.jsf;jsessionid=EC3B477650F5D25814CA6F00BDC554B2.wapp2nA?docId=ES299105622&_cid=P20-KCVZ3H-27158-3 (accedido en octubre 2025).

[13] CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness. 2012. Disponible online: http://competitions.cr.yp.to/caesar.html (accedido octubre 2025).

Evaluación de riesgos en el procesamiento automatizado de Documentos Nacionales de Identidad

José Carlos Sancho Núñez
Universidad de Extremadura
Centro Universitario de Mérida
Mérida, España
ORCID: 0000-0002-4584-6945
jcsanchon@unex.es

Antonio Calvo Picón
Universidad de Extremadura
Centro Universitario de Mérida
Mérida, España
ORCID: 0009-0008-1815-7686
acalvopic@unex.es

Sergio Guijarro Domínguez
Universidad de Extremadura
Escuela Politécnica de Cáceres
Cáceres, España
ORCID: 0009-0002-9436-8561
sergiosgd@unex.es

Víctor González Morales
Universidad de Extremadura
Escuela Politécnica de Cáceres
Cáceres, España
ORCID: 0009-0009-7504-9929
victorgomo@unex.es

***Resumen*—En este trabajo se propone un marco de evaluación de riesgos para el procesamiento automatizado de documentos de identidad a partir de imágenes capturadas por el propio usuario. El sistema considerado recibe fotografías del anverso y reverso del documento nacional de identidad (DNI), aplica un modelo de detección para localizar y recortar el documento, realiza reconocimiento óptico de caracteres (OCR) y ejecuta un conjunto de validaciones sobre los datos personales extraídos y sobre los elementos de seguridad visibles en el documento. A partir de las tasas de error observadas en cada etapa del *pipeline*, se definen escenarios de riesgo y se cuantifican probabilidades e impactos asociados a la aceptación de documentos falsos, al rechazo de documentos legítimos y a errores sistemáticos del sistema. El resultado es un marco que permite analizar, comparar y diseñar sistemas de verificación de documentos de identidad desde una perspectiva explícita de gestión de riesgos.**

***Index Terms*—Documentos de Identidad, evaluación de riesgos, verificación de identidad, OCR, fraude documental**

I. Introducción

El uso de fotografías de Documentos Nacionales de Identidad (DNI) tomadas con el teléfono móvil se ha generalizado en procesos de verificación remota de identidad, sobre todo en entornos financieros, administrativos y de servicios en línea. Esta tendencia ha impulsado el despliegue de sistemas automáticos que reciben imágenes del anverso y reverso del DNI, detectan y recortan el documento, aplican OCR y ejecutan validaciones sobre los datos extraídos.

Sin embargo, la dependencia de este *pipeline* automatizado introduce riesgos específicos: aceptación de documentos falsos o manipulados, rechazo de documentos legítimos y decisiones basadas en información parcial o errónea.

El objetivo de este trabajo es definir y aplicar un marco de evaluación de riesgos centrado en la cadena de procesamiento formada por captura de imágenes, detección y recorte del documento, OCR y validación de datos personales y elementos de seguridad del DNI. Por tanto, las contribuciones principales del artículo son las siguientes:

- Se define un marco de evaluación de riesgos específico para sistemas de verificación automática de documentos de identidad basados en captura de imágenes, detección, OCR y validación.
- Se describe una arquitectura de referencia para el procesamiento de DNI (anverso y reverso) que integra detección, OCR y validaciones sobre datos personales y elementos de seguridad.
- Se propone una metodología para derivar métricas de riesgo, como las tasas de aceptación de fraude y rechazo de documentos legítimos, a partir de las tasas de error técnicas observadas en cada etapa del sistema.

II. Marco teórico y trabajos relacionados

Los documentos oficiales (como DNI o pasaporte) son ampliamente utilizados para la verificación remota de identidad, especialmente en procesos KYC (*Know Your Customer*). Sin embargo, desde el punto de vista técnico los métodos tradicionales basados en el envío de copias escaneadas presentan riesgos de manipulación, falta de automatización y vulnerabilidades en la validación humana [1].

Por este motivo, son numerosos los trabajos que aplican técnicas de visión por computador para detectar, segmentar y recortar automáticamente documentos en imágenes capturadas desde dispositivos móviles [2]. Es en la etapa de extracción de datos, donde los modelos OCR como Tesseract o soluciones comerciales adaptadas permiten estructurar los campos textuales, aunque presentan limitaciones ante fondos complejos o baja calidad de imagen [**?**].

Por otro lado, la detección de documentos falsificados se apoya en el análisis visual de elementos de seguridad (hologramas, relieves, microtextos) y patrones de diseño característicos. En este sentido, algunos enfoques combinan redes neuronales convolucionales (CNN) con aprendizaje supervisado para distinguir documentos genuinos de alteraciones mediante anomalías gráficas o inconsistencias entre campos [3].

Aunque existen diversos marcos normativos y metodológicos para la gestión de riesgos en procesos de identificación, como [4], [5] o [6], la literatura específica sobre documentos de identidad físicos es limitada, lo que refuerza la relevancia de esta propuesta como herramienta práctica para caracterizar riesgos en entornos reales.

En base a lo referenciado, esta contribución identifica la necesidad de realizar evaluaciones de riesgo en verificación de identidad, asignando un nivel de confianza al proceso en función de parámetros técnicos (calidad de imagen, congruencia de campos, metadatos). Nuestra propuesta sigue la línea de modelos recientes que plantean marcos adaptativos, ajustando la validación según el tipo de documento, el canal de adquisición y la exposición al fraude [7].

III. Modelos de sistema y de amenazas

El sistema propuesto recibe las imágenes del anverso y reverso del DNI previamente capturadas digitalmente por el usuario. Sobre cada imagen se aplica un modelo de detección para localizar y recortar el documento, obteniendo una región de interés estable. A continuación, se ejecuta un proceso de OCR que extrae los campos de texto relevantes y aplica reglas de validación sobre los datos personales y sobre los elementos de seguridad visibles en el documento.

La presentación de documentos completamente falsificados, documentos legítimos alterados parcial o totalmente y usos indebidos de documentos legítimos en contextos para los que no estaban destinados son las principales amenazas y, por tanto, el análisis se estructura en torno a dos dimensiones: la confidencialidad de los datos de identidad tratados y la integridad de las decisiones de verificación. Se emplean métricas tales como la tasa de aceptación de fraude (FAR), la tasa de rechazo de documentos legítimos (FRR) y las tasas de error críticas en la extracción y validación de campos.

IV. Arquitectura técnica: captura, detección, OCR y validaciones

IV-A. Conjunto de datos

Para el entrenamiento y validación del modelo de detección y segmentación, se contruye un conjunto de datos propio compuesto por fotografías de DNI españoles. Todas las imágenes se obtienen con consentimiento explícito de los titulares y se procesan conforme a la normativa vigente (RGPD), garantizando su uso exclusivo para fines de investigación y anonimización de datos sensibles. Este enfoque asegura que el modele se entrene con imágenes que reflejan fielmente las condiciones de uso del sistema en un entorno real.

El dataset consta de un total de 73 imágenes, distribuidas de manera aproximadamente equitativa entre el anverso (41 imágenes) y el reverso (32 imágenes) del documento. Es importante destacar que, para el alcance de esta fase del proyecto, todas las imágenes corresponden a documentos genuinos, ya que el objetivo principal es la robustez en la detección geométrica del documento y no la clasificación de su autenticidad.

Una característica fundamental del conjunto de datos es su alta variabilidad, diseñada deliberadamente para simular un amplio espectro de escenarios de uso real. El conjunto incluye tanto imágenes de alta calidad, con buen enfoque, iluminación adecuada y perspectiva frontal, como un subconjunto de casos de desafío para entrenar la robustez del modelo. Estas últimas imágenes incorporan, de forma individual o combinada, errores comunes como desenfoque de movimiento (motion blur), rotaciones pronunciadas, distorsión de perspectiva y condiciones de iluminación deficientes.

El conjunto de datos se divide en dos subconjuntos siguiendo una proporción estándar: 60 imágenes (aproximadamente el 82 %) se destinan para el **entrenamiento** del modelo, y las 13 imágenes restantes (el 18 %) se utilizan para su **validación**. Finalmente, cada imagen en el conjunto de datos se anota manualmente, lo que genera máscaras de segmentación precisas que delinean el contorno exacto del DNI. Estas máscaras constituyen la verdad terreno (*ground truth*) que el modelo YOLO utiliza para aprender a localizar el documento.

IV-B. Captura de imágenes

La etapa de captura de imágenes es un pilar fundamental del sistema, ya que la calidad de la entrada impacta directamente en el rendimiento de los módulos de detección y OCR. Para mitigar los errores derivados de una captura deficiente, se definen requisitos técnicos clave: una resolución adecuada (e.g., Full HD) para asegurar la legibilidad de todos los campos; una iluminación uniforme que evite reflejos y sombras; un encuadre completo donde el documento sea totalmente visible; y un enfoque nítido para prevenir el desenfoque de movimiento (*motion blur*). Para facilitar al usuario el cumplimiento de estos requisitos, se diseña un flujo de interacción guiado. La interfaz muestra una silueta del documento para facilitar el encuadre y solicita de forma secuencial la captura del anverso y, posteriormente, la del reverso. Este proceso maximiza la calidad de la imagen de entrada, lo que reduce significativamente la probabilidad de errores en las etapas posteriores.

IV-C. Detección y recorte del documento

El primer paso del pipeline de procesamiento consiste en localizar de forma precisa el DNI dentro de la imagen de entrada. El objetivo de esta etapa es aislar el documento del fondo, independientemente de su posición, escala o de la distorsión de perspectiva, para normalizar la entrada de las fases posteriores.

Para esta tarea, se emplea un modelo de la familia YOLO (You Only Look Once), concretamente la versión `yolov8n-seg.pt` [8]. La elección de YOLOv8 se fundamenta en su arquitectura de última generación (*state-of-the-art*), que ofrece un excelente equilibrio entre velocidad y precisión. Se opta por la variante ‘n‘ (nano) por ser un modelo extremadamente ligero, rápido y óptimo.

La decisión de utilizar un modelo de segmentación de instancias (`-seg`) en lugar de uno de detección de objetos convencional es un requisito crítico para el sistema. Un detector estándar únicamente proporciona una caja delimitadora rectangular (*bounding box*), siendo insuficiente para procesar documentos capturados en ángulo. En cambio, el modelo de segmentación genera una máscara poligonal que se ajusta con alta fidelidad al contorno exacto del DNI. Esta máscara permite la extracción fiable de las coordenadas de las cuatro esquinas del documento, incluso bajo una fuerte distorsión de perspectiva. Estos cuatro puntos son la entrada imprescindible para la etapa de recorte, donde se aplica una transformación de perspectiva para rectificar la imagen.

Proceso de Entrenamiento: El modelo se entrena siguiendo una estrategia de transferencia de conocimiento (*transfer learning*) [9]. Se parte de los pesos del modelo `yolov8n-seg.pt` pre-entrenado sobre el extenso dataset COCO [10], y se realiza un ajuste fino (*fine-tuning*) utilizando el conjunto de datos de DNI descrito en la sección anterior. Este enfoque permite aprovechar el conocimiento general de características visuales aprendido previamente, lo que acelera la convergencia y mejora el rendimiento final.

El entrenamiento se lleva a cabo en una estación de trabajo equipada con una GPU NVIDIA GeForce RTX 4070 de 12 GB (PyTorch 2.7.1, CUDA 11.8). Los principales hiperparámetros incluyen un tamaño de imagen (imgsz) de

640x640 píxeles para un buen compromiso entre detalle y eficiencia, un tamaño de lote (batch) de 16 para optimizar el uso de la VRAM, y un total de 60 épocas (epochs) para asegurar el correcto ajuste de los pesos.

Para mejorar la generalización del modelo y su robustez ante la variabilidad de las imágenes de entrada, se utilizan las técnicas de aumento de datos (*data augmentation*) que incorpora por defecto el framework de Ultralytics. Estas incluyen transformaciones geométricas (rotaciones, escalado, traslaciones) y fotométricas (cambios de brillo, contraste y saturación). Adicionalmente, se implementa un mecanismo de parada temprana (*early stopping*) con una paciencia de 50 épocas. Este mecanismo monitoriza el rendimiento del modelo en el conjunto de validación y detiene el entrenamiento si no se observa una mejora, evitando así el sobreajuste (*overfitting*).

Corrección de Perspectiva y Recorte: Una vez obtenida la máscara de segmentación del DNI, se aplica un algoritmo de procesamiento geométrico para generar un recorte rectificado del documento. El proceso, implementado con la librería OpenCV, sigue los siguientes pasos:

1. **Extracción del Contorno:** A partir de la máscara binaria generada por el modelo, se extraen los contornos mediante el algoritmo de Suzuki [11]. Se selecciona el contorno de mayor área para asegurar que se procese el objeto principal detectado, descartando posibles artefactos o detecciones erróneas de menor tamaño.
2. **Aproximación de los Vértices:** El contorno extraído se aproxima a un polígono utilizando el algoritmo de Ramer-Douglas-Peucker [12] (`cv2.approxPolyDP`). El objetivo es encontrar los cuatro vértices que definen las esquinas del documento. Se implementa un mecanismo de contingencia: si el algoritmo no logra identificar un cuadrilátero, el sistema recurre a un recorte rectangular simple basado en el *bounding box* del contorno.
3. **Ordenamiento de Puntos:** Los cuatro vértices identificados se ordenan en una secuencia canónica y predecible: superior-izquierda, superior-derecha, inferior-derecha e inferior-izquierda. Este paso es indispensable para calcular correctamente la matriz de transformación.
4. **Cálculo y Aplicación de la Homografía:** Se calcula la matriz de transformación de perspectiva (homografía) de 3x3 que mapea los cuatro vértices del DNI en la imagen original a las esquinas de una nueva imagen de destino rectangular y plana. Finalmente, se aplica esta transformación a la imagen original mediante la función `cv2.warpPerspective` [13], lo que da como resultado una imagen del DNI con vista frontal, libre de distorsión y fondo.

El resultado de esta etapa es una imagen normalizada de alta calidad que sirve de entrada para el módulo de OCR, lo que maximiza la probabilidad de una extracción exitosa del texto.

IV-D. OCR y extracción de campos

Tras el recorte del documento en la fase de detección, se aplica un proceso de OCR sobre las regiones correspondientes a los campos del DNI, tanto en el anverso como en el reverso. Para ello se emplea la biblioteca `tesseract.js`, una adaptación de Tesseract OCR para su ejecución en el navegador, que permite mantener todo el procesamiento en el lado cliente sin exponer datos personales a servidores externos.

El procedimiento parte de un conjunto de coordenadas normalizadas (expresadas en proporciones relativas a la imagen) que definen la posición de cada campo en ambas caras del documento. Estas posiciones han sido previamente definidas manualmente a partir del diseño del DNI 4.0, incluyendo variaciones para algunos elementos (como el número de soporte o el número de DNI, que pueden aparecer en múltiples ubicaciones). A partir de estas coordenadas, se extraen las regiones correspondientes mediante canvas HTML5, sobre las que se ejecuta el reconocimiento con Tesseract de forma individual por campo. Esta estrategia permite mejorar la precisión respecto a un OCR global, reduciendo ruido entre campos adyacentes.

La Tabla I muestra un ejemplo de los campos extraídos a partir de un DNI sintético (Figura 1). Esta representación no contiene datos reales y cumple con la legislación vigente de protección de datos personales. Los campos extraídos incluyen, entre otros: nombre, apellidos, número de DNI, fecha de nacimiento, fecha de emisión y caducidad, número de soporte, nacionalidad, sexo y el código CAN en el anverso, así como la dirección, municipio, progenitores, provincia, equipo expedidor y la zona de lectura mecánica (MRZ) en el reverso.



(a) Anverso del DNI sintético. (b) Reverso del DNI sintético.

Figura 1: Ejemplo completo de DNI sintético utilizado para validar el sistema de extracción OCR. Las imágenes no representan ningún documento real de identidad.

Tabla I: Ejemplo de campos extraídos a partir de un DNI sintético (Fig. 1).

Campo	Valor extraído
Campos del anverso (Fig. 1a)	
Nombre	CARMEN
Apellidos	ESPAÑOLA ESPAÑOLA
DNI	99999999R
Sexo	F
Nacionalidad	ESP
Fecha de nacimiento	01/01/1980
Fecha de expedición	02/06/2021
Fecha de caducidad	02/06/2031
Número de soporte	CAA000000
Código CAN	020631
Firma	(detectada pero no extraída)
Campos del reverso (Fig. 1b)	
Domicilio	AVDA DE MADRID S-N, MADRID
Municipio	MADRID
Provincia	MADRID
Equipo expedidor	0000
Progenitores	JUAN / CARMEN
MRZ	IDESPCAA0000004... (truncado)

El resultado de esta etapa es un objeto estructurado con los campos extraídos, que será posteriormente sometido a validaciones de formato, coherencia y consistencia inter-cara.

IV-E. Validación de datos personales

Tras la extracción OCR, los campos son normalizados y validados mediante reglas de formato específicas y comprobaciones cruzadas entre anverso y reverso. La validación abarca aspectos estructurales, semánticos y de coherencia interna.

Normalización y estructura: Se aplican expresiones regulares y reglas heurísticas para corregir errores típicos de OCR (O → 0, B → 8, etc.). Campos como el DNI (99999999R) o el número de soporte se ajustan a sus patrones esperados y se validan mediante cálculos de control como el módulo 23 (DNI) o el checksum MRZ (soporte). La Tabla II muestra ejemplos de correcciones aplicadas.

Tabla II: Correcciones aplicadas durante la normalización del número de soporte.

Sección	OCR Erróneo	Corrección
Prefijo	Dígitos	Reemplazados por letras [A–Z]
Sufijo	O, D, N, Q, B, S, I	0, 0, 0, 0, 8, 5, 1
Otros	Símbolos	Eliminados

Validación de fechas y consistencia: Las fechas de nacimiento, expedición y caducidad se aceptan con distintos formatos y se validan por coherencia lógica: edad plausible (1–115 años), no fechas futuras, caducidad vigente. El sistema verifica que el DNI y el soporte coincidan en anverso y reverso, y que los checksums de la MRZ sean correctos.

Resumen de reglas por campo: La Tabla III resume los patrones de validación aplicados a cada campo. Los valores que no superan la validación se marcan como null. La firma no es evaluada.

Tabla III: Resumen de patrones de validación por campo.

Campo	Regla o patrón
DNI	[0–9]{8}[A–Z] + control
Nombre / Apellidos	Mayúsculas sin símbolos, 2 bloques
Sexo	[MF]
Nacionalidad	ISO 3166 (ESP)
Fechas	DD/MM/AAAA, coherencia temporal
N.º Soporte	[A–Z0–9]{9} + checksum MRZ
CAN	[0–9]{6}
Equipo	[0–9]{4}
MRZ	3 líneas × 30, checksums y consistencia
Firma	– (no evaluada)

IV-F. Validación de elementos de seguridad

Además del contenido textual, el sistema contempla una validación preliminar de ciertos elementos estructurales y visuales del documento, con especial atención a la zona MRZ, sin realizar aún análisis profundo de patrones gráficos.

Zonas y criterios observados: Se consideran indirectamente algunas zonas características del documento:

- **Zona MRZ:** Validación de formato estructural (3 líneas, 30 caracteres).
- **Disposición general:** Comprobación de orientación, márgenes y alineación básica.
- **Relación geométrica de campos:** Se verifica consistencia visual con plantillas oficiales.

Validaciones aplicadas: El sistema aplica las siguientes comprobaciones:

- **Estructura y checksums MRZ:** Se verifica la codificación válida de caracteres y el correcto cálculo de dígitos de control en campos clave.
- **Homografía básica:** Se compara la geometría del documento con una plantilla canónica para detectar recortes anómalos o rotaciones, mejorando también la entrada para el OCR, siguiendo enfoques similares a los descritos en [14].

Limitaciones: Aún no se emplean modelos específicos para detección de elementos gráficos de seguridad (e.g., hologramas o tintas especiales), debido a:

- Variabilidad en la calidad de imagen (desenfoque, brillo, etc.).
- Restricciones del modelo empleado para la detección actual.
- Carencia de datasets para el entrenamiento especializado.

V. Metodología de evaluación de riesgos

La agrupación en tres escenarios se deriva del análisis de fallos observados durante el entrenamiento y validaión del sistema en cada fase del flujo de trabajo. A partir de las cinco fases principales (captura, detección y recorte, OCR y validación de datos del documento), se identifican errores recurrentes y se clasifican según impacto operativo y su relevancia con la integridad del proceso.

- **Aceptación de un documento falsificado (falso positivo):** Se produce cuando un documento manipulado supera todas las verificaciones. Suele deberse a fallos en la lectura de la MRZ, errores en checksums (e.g., número de soporte) o detecciones incorrectas que permiten procesar imágenes incompletas.
- **Rechazo de un documento legítimo (falso negativo):** Ocurre cuando un documento auténtico es descartado por errores de lectura o normalización. Las principales causas son OCR impreciso en campos críticos (fechas, DNI), artefactos que impiden validar la MRZ o normalizaciones que generan valores null.
- **Errores sistemáticos de extracción o validación:** Aparecen bajo condiciones de captura adversas (glare, sombras, baja resolución), en campos con tildes o estructuras particulares, o en validaciones específicas como el número de soporte. Estos fallos repetidos comprometen la fiabilidad operativa del sistema.

V-0a. Caracterización por fase: Cada escenario se relaciona con riesgos específicos en las cinco fases:

- **Captura:** Riesgo de imágenes con baja calidad, glare o rotación que afectan la detección.
- **Detección y recorte:** Riesgo de recortes incompletos o detecciones erróneas que permiten procesar documentos falsos.
- **OCR:** Riesgo de errores sistemáticos en caracteres críticos (confusión 0/O, 8/B) que impactan en las validaciones.
- **Validación de datos personales:** Riesgo de rechazo de documentos legítimos por normalizaciones incorrectas.
- **Validación de elementos de seguridad:** Riesgo de aceptar documentos falsificados si fallan las comprobaciones de MRZ o checksums.

Estos escenarios recogen las amenazas reales observadas durante el desarrollo y representan los tipos de fallo que el sistema puede producir en función de las tasas de detección fallida, OCR erróneo y validaciones incorrectas.

V-A. *Estimación de probabilidad e impacto*

La probabilidad de cada escenario se calcula a partir de las tasas de error obtenidas en cada módulo:

- E_d: fallos en detección y recorte (documento no detectado o parcialmente recortado).
- E_o: errores de OCR y normalización (fechas inválidas, confusiones `0/O`, `8/B`, campos `null`).
- E_v: fallos en validación (checksum MRZ, DNI inconsistente, número de soporte no válido).

El impacto se clasifica cualitativamente según las consecuencias operativas:

- **Bajo (B):** Errores puntuales y recuperables.
- **Moderado (M):** Reprocesos o afectación a decisiones de negocio.
- **Crítico (C):** Permite fraude o bloquea usuarios legítimos.

V-B. *Índice global de riesgo*

La probabilidad del escenario s se estima:

$$P_s = w_d \cdot E_d + w_o \cdot E_o + w_v \cdot E_v \tag{1}$$

donde los pesos w_d, w_o, w_v reflejan la relevancia relativa de cada módulo y cumplen:

$$w_d + w_o + w_v = 1.$$

Clasificamos el valor resultante como:

- $P_s > 0{,}66 \rightarrow A$ (Alta)
- $0{,}33 < P_s \leq 0{,}66 \rightarrow M$ (Media)
- $P_s \leq 0{,}33 \rightarrow B$ (Baja)

Obtenemos el índice final combinando probabilidad e impacto mediante la matriz de la Tabla IV.

Tabla IV: Matriz de evaluación de riesgos: combinación de probabilidad e impacto.

Impacto \ Probabilidad	Alta	Media	Baja
Crítico	Alto	Medio	Medio
Moderado	Medio	Medio	Bajo
Bajo	Medio	Bajo	Bajo

VI. Resultados y discusión

VI-A. *Evaluación del modelo de detección y recorte*

La evaluación del modelo de segmentación `yolov8n-seg.pt` se realiza con el objetivo de medir su capacidad para localizar con precisión el DNI y generar un recorte normalizado que sirva como entrada de alta calidad para el módulo OCR. Se analizan tanto métricas cuantitativas como resultados cualitativos.

VI-A1. Métricas del modelo: El rendimiento del modelo se evalúa sobre el conjunto de validación de 13 imágenes, que incluye tanto capturas de alta calidad como casos de desafío (imágenes con alta distorsión de perspectiva, rotación y desenfoque). Los resultados obtenidos para la tarea de segmentación se resumen en la Tabla V.

Tabla V: Métricas de rendimiento del modelo YOLOv8n-seg en el conjunto de validación.

Métrica	Valor
Precisión (Precision)	1.000
Exhaustividad (Recall)	0.998
mAP50	0.995
mAP50-95	0.933

A continuación, se interpretan las métricas obtenidas:

- **Precisión (Precision):** Mide la fiabilidad de las detecciones. Un valor de **1.000** indica que todas las instancias clasificadas como DNI corresponden efectivamente a documentos reales, lo que implica una ausencia total de falsos positivos.
- **Exhaustividad (Recall):** Evalúa la capacidad del modelo para encontrar todos los DNIs existentes en las imágenes. Un valor de **0.998** evidencia que el sistema es capaz de detectar casi la totalidad de los documentos presentes en el conjunto de validación.
- **mAP50:** Es la métrica principal para esta tarea, ya que evalúa la calidad de la máscara de segmentación. Considera una detección como correcta si el solapamiento (IoU) entre la máscara predicha y la real es de al menos el 50 %. El resultado de **0.995** demuestra una capacidad sobresaliente del modelo para generar máscaras que se ajustan con gran fidelidad a la forma del documento.
- **mAP50-95:** Es una métrica más estricta que promedia el mAP en múltiples umbrales de IoU (de 0.5 a 0.95). El alto valor de **0.933** confirma que las máscaras generadas son correctas y precisas en su contorno, incluso bajo los criterios de evaluación más exigentes.

En conjunto, estos valores refuerzan la conclusión de que el modelo es extremadamente fiable tanto para evitar falsas detecciones como para generar una base geométrica precisa para las etapas posteriores de procesamiento.

Para complementar estos datos cuantitativos, se presenta un resultado cualitativo en la Figura 2. La imagen muestra la salida del modelo YOLOv8n-seg sobre una captura de entrada, donde se puede observar la caja delimitadora (*bounding box*), la puntuación de confianza (0.93) y, de forma superpuesta, la máscara de segmentación poligonal que se adhiere con precisión al contorno del documento. Esta máscara es el resultado fundamental de esta etapa, ya que proporciona las coordenadas exactas de las esquinas del DNI, que son la entrada necesaria para el subsiguiente algoritmo de corrección de perspectiva. Una vez rectificada la imagen, esta se pasa al módulo de OCR para la extracción de texto.



Figura 2: Salida del modelo de detección y segmentación YOLOv8n-seg. Se muestra la caja delimitadora, la confianza de la detección (0.93) y la máscara de segmentación (en rojo) sobre la imagen original.

VI-B. Evaluación del módulo OCR y validación

El sistema OCR se basa en `tesseract.js 5.1.0` con modelo `spa.traineddata`. La evaluación del rendimiento se lleva a cabo mediante un conjunto de imágenes sintéticas anotadas, usando validaciones por campo con expresiones regulares.

Los resultados muestran alta precisión en campos estructurados como DNI, fechas, nacionalidad, número de soporte o código CAN. Por el contrario, presentan mayor tasa de error campos sensibles al contexto o al diseño gráfico del fondo, como nombre, apellidos o sexo. Las principales fuentes de error son:

- **Ruido visual**: Zonas gráficas con escudos, líneas o fondos coloridos afectan la segmentación del texto.
- **Coordenadas fijas**: El uso de recortes estáticos reduce la robustez ante variaciones en tamaño o alineación.

Precisión OCR estimada por campo: La Tabla VI resume la precisión cualitativa observada por campo en el entorno de pruebas.

Tabla VI: Tasa estimada de acierto del OCR por campo (documentos sintéticos).

Campo	Precisión estimada
DNI, MRZ	$> 98\,\%$
Fechas (nac., exp., cad.)	95–98 %
Nacionalidad, CAN, soporte	$> 95\,\%$
Nombre simple, apellidos claros	85–90 %
Nombre/apellidos compuestos	65–80 %
Sexo, domicilio, municipio	80–90 %
Equipo expedidor, progenitores	$> 90\,\%$
Firma	No evaluada

VI-C. Discusión

Los resultados obtenidos confirman la solidez del *pipeline* para el procesamiento automatizado del DNI. El módulo de detección y recorte presenta un rendimiento sobresaliente (**mAP50 de 0.995**), generando máscaras precisas incluso en imágenes con distorsión o baja calidad. Esta fiabilidad es clave para el OCR, ya que la corrección de perspectiva y la normalización geométrica minimizan los errores derivados de recortes estáticos.

Respecto al OCR, los campos estructurados y con patrones regulares (DNI, fechas, MRZ, número de soporte) alcanzan tasas de acierto superiores al 95 %. En contraste, los campos más sensibles al contexto —como nombres y apellidos compuestos— muestran mayor variabilidad debido a la complejidad visual del fondo y a diferencias tipográficas. En conjunto, los resultados evidencian un rendimiento muy alto en la extracción de información crítica y dejan identificados los principales retos en el tratamiento de texto libre.

VII. Conclusiones y trabajos futuros

En este artículo se ha presentado un marco de evaluación de riesgos para sistemas de verificación automática de documentos de identidad basados en captura de imágenes, detección, OCR y validaciones. La propuesta permite vincular las tasas de error técnicas del sistema con indicadores de riesgo tales como la aceptación de fraude y el rechazo de documentos legítimos, facilitando el diseño y la comparación de soluciones alternativas.

Como líneas de trabajo futuras, se plantea: (1) la mejora de los modelos base, mediante la ampliación del conjunto de datos de detección con imágenes reales y sintéticas, la evaluación de arquitecturas YOLO alternativas y el entrenamiento específico del modelo OCR sobre documentos de identidad; (2) la obtención dinámica de coordenadas para cada campo mediante detección automática de regiones de interés, eliminando la dependencia de posiciones fijas; (3) la incorporación de validaciones de elementos de seguridad visuales, como la detección de hologramas; y (4) la extensión del sistema para que sea capaz de procesar otros tipos de documentos de identidad, como pasaportes o permisos de conducir.

Agradecimientos

Esta iniciativa se realiza en el marco de los fondos del Plan de Recuperación, Transformación y Resiliencia, financiados por la Unión Europea (Next Generation) - Instituto Nacional de Ciberseguridad (INCIBE) en el Proyecto C108/23 “Detección de Falsificación de Documentos de Identidad mediante Técnicas de Visión por Computador e Inteligencia Artificial”.

Referencias

[1] R. Patel, A. Sharma, and M. Khan, “Digital identity verification: Transforming kyc with ai and emerging technologies,” *International Research Journal of Modernization in Engineering, Technology and Science*, vol. 5, no. 9, pp. 3104–3112, 2023. [Online]. Available: https://www.irjmets.com/uploadedfiles/paper/issue_9_september_2023/44476/final/fin_irjmets1693845854.pdf

[2] D. Karatzas *et al.*, “Real-time document detection and rectification for mobile platforms,” in *Proc. Int. Workshop on Human-Document Interaction*, 2016, descripción en Karatzas et al., Human-Document Interaction Systems. [Online]. Available: https://refbase.cvc.uab.es/files/KPR2016.pdf

[3] T. Aljuaid, M. Alawairdhi, and R. Alotaibi, “Id card spoofing detection using frequency features and cnns,” *Ingénierie des Systèmes d’Information*, vol. 30, no. 8, pp. 1–15, 2025. [Online]. Available: https://www.iieta.org/journals/isi/paper/10.18280/isi.300813

[4] *ISO/IEC 29100:2024 – Information technology — Security techniques — Privacy framework*, ISO/IEC Std., 2024, https://www.iso.org/standard/85938.html.

[5] *ISO/IEC 27005:2022 – Information security, cybersecurity and privacy protection — Guidance on managing information security risks*, ISO/IEC Std., 2022, https://www.iso.org/standard/80585.html.

[6] NIST SP 800-37 Revision 2, “Risk management framework (rmf) for information systems & organizations,” National Institute of Standards and Technology (NIST), Tech. Rep., 2019, https://doi.org/10.6028/NIST.SP.800-37r2.

[7] L. S. Dasu, M. Dhamija, G. Dishitha, A. Vivekanandan, and V. Sarasvathi, “Defending against identity threats using risk-based authentication,” *Cybernetics and Information Technologies*, vol. 23, no. 2, pp. 105–123, 2023. [Online]. Available: https://doi.org/10.2478/cait-2023-0016

[8] G. Jocher, J. Qiu, and A. Chaurasia, “Ultralytics YOLO,” https://github.com/ultralytics/ultralytics, 2023, versión 8.0.0, Licencia AGPL-3.0.

[9] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, “A survey on deep transfer learning,” in *Proc. Int. Conf. Artif. Neural Netw. (ICANN)*, 2018, pp. 270–279.

[10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 740–755.

[11] S. Suzuki and K. Abe, “Topological structural analysis of binary images by border following,” *Comput. Vis. Graph. Image Process.*, vol. 30, no. 1, pp. 32–46, 1985.

[12] D. H. Douglas and T. K. Peucker, “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,” *Cartographica*, vol. 10, no. 2, pp. 112–122, 1973.

[13] G. Bradski, *The Opencv Library*, 11 2000, vol. 25.

[14] Z. Li, J. Zhang, S. Gu, L. Van Gool, and R. Timofte, “Document rectification and illumination correction using a patch-based cnn,” in *Proc. ACM Int. Conf. Multimedia*, 2019, pp. 1643–1651. [Online]. Available: https://dl.acm.org/doi/10.1145/3355089.3356563

Evaluación comparativa de modelos de aprendizaje automático utilizados para la detección de anomalías en registros de Active Directory

Allysson de Oliveira Noronha*, Robson de Oliveira Albuquerque*†
* Professional Post-Graduate Program in Electrical Engineering (PPEE)
Department of Electrical Engineering (ENE), Faculty of Technology,
University of Brasilia (UnB), Brasília-DF, zipcode 70910-900, Brazil
Email: allysson.noronha@aluno.unb.br, robson@redes.unb.br
† Postgraduate in Governance, Technology, and Innovation,
Catholic University of Brasília (UCB),
Brasília-DF, zipcode 71966-700, Brazil

***Resumen*—El presente artículo analiza de forma comparativa diversos modelos de aprendizaje automático (*Machine Learning*) y aprendizaje profundo (*Deep Learning*) orientados a la detección de anomalías en patrones de acceso en entornos Active Directory. Se detalla un flujo de trabajo integral que comprende la generación de datos, el preprocesamiento, la ingeniería de características y una evaluación homogénea de distintos enfoques bajo un entorno controlado. Los hallazgos derivados de la experimentación ponen de manifiesto que las arquitecturas basadas en redes neuronales —específicamente el Perceptrón Multicapa (MLP)— logran un rendimiento superior en términos de *F1-score* y sensibilidad. Finalmente, se examinan las principales limitaciones del estudio y los retos inherentes a la adopción de estos modelos en contextos operacionales reales.**

***Index Terms*—Anomaly detection, Active Directory, Machine Learning, UEBA, Deep Learning**

I. Introducción

La creciente sofisticación de las ciberamenazas y la proliferación de *malware* representan riesgos significativos para individuos y organizaciones, lo que puede derivar en violaciones de datos, fallos del sistema y pérdidas financieras de gran calado [1].

Ante estas amenazas, resulta viable identificar de forma proactiva intrusiones o accesos indebidos mediante el examen automatizado de los registros de actividad [2].

Dichas técnicas de detección de anomalías, fundamentadas en el aprendizaje automático, logran identificar patrones de acceso atípicos al prescindir del modelado manual de escenarios de ataque .

Tradicionalmente, la defensa contra el *malware* se ha basado en mecanismos de firmas, eficaces para amenazas previamente conocidas, pero limitados frente a entornos dinámicos y ataques que explotan el factor humano o el movimiento lateral dentro de la red corporativa [4].

En este sentido, el análisis sistemático de los registros generados por los controladores de dominio en entornos Active Directory (AD) adquiere una relevancia crítica. Al ser una infraestructura fundamental para la gestión de identidades y el control de accesos, su monitoreo permite observar directamente el comportamiento de autenticación de los usuarios e identificar indicadores tempranos de compromiso, como variaciones atípicas en horarios de sesión o cambios inesperados en la pertenencia a grupos privilegiados [5].

El estado del arte en la detección de anomalías en registros de sistemas ha evolucionado desde enfoques estadísticos tradicionales hacia métodos avanzados de aprendizaje automático y aprendizaje profundo. No obstante, existen pocos estudios que comparen de forma sistemática estos enfoques en el contexto específico de Active Directory y del Análisis de Comportamiento de Usuarios y Entidades (UEBA).

Para cubrir este vacío, el presente estudio realiza una evaluación comparativa de diferentes modelos de aprendizaje automático y aprendizaje profundo bajo un escenario controlado, con el objetivo de analizar su capacidad para detectar anomalías en patrones de inicio de sesión.

Se establece como meta pragmática alcanzar una exactitud superior al 75 %, umbral comúnmente considerado aceptable en escenarios de datos altamente desbalanceados.

Tras realizar una comparativa con trabajos previos centrados en modelos específicos o arquitecturas particulares, tales como LogAnomaly [6] y DeepLog [7], este estudio propone un enfoque comparativo que evalúa simultáneamente modelos clásicos y de aprendizaje profundo bajo un mismo escenario experimental, pero con un foco específico en registros de Active Directory (AD).

Los hallazgos de este trabajo demuestran que los modelos de aprendizaje profundo poseen una capacidad superior para capturar relaciones multivariadas complejas. Frente a propuestas de alta complejidad como LogGD [8], este artículo prioriza la interpretabilidad y la viabilidad técnica. El objetivo es ofrecer un flujo de trabajo reproducible que sea directamente aplicable a entornos corporativos reales basados en AD.

Más allá de la mera comparativa técnica, las aportaciones de este trabajo residen, en primer lugar, en la propuesta de un flujo de trabajo estructurado y adaptado específicamente a entornos AD. Asimismo, el análisis trasciende el plano teórico al confrontar distintos modelos con datos de tráfico real, lo que proporciona una base sólida para la mitigación de intrusiones en infraestructuras críticas.

Ante las evidencias empíricas obtenidas, se concluye que los modelos de aprendizaje profundo, con especial énfasis en el Perceptrón Multicapa (MLP), presentan un rendimiento más robusto frente a las metodologías clásicas al identificar ano-

malías en registros de autenticación. Esta superioridad técnica valida la viabilidad de integrar arquitecturas neuronales en el monitoreo proactivo de accesos indebidos.

El manuscrito se encuentra articulado de la siguiente manera: tras esta introducción, la Sección II contextualiza los trabajos relacionados; la Sección III detalla la metodología propuesta; finalmente, las Secciones IV y V exponen los resultados y las conclusiones, respectivamente.

II. Fundamentación Teórica y Trabajos Relacionados

Esta sección expone el sustento teórico de la presente investigación, el cual abarca desde la naturaleza de los registros de sistema hasta la aplicación de paradigmas de aprendizaje automático y profundo. El análisis aquí expuesto detalla cómo estas tecnologías permiten optimizar la detección de anomalías en entornos de Active Directory.

II-A. Registros de sistema y preprocesamiento de datos

Los registros (o *logs*) constituyen secuencias cronológicas de eventos generadas por entornos computacionales con el fin de documentar estados, actividades y transiciones internas. Para su análisis mediante técnicas de aprendizaje automático y profundo, estos datos deben someterse a una fase de preprocesamiento que facilite su transformación en representaciones numéricas adecuadas .

Dicha etapa suele integrar la agregación temporal de eventos, la extracción de atributos significativos y la codificación de variables categóricas, lo que permite la construcción de vectores de características que capturen el comportamiento de los usuarios o entidades a lo largo del tiempo.

II-B. Detección de anomalías en registros

Las anomalías se definen como instancias o patrones que manifiestan un comportamiento infrecuente o inesperado en relación con el patrón dominante de los datos. En el ámbito de los registros de sistemas, la literatura identifica habitualmente tres tipologías fundamentales de anomalías [3]:

1. **Anomalías puntuales:** instancias individuales que presentan una desviación significativa respecto al comportamiento normal.
2. **Anomalías contextuales:** instancias que solo son consideradas anómalas bajo un determinado contexto, como una franja horaria o ubicación específica.
3. **Anomalías colectivas:** conjuntos de eventos cuya ocurrencia conjunta o secuencial resulta anómala, a pesar de que cada evento individual, analizado de forma aislada, pueda parecer legítimo.

Gran parte de los enfoques orientados a la detección de anomalías parten de la premisa de un acentuado desequilibrio entre clases, en el cual las instancias normales presentan una clara preponderancia sobre las anómalas.

Esta característica propicia el uso de técnicas no supervisadas o semisupervisadas. En estos esquemas, los modelos se entrenan exclusivamente con datos que representan el comportamiento normal, de modo que la detección se ejecuta sobre datos no observados previamente que podrían contener anomalías.

II-C. Aprendizaje automático para la detección de anomalías

El aprendizaje automático engloba una vasta gama de metodologías orientadas a la identificación de irregularidades en los registros de sistemas. Recientemente, el foco de la investigación se ha desplazado hacia el despliegue de arquitecturas basadas en redes neuronales profundas, dada su capacidad intrínseca para capturar las dependencias complejas y las relaciones no lineales que caracterizan a este tipo de datos [3].

Dentro del espectro de las arquitecturas de aprendizaje profundo, sobresalen las redes neuronales recurrentes (RNN) por su idoneidad en el modelado de secuencias temporales, así como los modelos basados en autoencoders; estos últimos facilitan el aprendizaje de representaciones compactas del comportamiento normal para detectar desviaciones a partir del error de reconstrucción [9].

Sin embargo, esta capacidad analítica conlleva desafíos intrínsecos en términos de interpretabilidad, exigencia computacional y una marcada dependencia de vastos volúmenes de datos para el entrenamiento optimizado de los modelos.

Con el propósito de contrastar dichas capacidades en un entorno controlado, el presente estudio somete a una evaluación comparativa diversos modelos de amplia trayectoria en la literatura especializada: One-Class Support Vector Machine (OCSVM), Local Outlier Factor (LOF), Isolation Forest (IF), Autoencoders (AE) y Perceptrón Multicapa (MLP).

En primer lugar, el OCSVM (One-Class Support Vector Machine) destaca por su capacidad para delimitar una frontera hiperplana que encapsula la mayoría de los datos considerados normales en un espacio de características de alta dimensión. Tras un entrenamiento basado exclusivamente en instancias legítimas, cualquier observación que se sitúe fuera de este margen se clasifica como una anomalía [10].

Por su parte, el método Local Outlier Factor (LOF) se fundamenta en el análisis de la densidad local, comparando la dispersión de cada instancia respecto a sus vecinos más cercanos. En este esquema, se identifica un punto como anómalo cuando su densidad local es significativamente inferior a la de su vecindad inmediata, lo que revela su carácter divergente [11].

En lo que respecta a los autoencoders, estas redes neuronales operan mediante el aprendizaje de representaciones comprimidas de los patrones subyacentes. Su eficacia reside en la capacidad de reconstruir datos normales con alta fidelidad; por consiguiente, ante una anomalía, el modelo genera un error de reconstrucción elevado que facilita su detección [12].

Asimismo, el Isolation Forest adopta un enfoque basado en árboles de decisión aleatorios, cuya premisa es el aislamiento explícito de las observaciones. Dado que las instancias atípicas son más susceptibles de ser segregadas, estas requieren un menor número de particiones, lo que se traduce en rutas de decisión sensiblemente más cortas dentro del bosque [13].

Finalmente, la arquitectura del Perceptrón Multicapa (MLP) se emplea para capturar representaciones no lineales complejas del comportamiento estándar. Ante desviaciones sustanciales respecto a dichos patrones, el incremento en el error de predicción o reconstrucción posibilita su implementación

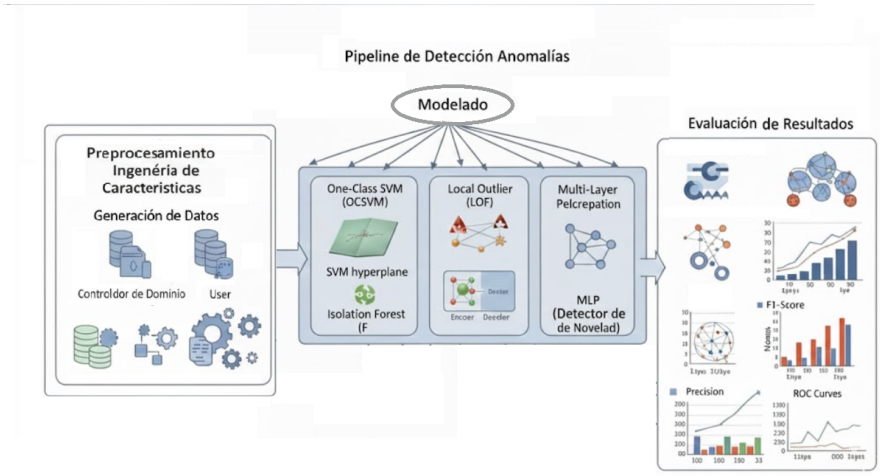


Figura 1. Flujo de trabajo estructurado para la detección de anomalías

como un detector de novedad altamente sensible [14].

II-D. Trabajos Relacionados

La investigación en este campo cuenta con una base documental consolidada. He *et al*. [15] ofrecen un análisis exhaustivo de estas técnicas basadas en logs, abarcando desde los métodos clásicos citados anteriormente hasta los paradigmas incipientes de aprendizaje profundo. En esta línea, propuestas como LogAnomaly [6] y DeepLog [7] resultaron determinantes al estandarizar el uso de autoencoders y modelos secuenciales para la identificación de eventos imprevistos en sistemas complejos.

La evolución constante de este campo de estudio ha propiciado el desarrollo de arquitecturas de creciente complejidad; un ejemplo notable es LogGD [8], que explora el potencial de los grafos para la representación de eventos.

Al analizar la intersección de UEBA (*User and Entity Behavior Analytics*) y los registros de autenticación, cobran especial relevancia las contribuciones de Nisioti *et al*. [16], Kandah *et al*. [17] y Zhang *et al*. [18]. Dichos trabajos resultan fundamentales para la presente investigación, puesto que ponen de manifiesto tanto las capacidades disruptivas como los condicionantes operativos que enfrentan los enfoques no supervisados al ser desplegados en infraestructuras corporativas reales.

III. Metodología

La metodología propuesta para la detección de anomalías en patrones de acceso se estructura en un flujo de trabajo compuesto por tres etapas principales: generación de datos, preprocesamiento e ingeniería de características, y modelado con evaluación comparativa, como se muestra en la Fig. 1.

Esta estructura permite garantizar una comparación homogénea entre los modelos evaluados y reducir sesgos derivados de diferencias en los datos o en el proceso de entrenamiento.

El uso de un conjunto de datos sintético y no público responde a restricciones prácticas, legales y de privacidad asociadas a los registros reales de Active Directory, los cuales contienen información altamente sensible relacionada con identidades, credenciales y patrones de acceso de usuarios.

En entornos organizacionales reales, este tipo de datos se encuentra protegido por políticas de seguridad y confidencialidad que limitan su divulgación para fines académicos.

En este contexto, la simulación controlada se adopta como una alternativa metodológica válida, permitiendo reproducir comportamientos realistas, inyectar anomalías de forma explícita y evaluar comparativamente distintos modelos bajo condiciones homogéneas.

No obstante, se reconoce que los resultados obtenidos no pretenden generalizarse directamente a entornos de producción, sino servir como una evaluación metodológica del comportamiento relativo de los modelos analizados.

III-A. Generación de datos de registros

Se desarrolló un programa en Python para simular un entorno corporativo y generar registros de eventos con características similares a las producidas por controladores de dominio de Active Directory. Los datos se almacenan en formato CSV e incluyen atributos temporales, de identidad de usuario, de contexto de acceso y campos de control que indican la presencia y el tipo de anomalía.

La simulación contempla perfiles de usuario diferenciados, tales como usuarios a tiempo completo, usuarios de tiempo parcial y cuentas de servicio, cada uno con patrones específicos de actividad, horarios de trabajo y conjuntos de direcciones IP y estaciones de trabajo asociadas. Asimismo, se generan eventos de seguridad habituales, como inicios de sesión exitosos, fallidos y cierres de sesión, así como eventos correlacionados del sistema que aportan mayor realismo al conjunto de datos.

Las anomalías se inyectan de forma controlada e incluyen escenarios como ataques de fuerza bruta, uso de credenciales comprometidas e inicios de sesión fuera del horario laboral. Estas anomalías se etiquetan explícitamente con fines de validación, sin ser utilizadas durante el entrenamiento de los modelos.

En total, se generaron aproximadamente 120 000 eventos correspondientes a más de 570 usuarios únicos, abarcando un período de varios meses. El conjunto de entrenamiento se caracteriza por una baja tasa de anomalías (aproximadamente 0,5 %), mientras que el conjunto de prueba incluye un escenario con una tasa de anomalías significativamente mayor,

permitiendo evaluar el comportamiento de los modelos bajo condiciones extremas.

La partición de los datos sigue un criterio temporal, utilizando el 70 % de los eventos más antiguos para el entrenamiento y el 30 % restante para la evaluación, con el objetivo de evitar la fuga de datos y simular un escenario de detección en línea.

III-B. Preprocesamiento e ingeniería de características

El preprocesamiento transforma los registros brutos en representaciones numéricas adecuadas para los algoritmos de aprendizaje automático. Este proceso incluye la ordenación temporal de los eventos, la extracción de características temporales y la codificación de variables categóricas.

La marca temporal se convierte al formato *datetime* y se extrae la hora del día como característica numérica. Las variables categóricas relevantes se codifican mediante un *Ordinal Encoder*, asignando un valor específico a categorías no vistas durante el entrenamiento. Posteriormente, todas las características numéricas se normalizan utilizando *StandardScaler*, garantizando escalas comparables y favoreciendo la estabilidad del entrenamiento.

De forma adicional, se explora el uso de entropía de Markov para capturar la imprevisibilidad en las transiciones entre estaciones de trabajo, incorporando información secuencial al vector de características y enriqueciendo la representación del comportamiento normal.

III-C. Modelos y configuración experimental

Se evaluaron cinco modelos ampliamente utilizados en la literatura para la detección de anomalías en conjuntos de datos desbalanceados: One-Class Support Vector Machine (OCSVM), Local Outlier Factor (LOF), Isolation Forest (IF), Autoencoder (AE) y Perceptrón Multicapa (MLP).

Los modelos OCSVM, LOF e IF representan enfoques tradicionales basados en fronteras, densidad y aislamiento, respectivamente. Estos modelos se entrenan exclusivamente con datos considerados normales y producen una puntuación de anomalía para cada instancia evaluada.

Los modelos basados en redes neuronales (AE y MLP) se entrenaron como detectores de novedad, aprendiendo a reconstruir el comportamiento normal a partir de los datos de entrenamiento. En ambos casos, las anomalías se identifican a partir de un aumento significativo del error de reconstrucción. En el caso del MLP, el perfil de comportamiento normal de cada usuario se aprende a partir de su historial consolidado de actividad, simulando un escenario realista de análisis de comportamiento de usuarios.

Para garantizar la comparabilidad entre enfoques, se utilizaron configuraciones estándar para cada algoritmo, fijando el parámetro de contaminación en 1 % para todos los modelos. El MLP se configuró con una arquitectura de dos capas ocultas de tamaños [32, 16], función de activación ReLU y normalización por lotes para mejorar la estabilidad del entrenamiento. La optimización se realizó mediante el algoritmo Adam [19], utilizando el error cuadrático medio como función de pérdida y una tasa de aprendizaje de 0,001.

Modelo	Precisión	Sensibilidad	F1-score	Exactitud
IF	0.1714	0.2143	0.1905	0.4870
LOF	0.3043	0.2500	0.2745	0.5698
OCSVM	0.3714	0.4643	0.4127	0.5698
AE	0.9333	**1.0000**	0.9655	0.9767
MLP	**0.9655**	**1.0000**	**0.9825**	**0.9884**

Tabla I: Consolidación de los resultados

III-D. Estrategia de evaluación

La evaluación del desempeño se realizó utilizando métricas adecuadas para conjuntos de datos desbalanceados, incluyendo precisión, sensibilidad (*recall*), *F1-score* y exactitud. De forma complementaria, se analizaron las matrices de confusión para caracterizar el comportamiento de los modelos en términos de falsos positivos y falsos negativos.

Cabe destacar que el objetivo principal de esta evaluación es comparar el comportamiento relativo de los modelos bajo condiciones controladas, y no establecer valores absolutos de desempeño esperables en entornos de producción reales.

Con el fin de favorecer la reproducibilidad metodológica del estudio, los scripts de implementación utilizados para la generación de datos, el preprocesamiento y la evaluación de los modelos se encuentran disponibles bajo solicitud directa a los autores.

No obstante, los conjuntos de datos empleados no pueden ser divulgados públicamente debido a restricciones institucionales y de seguridad asociadas a registros de autenticación y comportamiento de usuarios en entornos reales.

IV. Resultados

Los resultados obtenidos a partir del flujo de trabajo propuesto permiten comparar el desempeño de los distintos modelos de detección de anomalías bajo condiciones controladas. Las métricas utilizadas incluyen precisión, sensibilidad (*recall*), *F1-score* y exactitud, las cuales son adecuadas para evaluar conjuntos de datos altamente desbalanceados.

Como se observa en la Tabla I, los modelos tradicionales basados en densidad, fronteras o aislamiento (IF, LOF y OCSVM) presentaron un desempeño limitado, caracterizado por valores bajos de precisión y sensibilidad. Este comportamiento indica una elevada proporción de falsos negativos y falsos positivos, lo que evidencia dificultades para modelar adecuadamente patrones complejos de comportamiento en registros de Active Directory.

Por otro lado, los modelos basados en redes neuronales (AE y MLP) mostraron un desempeño significativamente superior. Ambos alcanzaron una sensibilidad de 1.0000, valor adimensional que indica la detección de todas las anomalías presentes en el conjunto de prueba. En términos prácticos, este resultado implica la ausencia de falsos negativos bajo las condiciones evaluadas.

A pesar de que tanto el AE como el MLP lograron una detección completa de las anomalías, el MLP presentó una ventaja adicional en términos de precisión, alcanzando un valor de 0.9655 frente a 0.9333 del AE. Esta diferencia refleja una menor tasa de falsos positivos lo que resulta especialmente relevante en escenarios operacionales donde

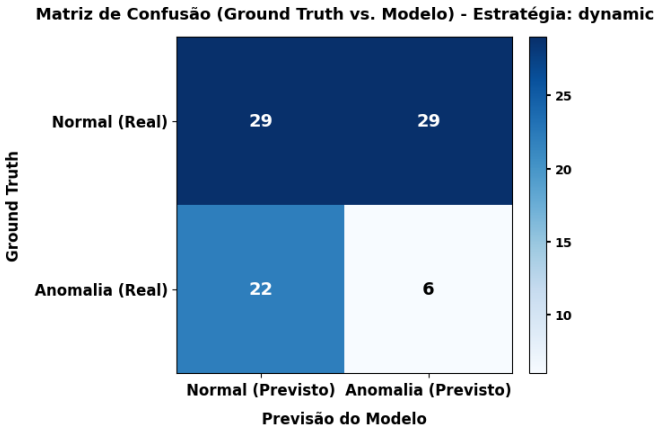


Figura 2. Matriz de confusión del modelo MLP

el exceso de alertas puede sobrecargar a los equipos de seguridad.

Es importante destacar que el desempeño observado, en particular la sensibilidad perfecta obtenida por los modelos de aprendizaje profundo, se alcanzó en un escenario experimental controlado, con anomalías inyectadas de forma explícita y un conjunto de datos diseñado para evaluar la viabilidad metodológica del enfoque propuesto.

En consecuencia, estos resultados no deben interpretarse como una garantía de rendimiento en entornos de producción reales, donde factores como la mayor complejidad de los registros, la deriva conceptual y la variabilidad del comportamiento de los usuarios pueden impactar negativamente el desempeño de los modelos.

Finalmente, la detección de anomalías en entornos reales enfrenta desafíos como la alta complejidad de los registros de producción, la necesidad de reajustar continuamente los umbrales de decisión y la presencia de deriva conceptual (*concept drift*), lo que puede requerir reentrenamientos periódicos de los modelos [20].

Además, la interpretabilidad limitada de los modelos de aprendizaje profundo dificulta el análisis de la causa raíz de las anomalías detectadas [21].

A diferencia de investigaciones previas que se limitan al análisis de modelos individuales o arquitecturas aisladas —casos emblemáticos de LogAnomaly [6] y DeepLog [7]—, el presente estudio adopta una perspectiva integradora. Se propone un análisis comparativo que somete tanto a modelos clásicos como a redes de aprendizaje profundo a un mismo marco experimental riguroso.

Sin embargo, nos distanciamos de enfoques extremadamente complejos como LogGD [8] al situar la interpretabilidad y la viabilidad operativa en el centro de la propuesta. En lugar de priorizar únicamente la métrica bruta, enfatizamos la creación de un flujo de trabajo reproducible que responda a las necesidades reales de seguridad en infraestructuras corporativas.

IV-A. Análisis del rendimiento del MLP

El rendimiento del modelo Perceptrón Multicapa (MLP) se fundamenta en su capacidad para capturar relaciones no lineales complejas entre las dimensiones temporales, categóricas y contextuales de los registros. En contraste con enfoques tradicionales como Isolation Forest o LOF, los cuales operan principalmente bajo métricas de distancia o densidad, el MLP logra codificar representaciones jerárquicas del comportamiento normal de los usuarios.

Dicha arquitectura, integrada por dos capas ocultas con dimensiones [32, 16] y una función de activación ReLU (Rectified Linear Unit), posibilita el modelado de interdependencias sutiles, tales como la correlación entre patrones horarios de acceso y contextos específicos de autenticación. La implementación de ReLU resulta estratégica, ya que introduce no linealidades al activar exclusivamente valores positivos; esto no solo favorece una representación eficiente de los datos, sino que mitiga la atenuación del gradiente, un desafío técnico recurrente en entornos donde los registros presentan distribuciones asimétricas y transiciones abruptas.

Asimismo, la configuración del MLP como un detector de novedad —orientado a minimizar el error cuadrático medio sobre instancias normales— potencia la identificación de desviaciones multivariadas que podrían pasar inadvertidas en análisis univariados. Esta facultad técnica sustenta el desempeño superior del modelo en términos de sensibilidad (recall) y F1-score dentro del marco experimental.

No obstante, este rigor en la detección conlleva un incremento notable en la tasa de falsos positivos, tal como se observa en la Fig. 2. Dicha tendencia no fue interpretada como una falla del modelo, sino como un punto de ajuste dependiente de la estrategia de seguridad adoptada en un entorno específico de Active Directory. En este sentido, la maximización de la sensibilidad implica necesariamente la definición de prioridades en la gestión de alertas, de modo que se alcance un coste operativo aceptable frente al riesgo de omitir una intrusión real. Por tanto, la eficacia del MLP se define por su capacidad para blindar el sistema contra falsos negativos, garantizando una supervisión exhaustiva de la red.

IV-B. Limitaciones observadas

A pesar de los resultados favorables, el estudio presenta limitaciones inherentes al diseño experimental. El uso de datos sintéticos y la inyección controlada de anomalías simplifican el problema en comparación con escenarios reales, donde los registros pueden contener ruido, inconsistencias y patrones no observados durante el entrenamiento.

Además, la necesidad de reajustar umbrales y de reentrenar los modelos frente a la deriva conceptual representa un desafío adicional para su adopción en entornos operacionales.

Asimismo, resulta prometedor explorar enfoques de aprendizaje multifuente que integren conocimiento humano y datos procedentes de distintas fuentes, como métricas del sistema y etiquetas de incidentes, junto con los registros de eventos [22].

Cabe mencionar que la incorporación de la entropía de Markov como característica adicional contribuyó a una mayor estabilidad en la detección de transiciones de comportamiento de los usuarios, especialmente en escenarios con cambios abruptos de contexto.

Sin embargo, el impacto cuantitativo aislado de esta característica no fue evaluado de forma independiente en el presente estudio y será analizado en trabajos futuros.

V. Conclusión

En este trabajo se presentó una evaluación comparativa de distintos modelos de aprendizaje automático y aprendizaje profundo aplicados a la detección de anomalías en patrones de acceso extraídos de registros de Active Directory.

A partir de un flujo de trabajo estructurado que abarca la generación de datos, el preprocesamiento, la ingeniería de características y la evaluación homogénea de modelos, fue posible analizar el comportamiento relativo de enfoques tradicionales y basados en redes neuronales bajo un escenario controlado.

Los resultados obtenidos indican que los modelos de aprendizaje profundo, en particular el Perceptrón Multicapa, presentan una mayor capacidad para modelar patrones complejos de comportamiento de usuarios y detectar desviaciones multivariadas en comparación con métodos tradicionales basados en densidad, fronteras o aislamiento. Este desempeño se refleja en valores superiores de *F1-score* y sensibilidad dentro del entorno experimental considerado.

No obstante, el estudio también pone de manifiesto limitaciones relevantes. El uso de datos sintéticos y la inyección controlada de anomalías simplifican el problema en relación con entornos de producción reales, donde los registros suelen presentar mayor heterogeneidad, ruido y variabilidad temporal.

En consecuencia, los resultados no deben interpretarse como una estimación directa del rendimiento esperado en escenarios operacionales, sino como una validación metodológica del comportamiento relativo de los modelos evaluados.

Desde una perspectiva práctica, los hallazgos sugieren que los modelos basados en redes neuronales pueden constituir un componente valioso en sistemas de monitorización de seguridad orientados al análisis de comportamiento de usuarios y entidades. Sin embargo, su adopción en entornos reales requiere abordar desafíos adicionales, como el coste computacional, la interpretabilidad de los resultados, la gestión de falsos positivos y la adaptación continua frente a la deriva conceptual.

En conjunto, este trabajo contribuye a una mejor comprensión de las ventajas y limitaciones de distintos enfoques de detección de anomalías en registros de Active Directory, estableciendo una base para investigaciones futuras orientadas a la evaluación en datos reales, la mejora de la explicabilidad y el desarrollo de soluciones más robustas y aplicables en contextos operacionales.

ACKNOWLEDGEMENTS: The authors would like to thank: the support of the LATITUDE Laboratory of the University of Brasília; the support of the Attorney General's Office of the National Treasury – PGFN (No. PGFN 23106.148934/2019-67); the support of CNPq – National Council for Scientific and Technological Development (No. PQ-2312180/2019-5 and Cybersecurity No. 465741/2014-2); Support from the Research Support Foundation of the Federal District – FAPDF with the project "Construction of natural language models for data processing in open sources" (no. 00193-00002360/2023-94, Grant and Acceptance Agreement 519/2023 - FAPDF/SUCTI/COOTI), with the project "SISTER City – Secure and Real-Time Intelligent Systems for Smart Cities" (Grant and Acceptance Agreement 625/2022); support from the Ministry of Justice and Public Security – MJSP through TED 23106.051558/2019-99.

Referencias

[1] R. Kaur, D. Gabrijelčič, and T. Klobučar, "Artificial Intelligence for Cybersecurity: Literature Review and Future Research Directions," *Inf. Fusion*, vol. 97, p. 101804, 2023. doi:10.1016/j.inffus.2023.101804.

[2] S. He, P. He, Z. Chen, T. Yang, Y. Su, and M. R. Lyu, "A Survey on Automated Log Analysis for Reliability Engineering," *ACM Comput. Surv.*, vol. 54, no. 1, Art. 1, 2021. doi:10.1145/3414551.

[3] M. Landauer, S. Onder, F. Skopik, and M. Wurzenberger, "Deep Learning for Anomaly Detection in Log Data: A Survey," *arXiv:2207.08320*, 2022.

[4] M. Esnaashari and N. Moradi, "Predicting Vulnerability to Malware Using Machine Learning Models," *arXiv:2501.02493*, 2025.

[5] D. Zhuravchak, M. Opanovych, A. Tolokachova, V. Dudkeych, and A. Piskozub, "Design of an Integrated Defense-in-Depth System with an Artificial Intelligence Assistant to Counter Malware," *East.-Eur. J. Enterprise Technol.*, vol. 6, no. 123, pp. 64–73, 2024. doi:10.15587/1729-4061.2024.318336.

[6] W. Meng, Y. Liu, Y. Chen, X. Zhang, and D. Pei, "LogAnomaly: Unsupervised Detection of Sequential and Quantitative Anomalies in Logs," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2019, pp. 4739–4745.

[7] M. Du, F. Li, G. Zheng, and V. Srikumar, "DeepLog: Anomaly Detection and Diagnosis from System Logs," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, 2017, pp. 1285–1298. doi:10.1145/3133956.3134015.

[8] Y. Xie, H. Zhang, and M. A. Babar, "LogGD: Detecting Anomalies from System Logs with Graph Neural Networks," in *Proc. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, 2022.

[9] Z. Z. Darban, G. I. Webb, S. Pan, C. C. Aggarwal, and M. Salehi, "Deep Learning for Time Series Anomaly Detection: A Survey," *arXiv:2211.05244*, 2023. doi:10.48550/arXiv.2211.05244.

[10] T. Hayashi, D. Chiba, and K. Fujita, "Critical Review for One-Class Classification," *IEEE Access*, vol. 12, 2024. doi:10.1109/ACCESS.2024.3380404.

[11] A. M. Jabbar, "Local and Global Outlier Detection Algorithms in Unsupervised Approach: A Review," *Iraqi J. Electr. Electron. Eng.*, vol. 17, no. 1, 2021. doi:10.37917/ijeee.17.1.9.

[12] I. D. Mienye and T. G. Swart, "Deep Autoencoder Neural Networks: A Comprehensive Review," *Arch. Comput. Methods Eng.*, 2025. doi:10.1007/s11831-025-10065-6.

[13] W. S. Al Farizi, I. Hidayah, and M. N. Rizal, "Isolation Forest Based Anomaly Detection: A Systematic Literature Review," in *Proc. Int. Conf. Inf. Technol. Comput. Electr. Eng. (ICITACEE)*, 2021, pp. 118–122. doi:10.1109/ICITACEE53184.2021.9617498.

[14] A. Rawat, S. Dahiya, and A. Rana, "Application of Multi Layer Perceptron Neural Network in Diagnosis Systems: A Review," 2018.

[15] S. He, J. Zhu, P. He, and M. R. Lyu, "Log-Based Anomaly Detection: A Survey," *IEEE Access*, vol. 9, pp. 164654–164678, 2021. doi:10.1109/ACCESS.2021.3122936.

[16] A. Nisioti, A. Mylonas, P. Yoo, and M. Katos, "User and Entity Behavior Analytics for Insider Threat Detection," *IEEE Access*, vol. 7, pp. 136674–136700, 2019. doi:10.1109/ACCESS.2019.2946990.

[17] F. Kandah, S. Bentahar, and M. Guizani, "Anomaly Detection in Authentication Logs Using Machine Learning," *Future Gener. Comput. Syst.*, vol. 129, pp. 90–102, 2022.

[18] Y. Zhang, Y. Wang, and Z. Wang, "Unsupervised Anomaly Detection in Enterprise Authentication Logs," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 3567–3581, 2023.

[19] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015.

[20] X. Zhang *et al.*, "Robust Log-Based Anomaly Detection on Unstable Log Data," in *Proc. ACM Joint Eur. Softw. Eng. Conf. Symp. Found. Softw. Eng. (ESEC/FSE)*, 2019.

[21] Z. Chen *et al.*, "Experience Report: Deep Learning-Based System Log Analysis," *arXiv:2107.05908*, 2021.

[22] R. Chen *et al.*, "LogTransfer: Cross-System Log Anomaly Detection for Software Systems with Transfer Learning," in *Proc. IEEE Int. Symp. Softw. Reliab. Eng. (ISSRE)*, 2020.

VESTAL: Marco holístico de ciberseguridad personal

Ángel Suárez-Bárcena
Grupo de Seguridad y Auditoría, GSyA
Universidad de Castilla-La Mancha, Ciudad Real, España
Angel.SBarcena@uclm.es

Jose Luis Ruiz
Grupo de Seguridad y Auditoría, GSyA
Universidad de Castilla-La Mancha, Ciudad Real, España
JoseLuis.Ruiz@uclm.es

Antonio Santos-Olmo
Grupo de Seguridad y Auditoría, GSyA
Universidad de Castilla-La Mancha, Ciudad Real, España
Antonio.SantosOlmo@uclm.es

Eduardo Fernández-Medina
Grupo de Seguridad y Auditoría, GSyA
Universidad de Castilla-La Mancha, Ciudad Real, España
Eduardo.FdezMedina@uclm.es

***Abstract*—La expansión del mundo digital nos rodea cada vez más, facilitando nuestras vidas y haciéndolas más interconectadas. Sin embargo, también introduce una serie de riesgos y amenazas que aceptamos, en ocasiones de manera consciente y en otras de forma inconsciente, en favor de la productividad o la comodidad. Estos riesgos de ciberseguridad afectan a prácticamente todos los sectores de la sociedad y han sido analizados principalmente en contextos gubernamentales y corporativos. A pesar de ello, siguen faltando estudios exhaustivos centrados en el ámbito personal y doméstico. En este dominio, los riesgos de ciberseguridad presentan un amplio alcance: pueden comprometer la seguridad física de los individuos, provocar violaciones de privacidad o exponer datos personales para su uso con fines delictivos. Asimismo, pueden surgir muchas otras situaciones que impactan de forma significativa a las personas en su entorno digital cotidiano. Este artículo analiza trabajos previos relacionados con la ausencia de soluciones en este sentido y el estado del arte de la ciberseguridad personal, y presenta una propuesta de marco de ciberseguridad personal y un modelo conceptual de aplicación realista al que denominaremos VESTAL. El framework de ciberseguridad personal completará las carencias de los modelos previos y su aplicación, en esta versión inicial, permitirá proteger a los individuos bajo esta solución integral.**

***Index Terms*—cybersecurity, risk analysis, personal cybersecurity, personal devices, domestic digital devices, personal data protection, VESTAL framework**

I. Introducción

El dominio digital está experimentando una expansión exponencial, impulsada por la innovación tecnológica y la proliferación de servicios en línea que actualmente impregnan todas las áreas de la actividad humana, desde la comunicación interpersonal hasta las operaciones militares. Esta rápida evolución ha priorizado la conectividad, la accesibilidad a los servicios y la rentabilidad por encima de otras consideraciones [5], convirtiendo la interacción digital en un elemento indispensable para la productividad, la industria, el comercio, la administración e incluso las relaciones personales. En consecuencia, los individuos se encuentran inevitablemente inmersos en un entorno hiperconectado que conlleva riesgos personales inherentes, asumidos con frecuencia de manera consciente o inconsciente.

Podemos afirmar que la ciberseguridad es la seguridad de una sociedad cada vez más conectada por la convergencia de tecnologías y por sus implicaciones en la seguridad física, y por tanto debe abordarse en los tres dominios principales de la actividad digital: gobiernos, empresas e individuos. Los gobiernos han desarrollado estrategias nacionales de seguridad de carácter restrictivo que, si bien resultan efectivas para los estados, no son replicables por el sector privado debido a su rigidez y falta de eficiencia en costes [4]. Las empresas, en cambio, han implementado una amplia variedad de medidas de protección y han adoptado estándares internacionalmente reconocidos, tales como los marcos ISO [1], [7], NIST [8], [28], OWASP [23], MITRE [27], CIS [29], ENISA [30], ISACA [31] e IEC [32]. Estos marcos definen prácticas corporativas de ciberseguridad, principios de gobernanza, mecanismos de protección de datos y resiliencia. Sin embargo, los individuos y los hogares permanecen en gran medida excluidos de estos modelos debido a la heterogeneidad de sus actividades digitales [2] y a la enorme diversidad de servicios, dispositivos y aplicaciones utilizados en la vida cotidiana como mostramos en la Fig. 1. Esta diversidad pone de manifiesto la complejidad de desarrollar un enfoque integrado de ciberseguridad a nivel personal.

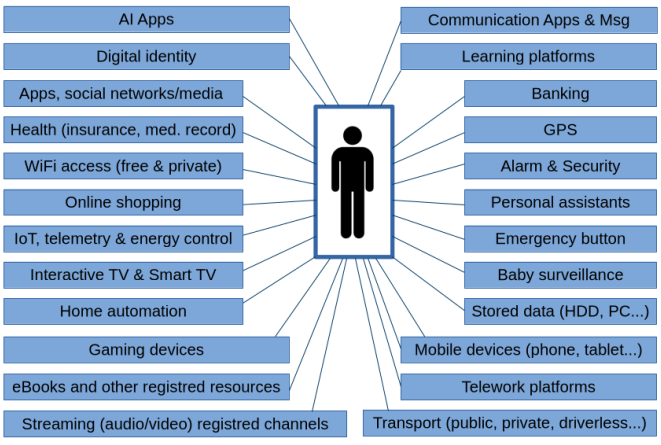


Fig. 1. Common personal digital services.

Los individuos constituyen un grupo heterogéneo y vulnerable, que incluye a niños, personas mayores y

personas con discapacidades cognitivas o físicas. Muchos carecen de conciencia sobre los riesgos existentes o de la capacidad técnica para mitigarlos, mientras que otros se ven obligados a aceptar ciertos riesgos para acceder a servicios digitales esenciales por necesidad o conveniencia. [6]. El Instituto Canadiense de Ciberseguridad identificó vulnerabilidades en más de 100 familias de dispositivos personales y domésticos de uso común, muchos de los cuales presentan una alta susceptibilidad a ataques [6]. Existen estudios que indican además que usuarios avanzados, como estudiantes universitarios, perciben una protección insuficiente de sus datos personales [3]. De estos trabajos se derivan tres hallazgos críticos: (1) educación y sensibilización limitadas en materia de protección de datos personales, (2) vulnerabilidades evidentes sin contramedidas efectivas y (3) ausencia de marcos integrales y sostenibles que aborden la esfera individual.

La investigación sobre ciberseguridad del IoT personal y doméstico [6], [9], [11] revela las amplias vulnerabilidades de los dispositivos conectados. Los sistemas de automatización del hogar y los contadores inteligentes recopilan y transmiten automáticamente información detallada de uso, como patrones de consumo, horarios de ocupación y tipos de dispositivos conectados sin el consentimiento explícito del usuario. Como ejemplo, la taxonomía de ciberataques de Heartfield en sistemas de automatización doméstica demuestra que las primeras implementaciones de IoT prestaron una atención mínima a la ciberseguridad incluso asumiendo un riesgo bajo o una capacidad de ataque limitada [9].

A pesar de la creciente conciencia sobre estos riesgos, la mayoría de los marcos de ciberseguridad siguen centrados en contextos corporativos y gubernamentales, ofreciendo una aplicabilidad limitada para las personas. Las propuestas existentes plantean modelos de protección global, pero descuidan el ámbito doméstico, reforzando la necesidad de enfoques específicos para la ciberseguridad de los individuos [5].

Para abordar esta carencia, se introduce el framework VESTAL, un modelo integral que adapta la metodología NIST 2.0, con las recomendaciones ISACA y ENISA, al entorno personal y doméstico. VESTAL establece un ciclo continuo de evaluación de riesgos, monitorización y gestión de incidentes, con el objetivo de proteger a los ciudadanos en sus interacciones digitales cotidianas en contextos multidispositivo y multiaplicación, considerando al mismo tiempo las diferencias entre las personas, la sostenibilidad y la viabilidad de implementación [26], [28], [29], [31], [32]. Basado metodologías de ciberseguridad consolidadas, VESTAL define un conjunto de módulos operativos diseñados específicamente para abordar los problemas más relevantes que afectan a los individuos, garantizando que toda persona, independientemente de su edad, capacidad o circunstancia, pueda estar protegida de manera efectiva dentro de su ecosistema digital.

Los apartados de este trabajo son:
I. Introducción.
II. Estado del arte.
III. Propuesta.
IV. Conclusiones y trabajo futuro.

II. Estado del arte

La revisión de los estudios existentes muestra que, aunque la ciberseguridad personal ha sido tratada de manera indirecta a través de investigaciones sobre IoT y hogares inteligentes, todavía no existe un marco integral y centrado en el usuario diseñado para proteger a las personas y sus datos dentro de entornos digitales domésticos. La literatura disponible ofrece numerosos análisis de vulnerabilidades y propuestas técnicas fragmentadas, pero ninguna integra el conjunto heterogéneo de dispositivos, servicios y comportamientos de usuario que caracterizan los ecosistemas personales reales.

Mishra [2] destaca la exposición individual como un problema crítico, pero se centra exclusivamente en las responsabilidades gubernamentales sin aportar soluciones prácticas para los usuarios finales. Bhatnagar [3] demuestra una baja concienciación sobre los riesgos digitales incluso entre usuarios altamente alfabetizados, lo que indica que los marcos institucionales tradicionales, como NIST, resultan insuficientes en contextos personales. Hartnett [4] e ICCWS [5] subrayan la necesidad de la participación ciudadana y de capas de protección personal, pero sus planteamientos permanecen en gran medida a nivel conceptual. Estudios experimentales, como el de Neto [6], confirman empíricamente la existencia generalizada de vulnerabilidades en dispositivos de consumo, aunque siguen limitados a escenarios IoT. Del mismo modo, Ghazal [10], Gupta [11], Khan [12], Haleem [14] y Yang [13] muestran que las estrategias de protección se limitan a dispositivos, aplicaciones o capas de red específicas, sin abordar la integración entre dominios ni la adaptabilidad al usuario.

Los mecanismos propuestos, como soluciones de autenticación o basadas en blockchain [15][16], permanecen aislados y no integrados. Heartfield [9] ofrece una taxonomía detallada de ataques en IoT doméstico, confirmando el nivel de exposición de los entornos del hogar, aunque sin proporcionar un marco desplegable para la protección personal.

Las soluciones existentes resultan inadecuadas para la ciberseguridad personal por varios factores: heterogeneidad de dispositivos y fabricantes [9], [11], [12], capacidad limitada para incorporar capas de protección en dispositivos prediseñados [14], [15], formación insuficiente de los usuarios [13], situación inicial de la cultura de la persona respecto de su comprensión de la seguridad de la información, complejidad derivada de la diversidad de dispositivos y servicios [10], compromiso entre el control remoto del proveedor y la seguridad del usuario [4], [8], y falta de legislación adecuada [2], [30]. Algunos proveedores de servicios explotan los datos personales con fines comerciales, a menudo sin consentimiento explícito del usuario [1], [7], [25], mientras que algunos fabricantes no incorporan protección efectiva debido a la complejidad técnica [12], [27].

Un análisis metodológico de vulnerabilidades y riesgos en una amplia gama de dispositivos, aplicaciones y tipos de usuario es factible [3], [13] y precísamente este enfoque resulta especialmente importante en los entornos personales y domésticos, donde la conectividad del hogar puede comprometer la seguridad física, la integridad de los datos y la privacidad [9], [10], [12]. Esta diversidad también exige

sostenibilidad y adaptabilidad en las soluciones [16], [26], [25], garantizando continuidad, facilidad de uso, resiliencia frente a amenazas emergentes e integración de medidas de seguridad impulsadas por el usuario [8], [28].

Se evidencian cinco grandes carencias: (1) un enfoque excesivo en el IoT, ignorando el entorno digital más amplio; (2) propuestas dirigidas a instituciones en lugar de a los usuarios finales; (3) falta de integración entre dispositivos, aplicaciones y factores conductuales; (4) escasez de implementaciones reales; y (5) ausencia de arquitecturas adaptativas y centradas en el usuario.

En consecuencia, a pesar de las numerosas contribuciones que describen vulnerabilidades, no existe ningún modelo que proporcione una ciberseguridad integral, adaptativa y centrada en el usuario para las personas en su entorno doméstico. Esta carencia motiva el desarrollo del VESTAL Framework, diseñado para unificar las dimensiones técnicas, conductuales y de supervisión de la protección, cerrando la brecha entre el análisis teórico de vulnerabilidades y una ciberseguridad personal efectiva y desplegable.

III. Propuesta

Con todo el trabajo realizado, se han identificado debilidades y deficiencias en la protección de los ciudadanos en su entorno personal y doméstico, ya que la mayoría de las propuestas existentes se centran en empresas y gobiernos. Por este motivo se propone el marco de trabajo VESTAL y una aproximación a su aplicación real, que integra los procedimientos necesarios para analizar y evaluar eventos de riesgo o ciberataques y gestionarlos adecuadamente en un ”paraguas de ciberseguridad personal”.

Este marco busca prevenir y mitigar ataques mediante la formación continua de los individuos, la capacitación y la monitorización permanente de los esquemas de seguridad de cada categoría. Su idea central es establecer un ciclo adaptativo para la evaluación, monitorización y gestión de los activos digitales y sus riesgos asociados, estructurado en tres grandes bloques, tal como se muestra en la Fig. 2.

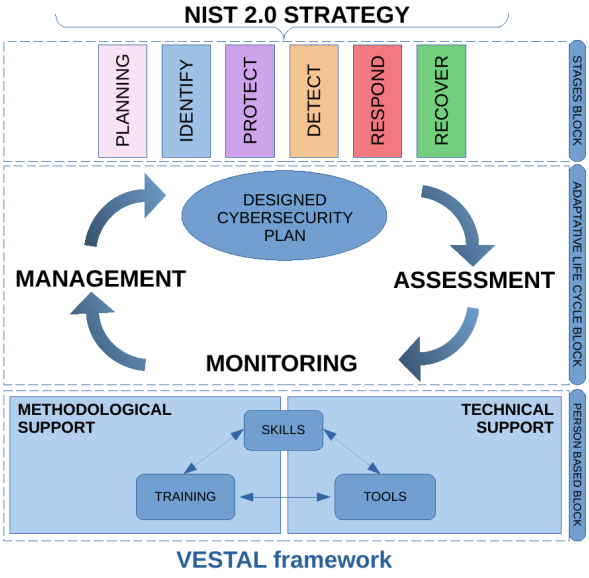


Fig. 2. VESTAL framework.

A. Bloque de etapas

El primer bloque se basa en el NIST Cybersecurity Framework 2.0 (Feb'24) [8], incorporando sus estrategias con las modificaciones necesarias para la ciberseguridad ciudadana. Las diferencias introducidas complementan el marco con las debilidades y deficiencias identificadas en la sección de trabajos relacionados, añadiendo además la etapa de planificación. Detrás de cada una de las ampliaciones hay una justificación realista adaptada a las necesidades de protección de las personas.

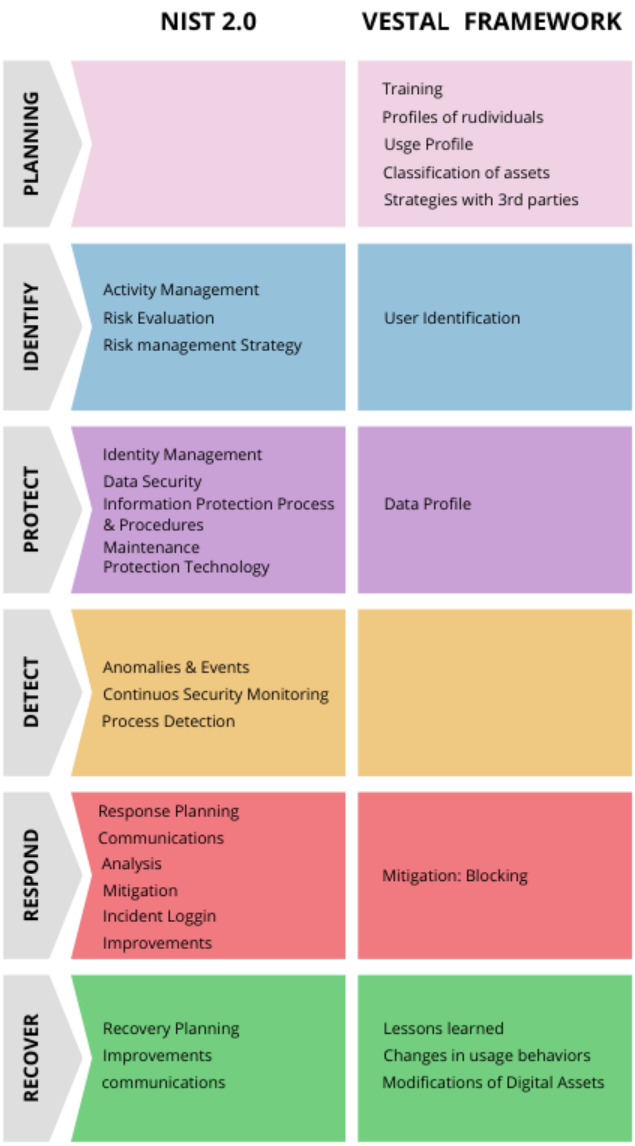


Fig. 3. Qué añade VESTAL a NIST 2.0.

A partir del NIST 2.0, se incorporan subetapas dentro de cada componente:

i) PLANNING: Su incorporación responde a la necesidad de adaptar los criterios del NIST al entorno personal y doméstico. Los modelos existentes se basan en organizaciones; en el ámbito personal se requiere planificación adicional en:

- Formación de los individuos: Determinar qué formación requiere cada usuario según su nivel técnico.

- Perfiles de los individuos: Cada persona necesita una planificación de ciberseguridad distinta.
- Perfiles de uso: Según los activos y su utilización, será necesaria una planificación diferenciada.
- Clasificación de activos y tecnologías: Dada la gran variedad de dispositivos y aplicaciones personales, deben agruparse para ofrecer soluciones comunes.
- Estrategias con terceros: Identificación de proveedores, estándares de cumplimiento y opciones de configuración o seguridad que puedan activarse en servicios o dispositivos. Certificación de terceros en este cumplimiento.

ii) IDENTIFY: Se añade una subetapa:

- Identificación del usuario: Antes de gestionar identidades en PROTECT, debe identificarse inequívocamente al usuario o usuarios del activo digital. Según el perfil (niños, mayores, personas con dificultades, etc.), la estrategia cambia.

iii) PROTECT: Además de garantizar la protección clásica de NIST, VESTAL añade:

- Perfil de datos: Los datos personales sensibles deben categorizarse y tratarse de forma diferenciada.
- Control parental: Cuando aplique, requiere reglas de protección y auditoría específicas.
- Datos no distribuibles: Identificación de datos que nunca deben entregarse, compartirse o ser accesibles.

iv) DETECT: No se añaden subetapas. Debe ser transparente para los usuarios digitales.

v) RESPOND: Además de las subetapas NIST, se añade la posibilidad de que la mitigación pueda requerir un bloqueo parcial o completo de comunicaciones.

vi) RECOVER: Se añaden:

- Lecciones aprendidas: Inclusión de herramientas para recopilar y compartir conocimiento útil.
- Cambios en comportamientos de uso: Ajustes (al usuario) de hábitos tecnológicos cuando proceda.
- Modificación de activos digitales: Propuesta de sustitución o eliminación de activos cuando el riesgo supere al beneficio o existan alternativas.

En resumen, VESTAL complementa NIST 2.0 al abordar necesidades de ciberseguridad personal, situando al ciudadano en el centro y proporcionando información completa sobre medidas de protección, incidentes y consecuencias. Por tanto, no diverge del NIST 2.0; lo amplía como se muestra en la Fig. 3. Adicionalmente es un marco holístico al cubrir las necesidades de todo tipo de personas y circunstancias en su ámbito doméstico.

B. Bloque de evaluación, monitorización y gestión

El segundo bloque es el ciclo continuo de evaluación, monitorización y gestión, incorporando los procesos necesarios para garantizar la evolución y adaptación del marco ante posibles eventos. Los procesos responden a las características deseables de un marco de ciberseguridad personal.

Se divide en tres etapas, como se muestra en la Fig. 4:

ASSESSMENT: Análisis exhaustivo de vulnerabilidades, riesgos y condiciones del usuario para cada activo, con referencia a CVEs y casos de alto impacto. Considera composición familiar, edad, discapacidades y teletrabajo. Evalúa probabilidad e impacto de incidentes, permitiendo aceptación informada del riesgo.

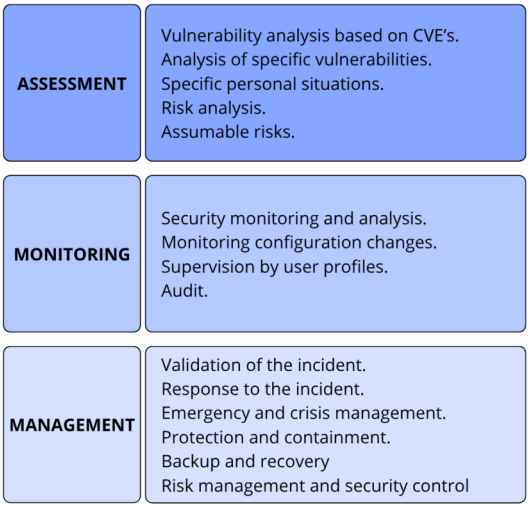


Fig. 4. Proceso de Evaluación, Monitorización y Gestión de VESTAL.

MONITORING: Monitorización adaptativa continua para detectar anomalías, analizar cambios de configuración y ajustarse a entornos en evolución. La supervisión reforzada y la auditoría garantizan cumplimiento y concienciación.

MANAGEMENT: Respuesta estructurada ante incidentes, incluyendo validación, evaluación de severidad, respuesta automatizada, gestión de crisis, protección, contención, recuperación y actualizaciones posteriores para evitar recurrencias.

C. Bloque de formación, habilidades y herramientas

El tercer bloque concentra las acciones dirigidas a garantizar que los individuos estén formados, adquieran habilidades y dispongan de herramientas adecuadas. Se apoya en dos pilares:

- Soporte técnico y tecnológico: Puede ser un desafío; VESTAL pretende simplificar la tarea alcanzando solo el nivel necesario.
- Soporte metodológico: Seguir reglas estructuradas es esencial. El marco propone una metodología suficiente para integrar a los individuos en su propia protección.

A partir de estos soportes se alcanzan formación, desarrollo de habilidades y uso de herramientas.

Una vez definidos los bloques y etapas, deben incorporarse criterios de diseño para asegurar su aplicabilidad en entornos reales. Una arquitectura coherente es esencial para identificar y clasificar los activos digitales en el entorno de un individuo, proporcionando contexto para hardware, software, comunicaciones y servicios. VESTAL mantiene un mapeo actualizado entre activos y amenazas potenciales, empleando OWASP, CAPEC, CVE, CWE, MITRE ATT&CK, NVD ,CSC [17] y ENISA. Esta trazabilidad permite aplicar controles adaptativos alineados con el perfil del usuario y el panorama de amenazas.

VESTAL sigue principios de simplicidad, sostenibilidad, escalabilidad y adaptabilidad. Debe permanecer intuitivo, eficiente y resiliente ante cambios tecnológicos, permitiendo

integrar nuevos dispositivos y fuentes de datos sin ajustes estructurales. Su adaptabilidad garantiza accesibilidad para usuarios de diferentes edades, niveles de alfabetización y capacidades cognitivas, reforzando que la ciberseguridad debe ser inclusiva y centrada en la persona. Como muestra la Fig. 5, el front-end de VESTAL debe ser simple y configurable por un supervisor.

El marco debe generar recomendaciones y acciones prácticas y éticas. Las medidas preventivas requieren concienciación y consentimiento, garantizando transparencia y autonomía. Las defensas automatizadas se activan solo cuando sea necesario, con notificación y auditabilidad. Esta interacción promueve aprendizaje y hábitos digitales más seguros.

VESTAL implementa un modelo de protección por niveles que se adapta a distintos contextos: autonomía y simplicidad en entornos privados; privilegios restringidos en espacios domésticos compartidos; confidencialidad e integridad reforzadas en redes públicas; y modos especiales para menores o usuarios con necesidades específicas. En entornos hostiles, endurece defensas para preservar operaciones esenciales.

La metodología está diseñada para su aplicación práctica, asegurando que los principios sean aplicables en el entorno doméstico y personal, excluyendo contextos profesionales. La interfaz preferida es una aplicación móvil u ordenador. El marco proporciona múltiples vistas, garantizando accesibilidad, mientras que el personal cualificado puede modificar configuraciones mediante una vista diferenciada.

Dentro del Bloque Basado en la Persona se construye un ciclo técnico y metodológico basado en TOOLS (incluida la aplicación VESTAL), SKILLS (competencias) y TRAINING (mejora de capacidades). TRAINING y SKILLS no son exclusivos del personal técnico; VESTAL promueve una cultura de ciberseguridad basada en conocimiento, concienciación y decisiones informadas, permitiendo aceptar o rechazar medidas o activar o bloquear etapas específicas.

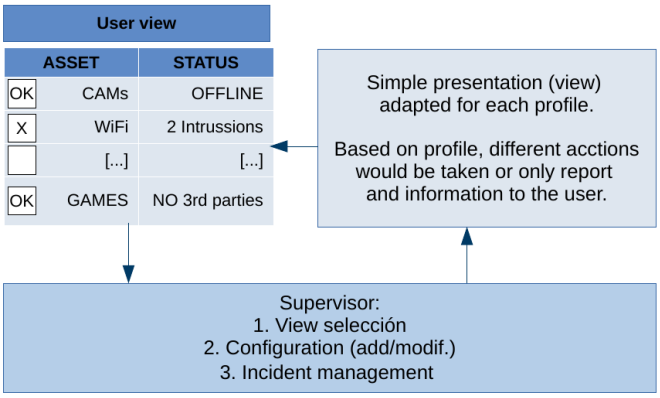


Fig. 5. Proceso de Evaluación, Monitorización y Gestión de VESTAL.

Los niveles de conocimiento y concienciación deben adaptarse a los individuos. Se excluye al personal técnico cualificado, ya que quienes requieren mayor ciberprotección no son ellos. La aplicación y el plan de formación deben considerar la heterogeneidad ciudadana: niños, mayores, cuidadores, personas con dificultades cognitivas, usuarios muy activos, desconectados o invitados, que pueden ser tanto amenaza como víctimas.

Además, deben considerarse dimensiones psicológicas: un adolescente puede proteger su privacidad pero introducir riesgos al instalar aplicaciones; un adulto puede ser consciente de amenazas laborales pero subestimar riesgos domésticos; una persona con deterioro cognitivo puede reaccionar adecuadamente en unos casos y confundirse en otros; la fatiga o la prisa pueden llevar incluso a usuarios formados a cometer errores.

Estas consideraciones se reflejan en las múltiples vistas de VESTAL, que permiten al supervisor mostrar solo lo que cada usuario necesita y puede gestionar.

La interfaz frontal es lo que perciben los usuarios; por debajo se encuentra el Ciclo de Vida Adaptativo definido en el marco. Todo se origina en el diseño del plan de ciberseguridad, elemento central, pues los individuos deben decidir qué nivel desean aplicar. El objetivo no es solo proteger, sino informarles y situarlos en el centro de la toma de decisiones.

Este plan, implementado por el supervisor, considera el perfil de cada usuario para determinar lo que se muestra. Algunos elementos son comunes (alarmas, domótica, telemetría energética, GPS) y otros individualizados (redes sociales, contenidos en línea, SmartTV, IA). Ciertas funcionalidades son restringidas (banca, botones de emergencia), habilitándose según la edad o necesidades.

Con esta configuración adaptativa, el sistema entra en un bucle continuo de monitorización y reevaluación de los activos configurados, siguiendo los parámetros del plan y gestionando incidentes clasificados como información relevante, advertencia, alerta o alarma.

Según la parametrización de riesgos o amenazas, se desencadenan dos categorías de acciones: notificación y solicitud de acción al usuario (mensaje emergente o alerta sonora según gravedad), con información simultánea al supervisor; y activación del proceso descrito en el Bloque de Etapas, alineado con el NIST y ampliado por VESTAL. Este proceso introduce elementos de Comunicación con el Usuario y Reporte al Supervisor, registrando incidentes y reacciones del usuario. Esto permite refinar la parametrización del perfil, fomenta la adopción de buenas prácticas y contribuye a la formación continua, además de permitir al supervisor comprender mejor la psicología de usuarios con dificultades, un aspecto a menudo ignorado dado que la ciberseguridad sigue orientada principalmente a expertos, organizaciones y gobiernos.

Esta herramienta de protección de las personas pretende seguir el framework VESTAL de forma integral, evaluando cada activo, realizando una monitorización continua y avisando de los riesgos e incluso tomando decisiones proactivas de protección.

IV. Conclusiones y trabajo futuro

El análisis del estado del arte revela una clara carencia de soluciones de ciberseguridad diseñadas para individuos en sus entornos personales y domésticos. Aunque numerosos estudios reconocen la necesidad de una mayor protección digital para los ciudadanos, la documentación disponible permanece abrumadoramente centrada en las organizaciones y

en las infraestructuras gubernamentales. El interés comercial por salvaguardar a las personas más allá del software de protección tradicional es mínimo, debido en gran medida a la diversidad y complejidad de los ecosistemas digitales personales. Las propuestas existentes, principalmente orientadas a IoT, suelen depender de plataformas controladas por operadores y no logran proporcionar una protección integrada para dispositivos heterogéneos. Además, los análisis más detallados sobre intrusiones en el entorno doméstico limitan su alcance a las redes, la domótica, los dispositivos móviles y los sistemas WiFi, dejando zonas sin cubrir, como los sistemas de alarma, los localizadores GPS, los wearables y las tecnologías de protección de personas mayores, donde la opacidad derivada de la dependencia del proveedor es frecuente.

A la luz de estas deficiencias, este trabajo propone el marco VESTAL, un modelo integral y holístico diseñado para unificar y articular la ciberseguridad en todos los ámbitos relevantes de la vida digital de los ciudadanos. VESTAL cubre un nicho previamente desatendido al ofrecer un enfoque adaptativo, inclusivo y centrado en la persona, que tiene en cuenta diferencias de edad, capacidad técnica, factores físicos y psicológicos. Desde la perspectiva de implementación, Java constituye actualmente el entorno de desarrollo más adecuado debido a su portabilidad y compatibilidad con una amplia variedad de hardware. Python, dada su rápida expansión tecnológica, podría convertirse en la opción preferida a medio plazo. Asimismo, como herramienta de ciberseguridad controlada por el propio usuario, una aplicación basada en VESTAL cumpliría con los requisitos legales y éticos correspondientes.

La línea más inmediata de trabajo futuro se centra en el desarrollo completo de la aplicación VESTAL, integrando el framework con las principales fuentes de inteligencia de amenazas y vulnerabilidades, OWASP, CVE, CWE, MITRE ATT&CK, CAPEC y los informes de amenazas de ENISA. Un paso crítico consistirá en la asociación dinámica de esta información con los activos digitales del usuario. Esta primera versión también serviría como una herramienta valiosa para analizar el comportamiento del usuario y la toma de decisiones, reconociendo que los factores humanos y los patrones psicológicos siguen siendo las dimensiones menos predecibles de la ciberseguridad. Las personas reales, con todos sus perfiles posibles, en el centro de las decisiones.

El trabajo futuro avanza en dos direcciones complementarias. Primero, la integración de VESTAL en una arquitectura de ciberseguridad multivectorial más amplia, que trascienda el ámbito personal donde una línea de investigación que ya está siendo considerada. Segundo, la aplicación del marco en entornos reales e incluso un caso de estudio, en los que ya se está trabajando, concretamente en los escenarios más comunes, con la intención de ampliar a todos los escenarios posibles cubiertos por VESTAL.

En conjunto, estas líneas establecen una base sólida para el desarrollo a largo plazo, asegurando la continuidad de esta iniciativa de ciberseguridad centrada en la persona y abriendo la puerta a un impacto social más amplio.

AGRADECIMIENTOS

Esta publicación no sería posible sin los proyectos: Di4SPDS (PCI2023145980-2), financiado por by MICIU/AEI/10.13039/501100011033 y la Unión Europea (Chist-Era Program); KOSMOS-UCLM (PID2024-155363OB-C44) financiado por MICIU/AEI/10.13039/501100011033 /FEDER, UE; AURORA (SBPLY/24/180225/000074), financiado por la Consejería de Educación, Cultura y Deportes de la JCCM, y FEDER (Programa Operativo de Castilla-La Mancha 2021–2027); RADAR (2025-GRIN-38447) financiado por FEDER; y RED2024-154240-T financiado por MICIU/AEI/10.13039/501100011033

REFERENCES

[1] ISO/IEC. *UNE-EN ISO/IEC 27001:2017 and extensions 27701 and 27017:2021*. (2017–2021).
[2] N. Mishra. *The trade: (cyber) security dilemma and its impact on global cybersecurity governance*. (2020).
[3] N. Bhatnagar and M. Pry. *Student attitudes, awareness, and perceptions of personal privacy and cybersecurity in the use of social media: An initial study*. (2020).
[4] R. J. Harknett and J. A. Stever. *The cybersecurity triad: Government, private sector partners, and the engaged cybersecurity citizen*. (2009).
[5] *Proceedings of the 14th International Conference on Cyber Warfare and Security (ICCWS 2019)*. (2019).
[6] E. C. P. Neto and S. Dadkhah and R. Ferreira and A. Zohourian and R. Lu and A. A. Ghorbani. *CICIoT: A real-time dataset and benchmark for large-scale attacks in IoT environments*. (2023).
[7] ISO/IEC. *UNE-EN ISO/IEC 29101:2021*. (2021).
[8] National Institute of Standards and Technology (NIST). *The NIST Cybersecurity Framework (CSF) 2.0*. (2024).
[9] R. Heartfield et al.. *A taxonomy of cyber-physical threats and impact in the smart home*. (2018).
[10] A. Ghazal and A. Almomani. *Home network security: A review*. (2020).
[11] A. Gupta and S. Kumar. *A survey of IoT-enabled cyber-physical home security systems*. (2019).
[12] A. M. Khan and K. Salah and A. Al-Fuqaha. *Toward secure smart homes: A survey of security algorithms, protocols, and open issues*. (2018).
[13] X. Yang and L. Zhang. *Home cybersecurity behavior and its impact on online security: An empirical analysis*. (2020).
[14] A. Haleem and N. Javaid and Q. Niyaz. *Smart home automation and security system using IoT and Raspberry Pi*. (2018).
[15] K. Noto and Y. Arazaki and Y. Kadobayashi. *Towards practical security solutions for the smart home*. (2018).
[16] T. M. Fernández-Caramés and P. Fraga-Lamas. *A review on the application of blockchain to the next generation of cybersecure industry 4.0 smart factories*. (2018).
[17] SANS Institute. *Top Critical Security Controls (CSC)*.
[18] P. Brereton and B. A. Kitchenham and D. Budgen and M. Turner and M. Khalil. *Lessons from applying the systematic literature review process within the software engineering domain*. (2007).
[19] S. Barat and T. Clark and B. Barn and V. Kulkarni. *A model-based approach to systematic review of research literature*. (2017).
[20] B. Barn and S. Barat and T. Clark. *Conducting systematic literature reviews and systematic mapping studies*. (2017).
[21] J. Steinberg. *Cybersecurity for Dummies*. (2019).
[22] National Institute of Standards and Technology (NIST). *National Initiative for Cybersecurity Education (NICE) Framework*.
[23] OWASP Foundation. *OWASP Top 10—2021: The Ten Most Critical Web Application Security Risks*.
[24] ISO/IEC. *ISO/IEC 27002:2022 — Information Security, Cybersecurity and Privacy Protection — Information Security Controls*. (2022).
[25] ISO/IEC. *ISO/IEC 27001:2022 — Information Security, Cybersecurity and Privacy Protection — Information Security Management Systems — Requirements*. (2022).
[26] NIST. *Framework for Improving Critical Infrastructure Cybersecurity, Version 2.0*.
[27] MITRE Corporation. *MITRE ATT&CK Framework*.
[28] NIST. *Security and Privacy Controls for Information Systems and Organizations (NIST SP 800-53, Rev. 5)*.
[29] Center for Internet Security (CIS). *CIS Critical Security Controls for Effective Cyber Defense, Version 8*.

[30] ENISA. *European Cybersecurity Framework for Certification*.
[31] ISACA. *COBIT 2019 Framework: Governance and Management Objectives*.
[32] IEC. *IEC 62443 Series — Security for Industrial Automation and Control Systems*.

Análisis de Vulnerabilidades sobre el Protocolo de Comunicación DNP 3.0 en Sistemas ICS/SCADA de Infraestructuras Críticas Eléctricas

Jacinto Pérez Garcia, Ana Lucila Sandoval Orozco, Luis Javier García Villalba
Grupo de Análisis, Seguridad y Sistemas (GASS)
Departamento de Ingeniería del Software e Inteligencia Artificial (DISIA)
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {jacintop, asandov, javierv}@ucm.es

***Resumen*—Los sistemas de adquisición de datos y control supervisorio son componentes fundamentales de los sistemas industriales de control que pertenecen a la infraestructura crítica del ámbito eléctrico. Estos han sido estudiados por décadas, pero en el último decenio han incrementado su interconectividad, complejidad y dependencia con respecto a otros sistemas. Asimismo, los datos producidos y enviados juegan un rol esencial en la adquisición y supervisión mediante el protocolo de red distribuida versión 3, que se utilizan para su análisis. Por lo tanto, se han vuelto superficies que son susceptibles a ataques cibernéticos que amenazan la integridad de los datos, en particular la manipulación no autorizada, lo cual puede causar interrupciones y daños importantes en las infraestructuras críticas. Por lo tanto, el propósito de este trabajo es examinar las restricciones, problemas y vulnerabilidades del protocolo de red distribuida versión 3 e introducir metodos para la identificación de patrones anómalos en la comunicación del protocolo.**

DNP3, Infraestructura crítica, SCADA, Vulnerabilidades.

I. Introducción

Los sistemas de control de supervisión y adquisición de datos (SCADA, por sus siglas en inglés) son responsables de administrar y vigilar las infraestructuras críticas en el sector eléctrico. Estas implementaciones están pensadas para automatizar la supervisión, el monitoreo y la regulación de sus procesos, lo que les confiere innumerables ventajas y aplicaciones. En el lugar donde se emplea el protocolo DNP3 para la comunicación entre sus componentes, las redes de comunicación que conectan las subestaciones con sus centros de control correspondientes, así como una estación centralizada de control (maestra) y las subestaciones (remotas), operan en función de eventos. La maestra supervisa, controla y monitorea numerosas remotas que están geográficamente esparcidas. Los dispositivos remotos son equipos específicos que automatizan el campo, como las unidades remotas o los controladores lógicos programables. Monitorean los elementos físicos del sistema de manera presencial. Asimismo, envían periódicamente al centro de control datos sobre el estado de los equipos de campo para su monitoreo, control y análisis. Previo a que SCADA se conectara con Internet, gracias a la creación de aplicaciones en la nube y al uso generalizado de sistemas de información en tiempo real, las redes SCADA solían estar separadas y desconectadas de Internet y otras redes, empleando protocolos propios.[1], En la Figura 1 podemos ver algunos de ellos.

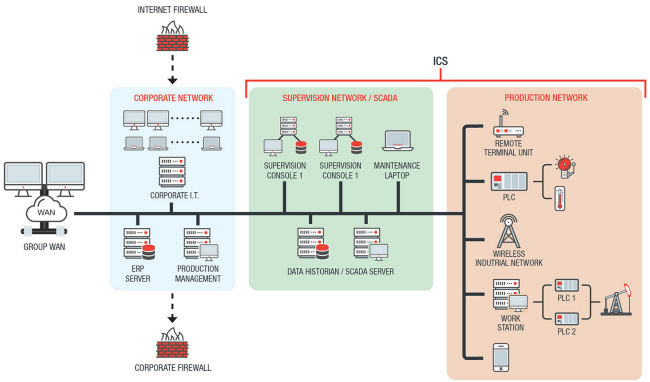


Figura 1. Topología

No obstante, a medida que las aplicaciones de los sistemas se expanden, la superficie de ataque contra estas infraestructuras críticas aumenta exponencialmente.

Para estos sistemas, la ciberseguridad nunca fue un asunto crucial; en lugar de ello, los retos eran crear un sistema de control industrial que fuera seguro, confiable y en tiempo real. La supervisión, la gestión y el control de procesos industriales se ha mejorado gracias a los sistemas SCADA modernos.[2]. Esta integración ha conducido a que se introduzcan muchas amenazas de seguridad en el ambiente. A causa de las tendencias tecnológicas, la seguridad en los sistemas SCADA se ha vuelto una cuestión fundamental. Se han presentado nuevas vulnerabilidades como resultado de la implementación de dispositivos comerciales estándar y protocolos de comunicación abiertos. Asimismo, la integración de redes corporativas ha vuelto a los sistemas vulnerables a ser explotados [3]. Por lo tanto, es imperativo garantizar una seguridad robusta y protegerlos contra posibles ciberamenazas.

Este trabajo está organizado de la siguiente manera; en la ***Sección* II** se describirá el protocolo DNP3, cuyo objetivo principal es dar una introducción abordando diferentes autores y sus trabajos acerca del protocolo de comunicación DNP3; en la ***Sección III*** se analizará la postura de seguridad en la estructura del protocolo dentro de las capas de **Enlace de Datos**, **Transporte** y **Capa de Aplicación**. en la ***Sección IV*** se describirán algunos ataques al protocolo DNP3; en la ***Sección V*** se hace la propuesta para el escenario de

generación de tráfico de comunicación en el protocolo DNP3 y la recolección de estos. se analizarán los datos recolectados con anterioridad y se hará una propuesta de solución haciendo uso de algoritmos de redes neuronales; en la ***Sección VI***, conclusiones. Como se ha expuesto en las secciones anteriores los protocolos destinados a aplicaciones industriales en el sector energético son intrínsecamente inseguros y presentan vulnerabilidades por diseño.

II. Protoolo DNP3

Los sistemas SCADA han evolucionado de estructuras centralizadas y propietarias a plataformas abiertas, inteligentes y resistentes. Esto ha sido posible gracias a la integración de tecnologías como el aprendizaje profundo, la inteligencia artificial y la computación en la nube para supervisar y automatizar áreas industriales clave. Estas innovaciones han hecho posible un monitoreo predictivo, un procesamiento avanzado de datos en tiempo real y el establecimiento de mecanismos sólidos para gobernanza y seguridad de información, elementos esenciales para enfrentar los desafíos de interoperabilidad y protección en los actuales entornos ciberfísicos e Industria 4.0.

El protocolo de red DNP3 se emplea extensamente en los sistemas SCADA (sistemas de control industrial) [4]. DNP3 tiene varias características que lo hacen apropiado para los sistemas de control industrial. Facilita la comunicación entre terminales remotas (UTR) y sus maestras, además de otros dispositivos de control; también recopila información de los sensores, dispositivos de campo y dispositivos de control, y los envía a la central. Estos atributos comprenden:

- Robustez: Está diseñado para funcionar en entornos hostiles con canales de comunicación con ancho de banda pequeño y ruidoso. Admite la detección y corrección de errores para la retransmisión de mensajes que garantizan una comunicación fiable.
- Flexibilidad: Está diseñado para ser flexible y adaptable a diversos dispositivos y aplicaciones.
- Escalabilidad: Está diseñado para soportar despliegues a gran escala con miles de dispositivos. Admite múltiples niveles de topología de red y puede utilizarse tanto en redes locales como en redes de área extensa; así como múltiples tipos de datos, velocidades y medios de comunicación

Es posible transmitir los mensajes del protocolo por medio de TCP/IP o conexiones seriales. Este estudio tiene en cuenta los mensajes de comunicación que se transmiten por medio del protocolo TCP/IP. En la actualidad, la mayor parte de los sistemas SCADA funcionan dentro de un ambiente de red TCP/IP. Si DNP3 opera mediante TCP/IP, la maestra es un cliente TCP que se conecta a varias remotas que funcionan como servidores TCP. Bajo estas circunstancias, se pueden asignar los mensajes a las capas superiores del modelo TCP/IP, que incluyen las capas de *transporte*, *aplicación* y *enlace de datos*. Cada capa tiene a su cargo una función concreta y brinda servicios a la capa que está por encima de ella, tal como se muestra en la Figura 2.

Como se ha señalado en los párrafos anteriores, es una necesidad primordial la detección rápida y precisa de amenazas, para poder realizar una mitigación temprana. Los enfoques de aprendizaje autónomo tienen un gran espectro de

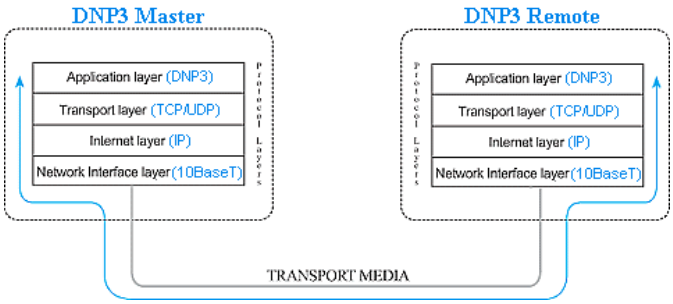


Figura 2. Capas protocolo DNP3.

implementación en la creación de herramientas que coadyuven en la detección de anomalías y clasificación de trafico de red, con el objetivo de agregar una capa de protección contra los ciberataques en sus procesos.

II-A. Trabajos relacionados

Es indiscutible que la cuestión de la seguridad en el sector energético es de extrema importancia, y las investigaciones en este campo se consideran de alta prioridad a causa del aumento del área objetivo y los potenciales daños que los ciberataques pueden causar. Es relevante señalar la creación de una línea base, inspirada en investigaciones semejantes que se han realizado en el campo de la ciberseguridad.

Las fuentes analizadas posibilitan la creación de un panorama extenso acerca de la digitalización de infraestructuras críticas, la incorporación de inteligencia computacional y los retos relacionados con la seguridad. Desde el área de los servicios básicos, [5] brinda un análisis exhaustivo acerca del desarrollo de los sistemas acuáticos, evidenciando la manera en que las decisiones del pasado y los marcos regulatorios han dado forma a los modelos presentes de gestión, resiliencia y evaluación de riesgos. Este contexto está relacionado con las demandas actuales de seguridad industrial que se encuentran en [6], donde se explican arquitecturas de confianza cero, normas como ISA/IEC 62443 y NIST CSF, así como ejemplos recientes de ciberataques que demuestran la creciente vulnerabilidad de ICS/OT frente a amenazas complejas. En [7] se trata la inclusión de aprendizaje automático en la salvaguarda de las redes energéticas, que verifica modelos supervisados muy precisos para identificar intrusiones fundamentadas en registros SCADA. Esta línea de integración tecnológica se desarrolla en Advanced management of photovoltaic systems using SCADA and machine learning techniques.pdf, en la que la combinación de información sobre operaciones, predicciones meteorológicas y redes neuronales posibilita la detección de irregularidades y la mejora del rendimiento en sistemas fotovoltaicos reales. De forma complementaria, En [8] amplía la mirada hacia industrias de proceso continuo, explicando la evolución arquitectónica de SCADA, la importancia de estándares como RAMI 4.0 e ISA-95, y el giro hacia Industria 5.0 con su énfasis en resiliencia y operación humanizada. Una visión más transversal aparece en [9], que examina el impacto de Industria 4.0 e Industria 5.0 en PyMEs, logística y modelos de negocio, destacando el papel de tecnologías como la computación en la nube, big data, ciberseguridad y técnicas avanzadas de inteligencia computacional, incluyendo lógica difusa, sistemas inmunes artificiales y redes profundas.

Finalmente, [10] sitúa a las redes inteligentes como elementos esenciales para la transición energética sostenible, articulando tecnologías como IoT, AMI, blockchain y sistemas de gestión de demanda, al tiempo que expone los desafíos de integración de energías renovables distribuidas y los requerimientos de seguridad que acompañan esta modernización.

- Radoglou-Grammatikis et al. [11] propone una técnica de detección, prevención de intrusiones y anomalías basadas en el flujo de la red, denominada *DIDEROT*, que utiliza aprendizaje automático supervisado y no supervisado, donde los flujos de red se clasifican con un modelo de árbol, según el conjunto de datos Rodofile, así como el comportamiento normal recolectado del tráfico de las subestaciones. Si el flujo se clasifica como no malicioso, el autocodificador se activa y busca posibles anomalías, si se descubre una actividad maliciosa o una anomalía, el módulo "Reponse" informa al controlador SDN para que abandone el flujo específico.
- Darwish et al. [12] presentan un trabajo sobre DNP3 y sus vulnerabilidades en el que destacan las debilidades de DNP3 e implementan dos escenarios de ataque contra el protocolo, en el primero tiene como objetivo interceptar y detener un mensaje no solicitado de la remota, y el segundo modifica el contenido de la carga útil TCP para manipular y redirigir el tráfico DNP3.
- Lin et al. [13] propone una técnica para proteger los sistemas SCADA basada en el analizador de tráfico, para obtener una visión concreta de los eventos relacionados. En adicción, se construye una política de validación de protocolo para verificar que el tráfico de red DNP3 se adhiere a los patrones de comunicación predefinidos. Irvene sostiene que las opciones de aprendizaje automático constituyen la mejor solución.

III. Estructura de protocolo DNP3.

Una unidad de datos de protocolo de aplicación *(APDU)* o fragmento se forma en la capa de aplicación combinando una unidad de datos de servicio de aplicación *(ASDU)*, que es un objeto empaquetado, con un bloque de información de control de protocolo de aplicación *(APCI)*, también conocido como cabecera de aplicación en la Figura 3. La capa de pseudo transporte divide el fragmento APDU creado en la capa de aplicación en segmentos de unidad de datos de protocolo de transporte (TPDU) de 16 bytes. Los envuelve con una cabecera de control de transporte de 8 bits (Pseudo-Transport Header). A continuación, la capa de enlace de datos añade una cabecera de enlace y una comprobación de redundancia cíclica (CRC) para la detección de errores y la corrección de cada una de las TPDU. Las capas de pseudo transporte y de enlace de datos se ocupan principalmente de la fragmentación de los mensajes de aplicación y de tareas adicionales como la gestión del flujo y el control de errores.

III-A. Capa de enlace de datos en DNP3.

La capa de enlace de datos facilita el flujo de mensajes manteniendo una conexión lógica fiable entre dispositivos. Una trama de capa de enlace de datos consta de una cabecera de 10-bytes y una porción de datos o ≪carga útil≫ que contiene datos transmitidos desde los niveles de pseudo transporte y aplicación. La longitud máxima de la sección de datos,

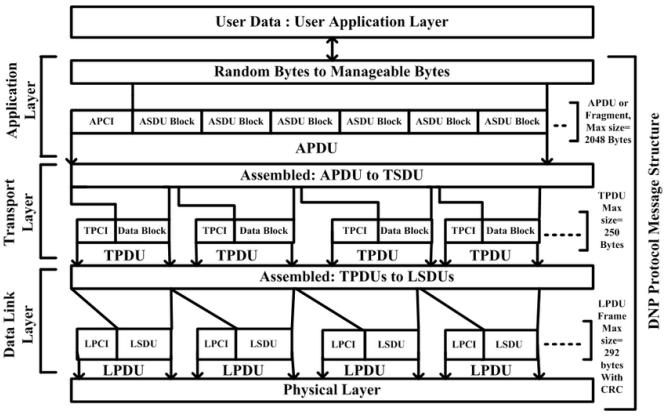


Figura 3. Estructura del protocolo DNP3.

incluidos los campos CRC de 16 bits por cada 16 bytes de datos, es de 282 bytes, con un límite de sección de datos de 250 bytes.

La dimensión agregada de una trama de transmisión de datos es de 292 bytes. El campo Start consta de un valor de 0x0564 de dos bytes, permita al destinatario identificar el inicio de la trama. Los bytes Start indican que ha llegado un paquete DNP3 y que debe ser tratado. El campo Longitud especifica el número de bytes restantes en la trama (sin incluir CRCs).En la Figura 4 podemos ver un ejemplo.

```
> Frame 163: 93 bytes on wire (744 bits), 93 bytes captured (744 bits)
> Ethernet II, Src: HonHaiPr_4d:b3:2f (00:22:68:4d:b3:2f), Dst: Dell_28:c9:54 (00
> Internet Protocol Version 4, Src: 192.168.0.198, Dst: 192.168.0.197
> Transmission Control Protocol, Src Port: 51955, Dst Port: 20000, Seq: 67, Ack:
v Distributed Network Protocol 3.0
  > Data Link Layer, Len: 20, From: 1, To: 10, DIR, PRM, Unconfirmed User Data
  > Transport Control: 0xc3, Final, First(FIR, FIN, Sequence 3)
  > Data Chunks
  > [1 DNP 3.0 AL Fragment (14 bytes): #163(14)]
  v Application Layer: (FIR, FIN, Sequence 2, Read)
    > Application Control: 0xc2, First, Final(FIR, FIN, Sequence 2)
      Function Code: Read (0x01)
    v READ Request Data Objects
      v Object(s): Class 1 Data (Obj:60, Var:02) (0x3c02)
        > Qualifier Field, Prefix: None, Range: No Range Field
          Number of Items: 0
      v Object(s): Class 2 Data (Obj:60, Var:03) (0x3c03)
        > Qualifier Field, Prefix: None, Range: No Range Field
          Number of Items: 0
      v Object(s): Class 3 Data (Obj:60, Var:04) (0x3c04)
        > Qualifier Field, Prefix: None, Range: No Range Field
          Number of Items: 0
      v Object(s): Class 0 Data (Obj:60, Var:01) (0x3c01)
        > Qualifier Field, Prefix: None, Range: No Range Field
          Number of Items: 0
```

Figura 4. Ejemplo de mensaje de solicitud DNP3

El campo Link Control de la cabecera del enlace de datos incluye los datos que origen del flujo de mensajes, establecen la secuencia y designan la función de la trama. Estos datos ayudan a determinar la clasificación del dispositivo como maestro o estación externa, a discernir el dispositivo que inició la comunicación y a presentar el estado del enlace lógico. Además, cabe destacar que el campo Link Control incluye un código de función de cuatro bits, el cual sirve para especificar el propósito del mensaje. Distintos conjuntos de códigos de función son utilizados por los mensajes que emanan de un maestro y los mensajes que se originan en dispositivos de estación externa. Los códigos de función maestros abarcan una variedad de tareas, como reiniciar un enlace remoto, reiniciar un proceso de usuario, solicitar el estado de una conexión y probar una función. Los códigos de

función para dispositivos de estación externa incluyen acuse de recibo positivo, rechazo de mensaje, estado de conexión y no disponibilidad de servicio de enlace. El campo Link Control comprende un par de banderas que sirven para la sincronización y control del flujo de comunicación. La cabecera del enlace de datos comprende una dirección de destino de 16 bits que sirve para identificar al destinatario previsto, que puede abarcar una dirección de difusión de 0xFFFF. Además, la dirección de origen de 16 bits se incluye en la cabecera para identificar al emisor de la comunicación.En la Figura 5 podemos ver dun ejemplo.

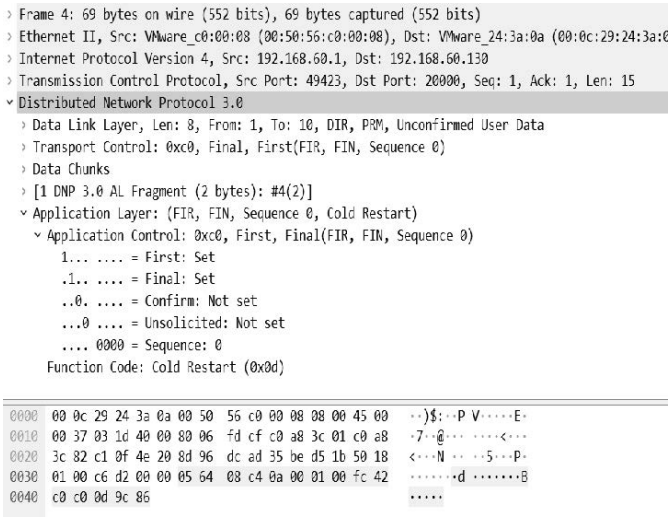


Figura 5. Ejemplo de mensaje de solicitud DNP3

III-B. Capa de transporte en DNP3

La capa de transporte gestiona la fragmentación de mensajes y el reensamblaje son gestionados por la capa de transporte en el protocolo DNP3. Como se ha indicado anteriormente, el protocolo permite la transmisión de mensajes de aplicación que superan el tamaño de una única trama de conexión de datos mediante la utilización de múltiples tramas. La capa de transporte se compone de un byte extra que contiene las banderas *FIR* y *FIN*; estas identifican las tramas inicial y final de un mensaje fragmentado respectivamente. El número de secuencia muestra un incremento con cada trama subsecuente y se emplea con el fin de reconstruir mensajes destinados a la aplicación. La información obtenida a través de la secuenciación puede ayudar en la identificación de tramas perdidas.En la Figura 6 podemos ver de manera gráfica su estructura.

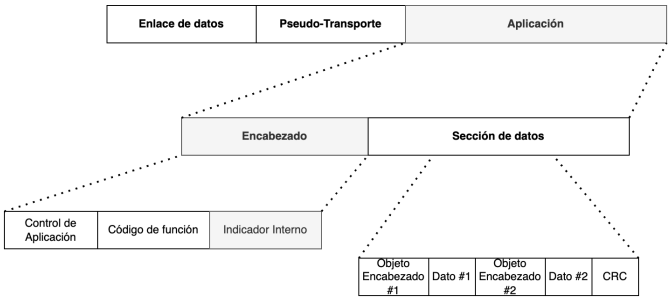


Figura 6. Capa de transporte de datos DNP3

III-C. Capa de aplicación en DNP3

La capa de aplicación ordena los mensajes de solicitud y de respuesta, esboza las distintas funciones del maestro y de las remotas. Un mensaje de solicitud del maestro esta diseñado para enviar a la remota una instrucción para realizar una tarea específica, recoger y proporcionar datos, o sincronizar su reloj interno. En el contexto de los protocolos de comunicación, es un requisito que sólo el dispositivo maestro esté autorizado a iniciar mensajes de petición, mientras que a las remotas se les permite transmitir comunicaciones solicitadas y no solicitadas. En el contexto de las redes informáticas, es habitual que la capa de aplicación divida los mensajes que superen el tamaño máximo de fragmento, que está predeterminado por la capacidad del búfer del destinatario. El tamaño de un fragmento de mensaje típico varía entre 2048 y 4096 bytes. Como se muestra en la Figura 7.

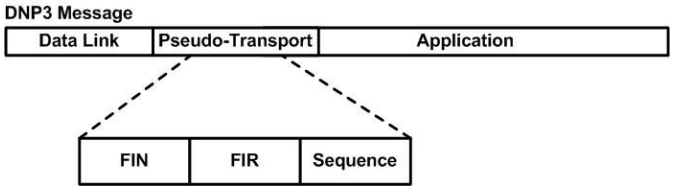


Figura 7. Capa de enlace de datos DNP3

El campo denominado ≪*Código de función*≫ sirve para transmitir la función prevista de un mensaje determinado por parentesis ”( )”. El dominio mencionado anteriormente se aplica tanto a las solicitudes como a las respuestas, aunque con diferentes funcionalidades disponibles en la función del tipo de mensaje. La tabla I describe los distintos tipos de mensajes de respuesta, que incluyen confirmación, respuesta y respuesta no solicitada. Los mensajes de solicitud de se clasifican en seis grupos basados en los 23 códigos de función *(FC)* designados.

Tabla I
CÓDIGOS DE LAS FUNCIONES DEL PROTOCOLO DNP3.

Tipo	FC	Significado
Request	0x00	Confirm
Request	0x01	Read
Request	0x02	Write
Request	0x03	Select
Request	0x04	Operate
Request	0x05	Dir Operate
Request	0x06	Dir Operate
Request	0x07	Freeze
Request	0x08	Freeze - No resp
Request	0x09	Freeze clear
Request	0x0A	Freeze clear - No resp
Request	0x0B	Freeze at time
Request	0x0C	Freeze at time - No resp
Request	0x0D	Cold restart
Request	0x0E	Warm restart
Request	0x11	Start application
Request	0x12	Stop application
Request	0x13	Save configuration
Request	0x14	Enable unsolicited

El sistema incluye seis funciones diferentes: (I) funciones de control, (II) funciones de configuración, (III) funciones de sincronización temporal, (IV) funciones de congelación y (V) funciones para controlar aplicaciones.

Sus instrucciones son las siguientes: La solicitud de reinicio en frío suele enviarse en un solo paquete. Por lo tanto, FIR y FIN se establecen en uno, las cabeceras contienen un campo de dos bytes Internal Indications *(IIN)* que transmite valiosa información sobre la unidad de estación externa al maestro. Cada bit en el campo *(IIN)* tiene un significado distinto que se cambia con cada mensaje de respuesta. Los ejemplos de código *(IIN)* incluyen sincronización de tiempo necesaria, reinicio del dispositivo, argumentos incorrectos, código de función no implementado y objetos solicitados desconocidos. El indicador Interno de la capa de aplicación del protocolo DNP3 es un campo de 16 bits, cada uno de los cuales tiene un significado específico. Por ejemplo, el bit número 7 del LSB de indica que el RTU se ha reiniciado desde un reinicio completo del sistema.

La capa de aplicación DNP3 está constituida por objetos de datos que se utilizan para transmitir representaciones de datos codificadas. Se han definido múltiples objetos de datos para posibilitar la eficaz comunicación de comandos y datos entre dispositivos que funcionan en varias plataformas. Los objetos de datos, tales como contadores, salidas analógicas, entradas binarias y salidas binarias, pueden considerarse como ejemplos de variables discretas o continuas dentro de un sistema.

IV. Perspectiva de seguridad del protocolo DNP3.

Con el descubrimiento e identificación de *Stuxnet*, ha llegado el amanecer de una nueva era de ciberguerra. Stuxnet está considerado el primer ciberataque del que se tiene constancia que estuvo dirigido contra sistemas físicos. Desde entonces, varios ciberataques y problemas de seguridad en el mundo real han asolado las redes SCADA. En la Figura 8 podemos ver parte de la arquitectura.

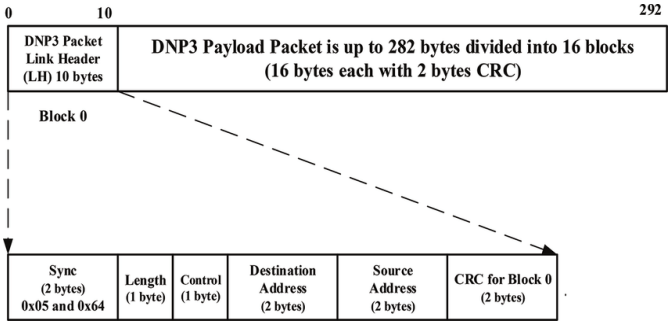


Figura 8. Arquitectura de seguridad del protocolo DNP3

Esos ataques muestran el impacto que tienen las vulnerabilidades en infraestructuras esenciales y las severas consecuencias de los ciberataques en las redes SCADA, lo cual se traduce en importantes pérdidas económicas. Ilustraciones de este tipo de sucesos son el ciberataque que tuvo lugar en el año 2015 y provocó un apagón en Ucrania.

- Ataque a la red eléctrica en Ucrania: En diciembre de de 2015, los atacantes utilizaron una combinación de correos electrónicos de spearphishing y ataques de protocolo DNP3 para obtener acceso a los sistemas de control de la red eléctrica ucraniana. Los atacantes lograron hacerse con el control de la red e interrumpir el suministro eléctrico a unas 230.000 personas durante varias horas.
- Dragonfly 2.0: En 2017, un grupo de atacantes conocido como *Dragonfly 2.0* llevó a cabo una serie de ataques contra organizaciones del sector energético en Estados Unidos y Europa. Los ataques aprovecharon vulnerabilidades en el uso del protocolo DNP3 entre otras tácticas, para obtener acceso a los sistemas de control de estas organizaciones.
- Malware Triton: En 2017, un grupo de atacantes utilizó una combinación de ataques de protocolo DNP3 y malware conocido como *Triton* para atacar los sistemas de seguridad de una planta petroquímica en Arabia Saudí. Los atacantes fueron capaces de tomar el control de estos sistemas y potencialmente causar daños materiales de gran magnitud.

IV-A. Vulnerabilidades del protocolo DNP3.

La implementación actual de sistemas SCADA hace uso del protocolo DNP3, el cual carece de mecanismos de seguridad integrado. Por lo tanto, estos sistemas presentan vulnerabilidades y son susceptibles a una variedad de ciberataques. Por ello, se describirán algunos ataques aprovechando las vulnerabilidades del protocolo, ya que los cuatro principales tipos de amenazas son: spoofing, modificación, reenvío, inyección de comandos y ataques de hombre en medio.

- Ataque al mensaje de reinicio en frío DNP3: Aprovecha un fallo en el protocolo DNP3, el ataque consiste en enviar al dispositivo maestro un mensaje cuidadosamente preparado que le pide que inicie el reinicio en frío del sistema, después del reinicio, se pierden todas sus configuraciones.
- Ataque al mensaje de reinicio en caliente DNP3: Al igual que el mensaje de reinicio en frío, el mensaje de reinicio en caliente interfiere en el funcionamiento del sistema SCADA aprovechándose de un fallo en el protocolo DNP3. Provocando un reinicio parcial. Esto provoca un vaciado de su memoria interna (buffer) pudiendo generar graves repercusiones, aunque mantiene su información de configuración y conexiones a las remotas.
- Ataque de mensajes NO solicitados DNP3: El protocolo de comunicación DNP3 facilita la transmisión de mensajes no solicitados desde las remotas a la maestra, permitiendo así una información rápida y actualizada sobre el estado operativo de los procesos, los datos pueden abarcar detalles relativos al estado de los equipos, variables de proceso, y alarmas.
- Ataque de petición DNP3: Implica la interceptación ilícita y posterior repetición de transmisiones auténticas de datos con la intención de ejecutar acciones malintencionadas.
- Ataque por inyección de mensajes DNP3: La inyección de mensajes DNP3 es una forma de ciberataque mediante la cual un adversario introduce mensajes falsificados o malintencionados en el canal de comunicación.

V. Propuesta para análisis de tráfico e identificación de patrones anómalos.

El análisis de tráfico e identificación de patrones anómalos en sistemas SCADA son aspectos críticos para garantizar

la seguridad e integridad de estos sistemas, los cuales son utilizados para el control dentro de las infraestructuras críticas en el sector eléctrico. Las tareas de análisis e identificación son de suma importancia, originadas de las vulnerabilidades y amenazas que surgen de las interconexiones con otros sistemas y entornos. Las conexiones externas a otros sistemas son fácilmente visibles incluso a partir de diagramas de red simplificados.

Sera necesario recopilar, analizar y monitorear continuamente el tráfico de la red, las fases son las siguientes:

- Monitoreo de los dispositivos y los enlaces en el entorno SCADA.
- Recopilación de tráfico NP3.
- Propuesta de algoritmos de análisis.
- Análisis de los datos
- Detección de patrones anómalos.
- Propuestas de controles para mitigación de riesgo.

De manera evidente, la ejecución con éxito de los ataques mencionados en un entorno SCADA puede provocar la interrupción de los servicios, alteración de los mensajes y evitar la entrega de notificaciones críticas. Por ello, para evitar las consecuencias de los ciberataques, es necesarios contar con soluciones que detecten con rapidez y precisión los intentos maliciosos. En el trabajo de investigación de se aborda una propuesta para tratar de identificar los patrones anómalos aprovechando los flujos de red con trafico DNP3, se propondrá el análisis de los flujos mediante *CICFlowMeter* disponible en el siguiente enlace https://github.com/ahlashkari/CICFlowMeter y simulación con *openDNP3* disponible en el siguiente enlace https://github.com/dnp3/opendnp3 en capas para operar en colaboración y realizar pruebas con los algoritmos propuesto para la identificación de patrones anómalos, de esta manera poder identificar a tiempo los ataques a la infraestructura critica, nuestro primer componente de encargará de analizar el tráfico de red a nivel de paquete generados con nuestro segundo componente. Estos flujos al ser generados en estas dos herramientas; generara la emulación de un tráfico, para observar y analizar los patrones de comunicación en modo aprendizaje y sea capaz de identificar y clasificar los flujos válidos y los que presenten anomalías, para estas pruebas se realizaran con envenenamiento ARP. Para posteriormente realizar un etiquetado, dando lugar a la creación de un nuevo archivo y de esta manera tener la capacidad de crear alertas que podrían identificar un posible ciberataque.

VI. Conclusiones.

DNP3 es un protocolo de comunicación SCADA que se ha estandarizado para garantizar la compatibilidad e interoperabilidad entre dispositivos. Se examino el protocolo DNP3, así como, sus limitaciones de seguridad y sus posibles ataques aprovechando los huecos de seguridad inherentes al propio protocolo de comunicación. El protocolo ha sido objeto de numerosos ciberataques debido a sus limitaciones y vulnerabilidades.

Por ello, ha sido el principal motivo de preocupación de los expertos en seguridad. En consecuencia, la insuficiencia para implantar protocolos de protección adecuados y asegurar estos sistemas contra los ciberataques y ataques físicos en constante evolución pueden materializarse en resultados catastróficos. Prevenir los ciberataques contra los sistemas SCADA es un requisito urgente y crucial para garantizar su protección. Esto puede lograrse mejorando las defensas y medidas de protección de estos sistemas. De manera puntual, el trabajo de investigación creara un modelo que sea capaz de identificar patrones anómalos. Utilizando flujos de red que contengan atributos específicos del protocolo para su análisis y presentar la propuesta para una solución que aporte robustez al protocolo DNP3 contra ataques cibernéticos al ser estos implementados y utilizados en las infraestructuras críticas.

Agradecimientos

Este trabajo se ha realizado en el marco de los fondos del Plan de Recuperación, Transformacionales y Resiliencia, financiados por la Unión Europea (Next Generation), a través de la Cátedra "Ciberseguridad para la Innovación y la Protección y la Protección Digital" (Cátedra INCIBE-UCM). Asimismo, este trabajo ha sido apoyado por la Comunidad Autónoma de Madrid, Proyecto CIRMA-CM (TEC-2024/COM-404). No obstante, las opiniones expresadas son responsabilidad exclusiva de los autores y no reflejan necesariamente las de INCIBE, la Unión Europea o la Comisión Europea-UE. Ni INCIBE, ni la Unión Europea, ni la Comisión Europea pueden ser consideradas responsables de las mismas.

Referencias

[1] K. Tsantikidou and N. Sklavos, "Threats, attacks, and cryptography frameworks of cybersecurity in critical infrastructures," *Cryptography*, vol. 8, no. 1, 2024.

[2] V. Kelli *et al.*, "Attacking and defending dnp3 ics/scada systems," in *2022 18th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, May 2022, pp. 183–190.

[3] R. Pandit, D. Astolfi, J. Hong, D. Infield, and M. Santos, "Scada data for wind turbine data-driven condition/performance monitoring: A review on state-of-art, challenges and future trends," *Wind Engineering*, vol. 47, no. 2, pp. 422–441, September 2023.

[4] M. O. Qays, M. M. Ahmed, M. A. P. Mahmud, A. Abu-Siada, S. M. Muyeen, M. L. Hossain, F. Yasmin, and M. M. Rahman, "Monitoring of renewable energy systems by iot-aided scada system," *Energy Science & Engineering*, vol. 10, no. 6, pp. 1874–1885, June 2022.

[5] P. Bhandari, D. Creighton, J. Gong, C. Boyle, and K. M. Y. Law, "Evolution of cyber-physical-human water systems: Challenges and gaps," *Technological Forecasting and Social Change*, vol. 191, p. 122540, 2023.

[6] S. Kumar and H. Vardhan, "Cyber security of ot networks: A tutorial and overview," *arXiv preprint arXiv:2502.14017*, 2025.

[7] F. Martinelli, F. Mercaldo, and A. Santone, "A method for intrusion detection in smart grid," *Procedia Computer Science*, vol. 207, pp. 327–334, 2022.

[8] M. Sverko, T. G. Grbac, and M. Mikuc, "Scada systems with focus on continuous manufacturing and steel industry: A survey on architectures, standards, challenges and industry 5.0," *IEEE access*, vol. 10, pp. 109 395–109 430, 2022.

[9] D. T. Matt, V. Modrák, and H. Zsifkovits, *Implementing Industry 4.0 in SMEs: Concepts, Examples and Applications*. Springer Nature, 2021.

[10] M. J. B. Kabeyi and O. A. Olanrewaju, "Smart grid technologies and application in the sustainable energy transition: a review," *International Journal of Sustainable Energy*, vol. 42, no. 1, pp. 685–758, 2023.

[11] P. R. Grammatikis, P. Sarigiannidis, G. Efstathopoulus, P. Karipidis, and A. Sarigiannidis, "Diderot: an intrusion detection and prevention system for dnp3-based scada systems," in *Proceedings of the 15th International Conference on Availability, Reliability and Security*, August 2020, pp. 1–8.

[12] I. Darwish, O. Celebi, T. Saadawi, and J. Soryal, "Smart grid dnp3 vulnerability analysis and experimentation," in *2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing*, November 2015.

[13] H. Lin, A. Slagell, C. D. Martino, and R. Iyer, "Adapting bro into scada: Building a specification-based intrusion detection system for the dnp3 protocol," in *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop*, January 2013.

Optimización del test de primalidad para la generación de claves de firmas en anillo

Álvaro Larrey-Gómez
Universidad de Zaragoza
Escuela Universit. Politécnica de Teruel
alarrey@unizar.es

José Luis Salazar Riaño
Universidad de Zaragoza
Escuela de Ingeniería y Arquitectura
jsalazar@unizar.es

Guillermo Azuara Guillén
Universidad de Zaragoza
Escuela Universit. Politécnica de Teruel
gazuara@unizar.es

***Resumen*—La generación eficiente de claves es un pilar para la criptografía aplicada, especialmente en sistemas de voto electrónico que requieren seguridad y agilidad. Este trabajo aborda la optimización de la generación de claves para firmas en anillo con una estructura matemática particular, cuyo coste computacional es un obstáculo para su uso en dispositivos con recursos limitados. Proponemos un enfoque de optimización que explota dicha estructura, introduciendo dos mejoras: una criba de divisibilidad acelerada, denominada "criba de veto", y la sustitución de tests probabilísticos por el test determinista de Pocklington-Lehmer. La aplicación de estas técnicas a medida logra una aceleración sustancial en el proceso de generación de claves, a la vez que refuerza la seguridad al eliminar la probabilidad de error de los tests probabilísticos.**

***Index Terms*—generación de claves, optimización, primos estructurados, test de primalidad, criba de divisibilidad, firmas en anillo, test de Pocklington-Lehmer**

I. Introducción

El principio de anonimato constituye un elemento esencial en todo proceso electoral, ya que garantiza la libertad del votante y la integridad democrática del resultado [1]. Para mantener la confidencialidad del sufragio, la investigación en criptografía aplicada ha desarrollado diversos mecanismos que buscan asegurar el anonimato y la verificabilidad simultáneamente [2]. Entre los más destacados se encuentran las firmas ciegas [3], los servidores mezcladores [4], el cifrado homomórfico [5] y las firmas en anillo [6].

Otro requisito más actual y restrictivo de las votaciones electrónicas es la inmediatez. Esto implica que a los sistemas criptográficos anteriores habría que unir la característica de minimizar el tiempo de proceso y la infraestructura de votación para el usuario. Con este último requisito, las firmas en anillo se presentan como una opción más que aceptable para poder ser implementada incluso en dispositivos con restricciones computacionales como pueden ser los terminales móviles. Para aligerar la infraestructura, deberíamos exigir que la firma en anillo sea también corta, es decir, que no crezca de forma lineal con el número de votantes. En [7] se consiguen unas firmas en anillo con tamaño lineal y que no requieren de una entidad de confianza organizadora (espontáneas).

Sin embargo, estas firmas cortas tienen como requisito unas claves con unas características concretas que hace difícil su establecimiento en un entorno con restricciones computacionales. En [8] se proponen dos estrategias para la optimización de la búsqueda de dichas claves, sin embargo, no aprovechan la posibilidad que ofrece la estructura de las mismas para, al validarlas, reducir las operaciones a realizar en los test de primalidad.

En este artículo, nos centramos en la optimización de generación de una de sus claves: el número primo $e = 2e_1e_2 + 1$, donde e_1 y e_2 son también números primos. Para comprobar la primalidad de e, habitualmente se aplica una criba con el test de divisibilidad con una cantidad fija de primos pequeños; y una vez pasado este test, se le aplica algún test de divisibilidad probabilístico. Nosotros proponemos explotar la estructura única e, con optimizaciones a medida en el cribado y la incorporación del test de Pocklington-Lehmer, logrando una aceleración sustancial en la generación de claves y mejorando así, drásticamente la viabilidad del sistema para su uso en dispositivos de usuario.

En el siguiente punto exponemos los trabajos relacionados con el que presentamos. Seguidamente describimos nuestras propuestas de mejora, tanto en la criba inicial, como en el test de primalidad a emplear en la generación de la clave e. A continuación analizamos los resultados obtenidos en los que basamos las conclusiones de la sección final.

II. Antecedentes y trabajos relacionados

De acuerdo con lo establecido en [7], el usuario ha obtenido los parámetros asociados a una votación: n, λ y μ.

El parámetro λ mide la seguridad del sistema mediante el tamaño de n, definido como:

$$n = pq = (2p' + 1)(2q' + 1), \tag{1}$$

donde p, q, p', q' son números primos y se cumple que $p', q' > 2^{\lambda/2}$. Además, se satisfacen las desigualdades

$$|e_1 - 2^l|, |e_2 - 2^l| < 2^\mu \tag{2}$$

donde los parámetros λ y μ se eligen de forma que se eviten ataques de coalición [9]. Para ello, basta con tomar $l \approx \lambda/2$ y un valor de μ suficientemente pequeño, según lo indicado en [7]. En nuestro caso, adoptamos la elección $\mu = l - 2$, siguiendo la recomendación de [10].

Si bien la generación de n a partir de p, q, p' y q' no es un proceso inmediato, éste corresponde a un caso particular de la generación de *primos robustos* [11], procedimiento que se encuentra suficientemente estandarizado y queda fuera del alcance del presente artículo.

Definimos el algoritmo de generación de claves de usuario como

$$KeyGen(n, \lambda, \mu) = (e_1, e_2, e = 2e_1e_2 + 1, e < n). \tag{3}$$

donde los tres valores e_1, e_2 y e son números primos. Cada par (e_1, e_2) constituye una clave privada utilizada para la firma

en anillo, mientras que e representa la clave pública asociada a dichos valores.

Bajo esta construcción, puede verificarse fácilmente que $0 \neq e_1 \neq e_2 \neq 0 \bmod 3$. Sin pérdida de generalidad, se cumple que $e_1 = 1 \bmod 3$ y $e_2 = 2 \bmod 3$, y los valores e_1, e_2 pertenecen al intervalo

$$e_1, e_2 \in (2^{\lambda/2} - 2^{\mu}, 2^{\lambda/2} + 2^{\mu}). \quad (4)$$

La estrategia empleada para la búsqueda de estas claves depende de la capacidad computacional disponible, tanto en términos de tiempo de procesado como de recursos de memoria. En [7] se optimizan las estrategias para ambos recursos (tiempo y memoria), dejando claro que la complejidad del algoritmo depende de la complejidad del test de primalidad que se emplea y de la distribución probabilística de los números primos. Dado que la distribución es inamovible, y sobre todo, probabilística, optimizar el método de búsqueda resulta difícil de abordar, por lo que nuestra propuesta se centra en optimizar los test de primalidad.

La determinación de la primalidad de un número entero *n* es un problema fundamental en teoría de números y criptografía. Existen diversos algoritmos para este propósito, que pueden clasificarse en dos grandes categorías: deterministas, que garantizan una respuesta exacta, y probabilísticos, que ofrecen un resultado con un margen de error estadísticamente controlado.

Los métodos deterministas establecen de forma inequívoca si un número *n* es primo o compuesto. Aunque son exactos, suelen tener una complejidad computacional más elevada. Los más conocidos son el test de divisibilidad directa o cribado, (probando si los primos menores que $\sqrt(n)$ dividen a *n*), el método Pocklington-Lehmer (que es abordable cuando se conoce parte de la factorización de $n - 1$) [12] y el método AKS [13][14] que tiene complejidad computacional polinómica, pero con una constante tan alta multiplicando en la complejidad que lo hace poco eficiente en la práctica.

Los test probabilísticos no determinan con certeza absoluta la primalidad, pero permiten controlar la probabilidad de error, siendo extremadamente eficientes para números grandes. Los más utilizados para aplicaciones criptográficas son los test de Fermat[15][16], Miller-Rabin [17][18], Solovay-Strassen [19] y el de Baillie-PSW [20][21].

Concretamente, cuando se procede a la implementación de un test de primalidad se suele combinar el método de divisibilidad directa, comprobando la divisibilidad del número candidato entre los primos más bajos (hasta una cota determinada) y luego se efectúa el test de Miller-Rabin por ofrecer el mejor equilibrio entre eficiencia y fiabilidad. Esta combinación de una criba de divisibilidad simple seguida por el test de Miller-Rabin estándar constituye el enfoque de referencia para evaluar el rendimiento de nuestras optimizaciones.

En el próximo punto exponemos nuestra propuesta de mejora, precisamente para esta metodologia aprovechando las características especiales que tiene el candidato a primo, $e = 2e_1e_2 + 1$.

III. Optimización del Test de Primalidad

Nuestra metodología se fundamenta en el aprovechamiento de la estructura única de las claves. Como se ha mencionado, la clave pública e se define como $e = 2e_1e_2 + 1$, donde tanto e como las claves privadas e_1 y e_2 deben ser números primos.

Nuestra propuesta parte de las estrategias expuestas en [8]. En ella se estableció que, para que e no fuera divisible por 3, se debe cumplir que $e_1 \equiv 1 \pmod 3$ y $e_2 \equiv 2 \pmod 3$ (o viceversa), permitiendo la creación de pares de primos a partir de estos dos conjuntos disjuntos de primos. Las estrategias definidas en [8] se basan en encontrar primero los primos pequeños (e_1, e_2) y luego comprobar la primalidad del candidato e.

Es importante señalar que no hemos buscado acelerar la generación de los primos e_1 y e_2 en sí mismos. Aunque este paso no es computacionalmente gratuito, los tests de primalidad para e_1 y e_2 son considerablemente más ligeros que la comprobación del primo resultante e. Integrar la generación de primos e_1 y e_2 en la optimización se deja como posible línea de trabajo futuro.

Sin embargo, dado que para la fase de cribado vamos a calcular ciertos valores auxiliares para cada par (e_1, e_2), procedermos a definir el algoritmo de generación de los mismos, distribuyéndolos en dos conjuntos disjuntos: P_1 y P_2.

Algorithm 1 Generación de conjuntos de primos P_1 y P_2

Require: Conjunto de primos pequeños Q, Entero Max
Ensure: Conjuntos de Max primos distribuidos en los conjuntos P_1 y P_2

1: $P_1 = \varnothing, P_2 = \varnothing, i = 0$
2: **while** $i < Max$ **do**
3: p=GENERAR PRIMO PROBABILISTICO(λ, μ)
4: **end while**
5: $j = p \pmod 3$
6: $x[0] = p$
7: **if** $j = 1$ **then**
8: **for** $k = 1$ **to** $|Q|$ **do**
9: $x[k] = -1/(2p) \pmod Q[k]$
10: **end for**
11: **else**
12: **for** $k = 1$ **to** $|Q|$ **do**
13: $x[k] = p \pmod Q[k]$
14: **end for**
15: **end if**
16: $P_j = P_j \cup \{x\}$
17: **return** (P_1, P_2)

III-A. Cribado Eficiente de Candidatos

El primer paso en cualquier test de primalidad eficiente consiste en una criba de divisibilidad, que descarte rápidamente los candidatos que son múltiplos de primos pequeños. La aproximación estándar sería comprobar si $e \equiv 0 \pmod q$ para un conjunto de primos pequeños $q \in Q$.

La condición $e = 2e_1e_2 + 1 \equiv 0 \pmod q$ es equivalente a $e_1 \equiv -1/(2e_2) \pmod q$. Esta equivalencia nos permite transformar una operación sobre el número grande e en una simple comprobación de congruencia sobre sus componentes más pequeños, e_1 y e_2.

A esta técnica la hemos denominado criba de veto. La idea es que, para cada primo pequeño q de nuestro conjunto

de criba Q, un primo e_1 "veta" a ciertos primos e_2 si la combinación (e_1, e_2) hace que el candidato e resultante sea divisible por ese q. Si una pareja (e_1, e_2) es vetada por cualquier primo q de nuestro conjunto de criba Q, se descarta inmediatamente sin necesidad de calcular e.

Para implementar la criba de veto de forma eficiente, precalculamos y almacenamos valores de veto, para cada primo que se va a usar en el proceso de emparejado. Cuando se considera un nuevo primo p del conjunto P_1 (donde $p \equiv 1 \pmod 3$), calculamos y guardamos el conjunto de sus residuos de veto, $v_{1q} = -1/(2p) \pmod q$, para cada $q \in Q$. De forma análoga, para un primo p' del conjunto P_2 (donde $p' \equiv 2 \pmod 3$), almacenamos sus residuos directos, $v_{2q} = p' \pmod q$. Así, la comprobación de la criba para la pareja (p, p') se reduce a verificar que $v_{1q} \neq v_{2q}$ para todo $q \in Q$. Este método promete reducir el coste computacional ya que todas las operaciones se realizan sobre números mucho más pequeños que el candidato e.

III-B. Test Determinista: Criterio de Pocklington-Lehmer

Para explicar la modificación hecha a este test para adaptarlo al tipo de número primo que estamos buscando recordemos brevemente en qué consiste.

Si $N > 1$ y se conoce $F|(N-1)$ con $F > \sqrt{N}$, basta hallar un a tal que:

$$a^{N-1} = 1 \text{ mód } N \tag{5}$$

$$mcd\Big(a^{(N-1)/q} - 1,\, N\Big) = 1 \tag{6}$$

para todo primo $q|F$. En ese caso, N es primo.

La mayor complicación de este método es calcular la factorización de $N-1$. Pero en nuestro caso la conocemos porque $e - 1 = 2e_1e_2$. El siguiente punto de dificultad es encontrar el testigo a, si es que existe. Veamos cuál sería la probabilidad de escoger un candidato aleatorio a y que éste pasara las prubas de Pocklington-Lehmer:

Sea N primo y $F|(N-1)$ con $N - 1 = F \cdot R$,

$$F = \prod_{q|F} q, \tag{7}$$

donde los q son primos conocidos. La probabilidad de que un candidato a elegido uniformemente al azar cumpla la *condición (6)* del test es

$$p_{\text{éxito}} = \prod_{q|F} \left(1 - \frac{1}{q}\right) \tag{8}$$

Esto es así porque para que $mcd\big(a^{(N-1)/q} - 1,\, N\big) \neq 1$ debería ocurrir que a fuera un múltiplo de cualquiera de los divisores q de F, que ocurre con una probabilidad de $1 - \frac{1}{q}$.

Si los candidatos a se eligen de manera aleatoria e independiente, el número esperado de intentos hasta encontrar un testigo válido sigue una distribución geométrica: $\mathbb{E}[k] = \frac{1}{p_{\text{éxito}}}$. Si queremos que la probabilidad de no encontrar un testigo tras k intentos sea como máximo ε, basta con resolver

$$(1 - p_{\text{éxito}})^k \leq \varepsilon \quad \Longrightarrow \quad k \geq \frac{\ln \varepsilon}{\ln(1 - p_{\text{éxito}})}. \tag{9}$$

Si tomamos $\varepsilon = 2^{-20}$ (cota de error habitual en el método Miller-Rabin) y que e_i tiene 250 bits, sustituyendo en la *fórmula (9)* obtenemos que:

$$k \geq \frac{ln 2^{-20}}{\ln 2^{-251}} = \frac{-20}{-251} < 1 \tag{10}$$

Tras estos cálculos, y tomando $F = 2e_i$, o $F = max\{e_1, e_2\}$ (dependiendo de qué manera $F > \sqrt{e}$), podríamos concluir que haciendo un test con un único testigo la probabilidad de obtener un falso negativo (rechazar un número primo en el test) es despreciable y por supuesto, mucho menor de la que el test de Miller-Rabin nos dé un falso positivo (que un número compuesto pase la prueba de primalidad) en las habituales 10 iteraciones con testigos diferentes y aleatorios.

De esta manera el algoritmo del test completo de primalidad para e que proponemos es el siguiente.

Algorithm 2 Criba de veto y test de primalidad de Pocklington-Lehmer sobre e

Require: Conjunto de primos pequeños Q, Primos e_1 y e_2
Ensure: Primalidad de $2e_1e_2 + 1$

```
for k = 1 to |Q| do
    if e1[k] = e2[k] then
        return FALSE
    end if
end for
e = 2e1e2 + 1
a ←R {1, ..., e}
if a^e ≠ 1 (mód e) then
    return FALSE
else
    F = máx{e1, e2}
    if mcd(a^((e−1)/F) − 1, e) ≠ 1 then
        return FALSE
    end if
    if F < √e then
        if mcd(a^((e−1)/2) − 1, e) ≠ 1 then
            return FALSE
        end if
    end if
end if
return TRUE
```

IV. Resultados y Evaluación

La evaluación de nuestras propuestas se estructuró en dos fases principales. La primera fase consistió en pruebas comparativas para analizar el rendimiento de cada componente de forma aislada. La segunda fase se centró en pruebas aplicadas a nuestro uso específico, simulando un escenario real donde el objetivo es encontrar una única pareja de claves válida lo más rápido posible.

IV-A. Análisis de las Cribas

Para evaluar el rendimiento de las cribas se optó por la siguiente metodología. Para un conjunto predefinido de tamaños de bit (512, 640, etc.), se generaron dos colecciones de 150 primos cada una: $e_1 \equiv 1 \pmod 3$ y $e_2 \equiv 2 \pmod 3$. Posteriormente, se midió el tiempo de ejecución de cada

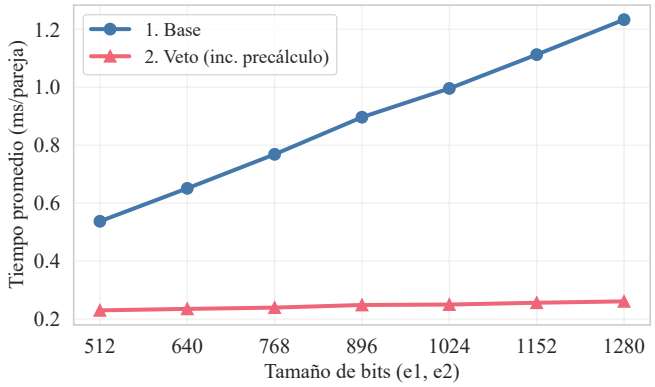


Figura 1. Comparativa del tiempo promedio de ejecución de las técnicas de cribado.

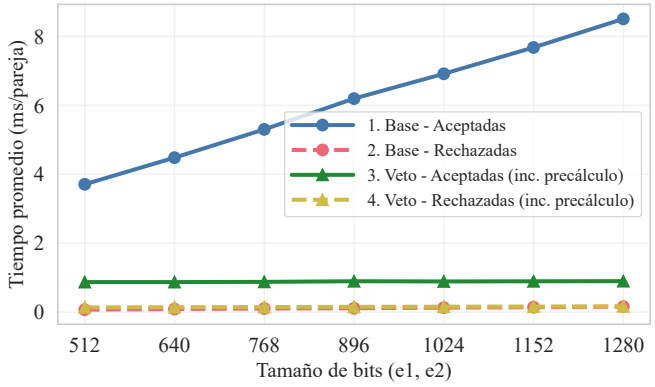


Figura 2. Comparativa desglosada entre criba base y criba veto al procesar primos y compuestos.

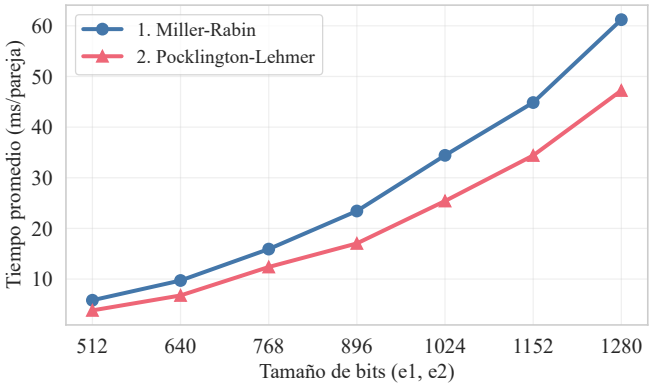


Figura 3. Comparativa del tiempo promedio de ejecución de diferentes tests de primalidad.

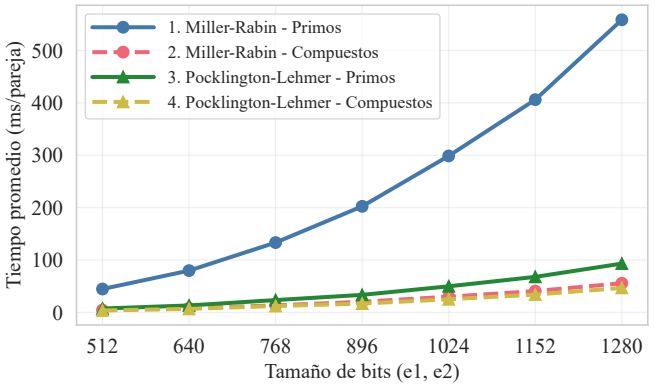


Figura 4. Comparativa desglosada entre Miller-Rabin y Pocklington-Lehmer al procesar primos y compuestos.

criba sobre el producto cartesiano de ambos conjuntos, lo que equivale a 22,500 comprobaciones por cada tamaño de bit.

La Figura 1 resume los resultados, mostrando que el tiempo del método base crece linealmente con el tamaño de los primos, mientras que la criba de veto mantiene un tiempo de ejecución prácticamente constante y muy inferior. El coste de precómputo de la criba de veto para 300 primos de 512 bits fue de 2.751,45 ms, y para 300 primos de 1280 bits, de 3.564,22 ms.

La Figura 2 desglosa el tiempo de procesado para parejas aceptadas y rechazadas. Mientras que el coste de rechazar un par es mínimo en ambos casos, debido a que la gran mayoría de los candidatos son divisibles por 3, 5 y 7, el método base muestra un coste drástico y creciente para los pares aceptados, al requerir comprobaciones de divisibilidad con todos los 10.000 primos de la criba, para cada pareja candidata. En contraste, la criba de veto precalcula sobre los primos antes del emparejado, de manera que la comprobación final para cada pareja candidata es más trivial. Esto mantiene un coste más equiparable entre las parejas aceptadas y rechazadas.

IV-B. Análisis de los Tests de Primalidad

La Figura 3 compara el rendimiento del test de Miller-Rabin estándar con el test de Pocklington-Lehmer. Aunque ambos métodos muestran un crecimiento similar en este rango (exponencial o polinómico), el test de Pocklington-Lehmer es más eficiente que Miller-Rabin en todo el rango de tamaños de primos. Es importante destacar que estos tests de primalidad se aplican únicamente a las parejas de primos candidatos que ya han superado la fase de cribado, para que sea más representativo del uso real que tienen estos tests. En nuestras pruebas supone comprobar alrededor de 2892 ± 20 parejas, un 12,89 % del total de candidatas.

El rendimiento del test de Pocklington-Lehmer se explica en la Figura 4, que compara su comportamiento con el de Miller-Rabin al procesar números primos y compuestos. Mientras que ambos tests son rápidos para descartar compuestos, Miller-Rabin es lento para detectar la potencial primalidad de un número primo. En contraste, Pocklington-Lehmer, gracias a que conocemos la factorización de $e - 1$, es capaz de certificar la primalidad en menos tiempo.

IV-C. Evaluación en el Caso de Uso Real

Finalmente, la Figura 5 muestra los resultados de la segunda fase de evaluación, donde se mide el tiempo total para encontrar una única pareja de claves válida. En cada iteración en el proceso de encontrar una clave válida, se genera sólo un nuevo primo (e_1 o e_2) y se comprueban las nuevas parejas de este con los existentes en el conjunto contrario (P_1 o P_2). Para obtener estos resultados, se generaron 50 claves para cada tamaño de bit y se promediaron los resultados. Se compararon tres cadenas de procesamiento completas: la criba base con Miller-Rabin (curva 1), la criba base con Pocklington-Lehmer (curva 2), y criba de veto con Pocklington-Lehmer (curva 3).

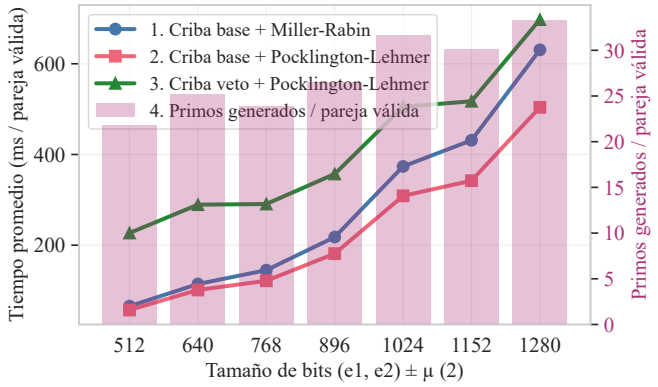


Figura 5. Comparativa de las cadenas de procesamiento completas en el escenario de uso real.

Los resultados evidencian que la simple sustitución del test de Miller-Rabin por Pocklington-Lehmer en la cadena de procesamiento base ofrece una mejora de rendimiento consistente en todo el rango de tamaños de bit, posicionándose como la opción más eficiente. Contrariamente a lo esperado de los análisis individuales de las cribas, la combinación de la criba de veto con el test de Pocklington-Lehmer muestra sistemáticamente un empeoramiento del rendimiento en este escenario de uso real, siendo incluso menos eficiente que la cadena de procesamiento base con Miller-Rabin para tamaños de bit mayores. En el siguiente punto analizamos este resultado y explicamos por qué ocurre.

V. Discusión y Trabajo Futuro

V-A. *Discusión de los Resultados*

Los resultados obtenidos en la fase de pruebas comparativas demuestran que las optimizaciones propuestas son muy efectivas de forma aislada. Tanto la criba de veto como el test de Pocklington-Lehmer superan con creces a sus alternativas estándar (criba base y Miller-Rabin, respectivamente), tal y como se observa en las Figuras 1 a 4.

Sin embargo, el análisis del caso de uso real (encontrar una única clave válida) revela un panorama más complejo. Como muestra la Figura 5, la cadena de procesamiento más eficiente combina la criba base con el test de Pocklington-Lehmer. La criba de veto, a pesar de ser más rápida de forma aislada, degrada el rendimiento global en este escenario. Su coste de precálculo no se amortiza cuando cada primo se empareja con un número reducido de candidatos (aproximadamente 15 en nuestras pruebas), que es lo que ocurre al buscar la primera clave válida. Por tanto, su uso solo es rentable en escenarios con un volumen de emparejamientos por primo mucho mayor.

En cuanto a la seguridad, las optimizaciones propuestas no la comprometen, sino que la refuerzan. La información adicional utilizada (los factores e_1 y e_2) solo está disponible durante la generación de la clave. Un atacante que solo conoce la clave pública e se enfrenta al mismo problema de factorización de un semiprimo ($e_1 e_2 = \frac{e-1}{2}$) resistente a la coalición (parámetro μ), que sigue siendo computacionalmente intratable. Adicionalmente, el uso del test de Pocklington-Lehmer, al ser determinista, elimina por completo la ínfima pero existente probabilidad de error de los tests probabilísticos como Miller-Rabin, que podrían (con una probabilidad despreciable) categorizar un número compuesto como primo. Esto proporciona una mejor garantía sobre la primalidad de las claves generadas.

Finalmente, este trabajo subraya varias ideas importantes. La primera es que utilizar toda la información adicional de un problema en lugar de tratarlo de forma genérica, nos permite barajar opciones de optimización mucho más específicas y eficientes. Desde un punto de vista computacional, se demuestra que es provechoso intentar aplicar estrategias de precálculo y reutilización de resultados. También es valioso intentar reformular algoritmos buscando calcular sobre tamaños de bit menores, que implica realizar una menor cantidad de operaciones a nivel de bit.

Por último, se refuerza la idea de no descartar el uso de tests deterministas en aplicaciones prácticas. La incorporación del test de Pocklington-Lehmer demuestra que, para problemas con una estructura matemática específica, estos tests pueden ser una alternativa práctica y superior a los tests probabilísticos, desafiando la dependencia casi exclusiva en estos últimos para aplicaciones criptográficas.

V-B. *Limitaciones y Trabajo Futuro*

A pesar de los resultados positivos, este estudio presenta ciertas limitaciones que, a su vez, abren la puerta a posibles mejoras a estudiar.

Para garantizar la estabilidad de las pruebas y poder comparar de forma clara las diferentes cadenas de procesamiento, se han mantenido fijas ciertas variables, como la cantidad de primos utilizados en el proceso de cribado. Una primera línea de trabajo futuro sería, por tanto, realizar un estudio para determinar una profundidad de criba óptima. Adicionalmente, sería de gran interés determinar el punto de equilibrio a partir del cual la criba de veto es rentable; es decir, encontrar el número mínimo de parejas por primo necesarias para que el coste del precálculo se amortice y supere en rendimiento a la criba base.

En segundo lugar, continuando con un comentario anterior, es posible integrar la generación de primos e_1 y e_2 en la criba de veto. Reutilizar los restos resultantes de la criba que se les aplica al generarlos reduciría el número de precálculos a realizar por subsequentes cribas.

Finalmente, la búsqueda de parejas de primos (e_1, e_2) se ha realizado de forma exhaustiva, introduciendo los primos de forma individual hasta encontrar una clave. Una tercera y extensa línea a explorar trataría de diseñar estrategias de procesamiento por lotes aplicadas a este caso. Por ejemplo, procesar los primos en lotes (en lugar de uno a uno) permite aprovechar mejor la arquitectura de las CPUs modernas. La ejecución de operaciones idénticas sobre un gran conjunto de datos mejora la localidad de los mismos, lo que se traduce en un uso más eficiente de la memoria caché y una predicción de saltos más acertada. Además, este enfoque abre la puerta a un mayor nivel de paralelismo mediante el uso de instrucciones SIMD (Single Instruction, Multiple Data), donde una sola instrucción puede operar sobre múltiples datos simultáneamente.

VI. Conclusión

En este trabajo se ha abordado la optimización del proceso de generación de claves para un tipo de firmas en anillo es-

pontáneas, cuya compleja estructura matemática ($e = 2e_1e_2 + 1$) impone un alto coste computacional. Se ha demostrado que el uso de información específica del problema permite mejorar sustancialmente el rendimiento. La principal contribución es la sustitución del test de primalidad estándar de Miller-Rabin por el test determinista de Pocklington-Lehmer, que no solo resulta ser más eficiente, sino que además refuerza la seguridad al eliminar la probabilidad de error de los tests probabilísticos.

Adicionalmente, se ha propuesto una criba de veto que, si bien ha demostrado ser muy eficiente en pruebas comparativas, no resulta idónea para nuestro caso de uso específico, donde la búsqueda de una única clave válida no permite amortizar el coste de su precálculo. Su aplicabilidad es para escenarios con un volumen de emparejamientos mayor. Por tanto, la cadena de procesamiento óptima para el escenario estudiado consiste en una criba base seguida del test de Pocklington-Lehmer.

Estos hallazgos mejoran la viabilidad de generar este tipo de claves en dispositivos con recursos limitados y subrayan la importancia de adaptar al problema concreto, no sólo las herramientas computacionales, sino también las técnicas matemáticas y/o criptográficas.

AGRADECIMIENTOS

Esta investigación se ha realizado financiada por los proyectos: NEWLAN (PID2022-136476OB-100) del gobierno de España (financiado por MICIU/AEI/10.13039/501100011033/FEDER EU, ERDF/EU) y por el Departamento de Educación, Ciencia y Universidades del Gobierno de Aragón, Grupo CeNIT (Ref T31_23R).

REFERENCIAS

[1] A. Riera and P. Brown, "Bringing confidence to electronic voting," en *Electronic Journal of e-Government*, vol. 2, no. 2, pp. 85–90, 2004

[2] C. A. Neff, "A verifiable secret shuffle and its application to e-voting," en *Proc. 8th ACM Conf. Computer and Communications Security (CCS '01)*, pp. 116–125. doi: 10.1145/501983.502006, 2001

[3] D. Chaum, "Blind signatures for untraceable payments," en Advances in Cryptology: Proc. CRYPTO '82, Springer, pp. 199–203, 1983.

[4] B. Adida, "Helios: Web-based open-audit voting," en *Proc. 17th USENIX Security Symp.*, pp. 335–348, 2008.

[5] C. Gentry, "A fully homomorphic encryption scheme," en *Communications of the ACM*, vol. 53, no. 3, pp. 97–105, doi: 10.1145/1666420.1666444 , 2009.

[6] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," en *Advances in Cryptology – ASIACRYPT 2001*, Springer, pp. 552–565, 2001.

[7] P. P. Tsang and V. K. Wei, "Short linkable ring signatures for e-voting, e-cash and attestation," en *Proc. 1st Int. Conf. Information Security Practice and Experience (ISPEC'05)*, pp. 48–60, 2005.

[8] J. L. Salazar and J. L. Tornos, "Optimización en la generación de claves para firmas en anillo, espontáneas y enlazables," en *Actas de la XIII Reunión Española sobre Criptología y Seguridad Informática (RECSI XIII)*, pp. 113–117, 2014.

[9] J. Camenisch and A. Lysyanskaya, "A signature scheme with efficient protocols," en *Security in Communication Networks (SCN 2002), S. Cimato, C. Galdi, and G. Persiano, Eds. Springer*, vol. 2576, pp. 268–289, 2002.

[10] G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik, "A practical and provably secure coalition-resistant group signature scheme," en *Proc. CRYPTO 2000, Springer*, vol. 1880, pp. 255–270, 2000.

[11] J. Gordon, "Strong primes are easy to find," en *Advances in Cryptology – EUROCRYPT '84*, pp. 216–223, 1984.

[12] J. Brillhart, D. H. Lehmer, and J. L. Selfridge, "New primality criteria and factorizations of $2^m \pm 1$" Mathematics of Computation, vol. 29, no. 130, pp. 620–647, Apr. 1975.

[13] M. Agrawal, N. Kayal, and N. Saxena, "PRIMES is in P," *Ann. Math.*, vol. 160, no. 2, pp. 781–793, 2004.

[14] H. W. Lenstra and C. Pomerance, "Primality Testing with Gaussian Periods," *J. Eur. Math. Soc.*, vol. 5, no. 1, pp. 1–19, 2005.

[15] H. Riesel, "Prime Numbers and Computer Methods for Factorization", *2nd ed., Birkhäuser*, 1994.

[16] R. Crandall and C. Pomerance, "Prime Numbers: A Computational Perspective", *2nd ed., Springer*, 2005.

[17] G. L. Miller, "Riemann's Hypothesis and Tests for Primality," *J. Comput. Syst. Sci.*, vol. 13, no. 3, pp. 300–317, 1976.

[18] M. O. Rabin, "Probabilistic Algorithm for Testing Primality," *J. Number Theory*, vol. 12, no. 1, pp. 128–138, 1980.

[19] R. Solovay and V. Strassen, "A Fast Monte-Carlo Test for Primality," *SIAM J. Comput.*, vol. 6, no. 1, pp. 84–85, 1977.

[20] R. Baillie and S. S. Wagstaff, "Lucas Pseudoprimes," *Math. Comput.*, vol. 35, no. 152, pp. 1391–1417, 1980.

[21] C. K. Caldwell and H. J. Dubner, "The Baillie–PSW Primality Test," *J. Recreational Math.*, vol. 25, no. 1, pp. 1–7, 1993.

Mejoras para el cifrado broadcast para mensajes cortos en WiFi

José Luis Salazar Riaño
Universidad de Zaragoza
Escuela de Ingeniería y Arquitectura
jsalazar@unizar.es

Julian Fernández Navajas
Universidad de Zaragoza
Escuela de Ingeniería y Arquitectura
navajas@unizar.es

Guillermo Azuara Guillén
Universidad de Zaragoza
Escuela Politécnica de Teruel
azuara@unizar.es

***Resumen*—Este documento presenta una mejora a una propuesta del uso del Teorema Chino de los Restos para el tráfico de bajada desde un punto de acceso a varias estaciones, utilizando el medio inalámbrico de forma simultánea. La modificación consiste en la sustitución del cifrado de clave pública por el de clave privada para mejorar su eficiencia y utiliza un tamaño de mensajes variable para poder introducir en la misma trama agregada varios mensajes de tamaño diferente y con destino a varias estaciones diferentes, manteniendo la confidencialidad de cada uno de ellos. El beneficio se aprecia en la gestión del punto de acceso de su tiempo de transmisión, y de forma más notoria en la reducción del tiempo de procesado.**

***Index Terms*—Teorema Chino de los Restos, Tráfico multi-usuario, IEEE 802.11**

I. Introducción

El uso creciente de servicios móviles ha impulsado un aumento en la transmisión de paquetes pequeños, especialmente en aplicaciones con altas restricciones de interactividad, como VoIP (Voice over IP) y juegos en línea. En estos casos, es necesario enviar fragmentos de información de manera frecuente, lo que incrementa la ineficiencia debido a las cabeceras de los protocolos de comunicación. Esta ineficiencia es aún más notable en redes inalámbricas, donde las restricciones físicas pueden afectar la Calidad de Servicio (QoS). Un claro ejemplo de este problema se observa en la transmisión de paquetes pequeños sobre redes WiFi (802.11) [1]. La cabecera MAC y la FCS del estándar suman 40 bytes, lo que significa que un paquete VoIP de 40 bytes tiene una eficiencia de solo el 50 %. En cambio, para paquetes más grandes, el impacto de la cabecera es menos significativo. Además, en redes inalámbricas compartidas, los mecanismos de control de acceso al medio añaden más sobrecarga, reduciendo aún más la eficiencia. Para abordar estos problemas, se han desarrollado diversas soluciones, como la agregación de tramas en 802.11 y el uso de paquetes broadcast. Sin embargo, la seguridad sigue siendo un factor que introduce más ineficiencia, ya que los encabezados de seguridad pueden aumentar significativamente la carga total de datos transmitidos. En este contexto, los enfoques que combinan agregación y seguridad se vuelven cruciales: si un único encabezado seguro puede abarcar varios paquetes pequeños, su impacto se distribuye, mejorando la eficiencia general. En la literatura, se han propuesto soluciones para equilibrar seguridad y eficiencia. Por ejemplo, en [2], se presenta un método en el que varios paquetes cifrados pequeños se agrupan en una sola transmisión, asegurando que cada subpaquete sólo pueda ser descifrado por su destinatario legítimo. Este esquema se inspira en los sistemas de transmisión segura de televisión por cable, donde los encabezados individuales se combinan en una única cabecera para reducir la sobrecarga de datos. Otro enfoque relevante es el presentado en [7], donde se optimiza este modelo para redes WiFi. Esta propuesta mantiene la seguridad del sistema, pero introduce mejoras en dos aspectos clave: el tamaño de los números primos usados en la clave privada y la eficiencia en la generación de semillas aleatorias. Sin embargo, aun quedan problemas sin resolver, como el retardo causado por el procesamiento de operaciones criptográficas con exponenciaciones modulares y la variabilidad en los tamaños de los paquetes en entornos reales, lo que afecta la eficiencia del procesamiento. En este artículo, abordamos estas limitaciones mediante dos estrategias principales. Primero, introducimos criptografía simétrica para reducir el tiempo de procesamiento en el cifrado de los datos. Segundo, reinterpretamos el Teorema Chino de los Restos para optimizar la transmisión de paquetes de distintos tamaños dentro de una misma estación de comunicaciones. Con estas mejoras, buscamos equilibrar eficiencia y seguridad en entornos inalámbricos con alta demanda de transmisión de datos

En el siguiente punto realizaremos una descripción del algoritmo de cifrado broadcast para mensajes cortos que pretendemos mejorar. A continuación describiremos nuestra propuesta de mejora, para hacer un breve análisis de la misma en el siguiente apartado. Finalmente, terminaremos con las conclusiones.

II. Cifrado broadcast para mensajes cortos

En esta sección haremos una breve descripción del modelo de cifrado broadcast genérico y para mensajes cortos para poder exponer con más claridad nuestra propuesta de mejora.

II-A. Modelo de cifrado broadcast

Para describir este tipo de cifrado usaremos el modelo para un sistema de cifrado broadcast propuesto en [2]. Formalmente, dicho sistema consta de cuatro algoritmos probabilísticos:

- **Configuración** (λ)**:** Toma como entrada el parámetro de seguridad λ, genera los parámetros globales del sistema y devuelve la clave secreta de cifrado EK.
- **Extracción** (i, EK)**:** Toma como entrada el índice i del usuario, junto con la clave de cifrado EK y genera las claves privadas del usuario (p_i, x_i).
- **Cifrado** $(u_1, u_2, ..., u_n, EK, m_1, m_2, ..., m_n)$**:** Toma como entrada los identificadores de n usuarios $(u_i), n$ mensajes (m_i) y la clave de cifrado EK. Produce (Hdr, ET) donde Hdr es un número aleatorio para el

cifrado/descifrado, y ET es el texto cifrado, calculado con todos los textos planos.

- **Descifrado** (Hdr, ET, i, p_i, x_i)**:** Toma como entrada el encabezado aleatorio Hdr, el texto cifrado ET, el índice del usuario i y sus claves privadas p_i y x_i, y genera el texto descifrado m_i, para i.

Para asegurar la corrección del sistema se requiere que $\forall i \in \{1, \ldots, n\}$, si $EK \leftarrow$ Configuración (λ), $(p_i, x_i) \leftarrow$ Extracción (i, EK), $(Hdr, ET) \leftarrow$ Cifrado $(u_1, u_2, \ldots, u_n, EK, m_1, m_2, \ldots, m_n)$, entonces $m_i \leftarrow$ Descifrado (Hdr, ET, i, p_i, x_i).

II-B. Entorno de Aplicación del algoritmo: WiFi

Una vez presentada la solución propuesta en [2], analizamos el equilibrio entre eficiencia y seguridad en un escenario inalámbrico, concretamente en el estándar IEEE 802.11 (Wi-Fi) [1].

Dado que la transmisión de paquetes pequeños en redes WiFi implica restricciones en términos de seguridad y eficiencia, el uso del algoritmo propuesto está condicionado principalmente por el tamaño de las claves y su generación. Nuestro objetivo es encontrar un equilibrio entre eficiencia y seguridad, dos aspectos que, por definición, suelen estar en conflicto. Según la propuesta, el algoritmo será más eficiente siempre que el tamaño de los paquetes de usuario tenga un límite superior. De hecho, el rendimiento óptimo se alcanza cuando el tamaño de la clave es solo ligeramente mayor que el del paquete. Así, en servicios que transmiten paquetes de tamaño uniforme, como VoIP, es posible seleccionar un tamaño de clave óptimo con facilidad.

Al analizar un enlace inalámbrico, no solo debemos considerar el ancho de banda en términos de bytes enviados, sino también los mecanismos de acceso al medio. Si múltiples paquetes deben esperar a que el canal esté libre, se genera una acumulación de tiempos de espera. Por ello, reducir el número de tramas transmitidas (por ejemplo, enviando una sola trama multicast para varios usuarios en lugar de múltiples tramas unicast) mejora el uso del espectro.

El estándar 802.11n incorpora dos mecanismos de agregación: **A-MPDU** (Aggregated MAC Protocol Data Unit), que permite enviar varias MPDU en una sola transmisión, y **A-MSDU** (Aggregated MAC Service Data Unit), que agrupa varias MSDU a nivel de capa MAC [4]. En la literatura se ha propuesto el uso de estos mecanismos para la transmisión de tramas multicast [5][6]. Dado este contexto, el sistema propuesto en [2] resulta compatible con estas estrategias de agregación. En particular, el uso de **A-MSDU** es más eficiente en términos de transmisión, ya que evita la sobrecarga de enviar múltiples cabeceras MAC para cada MPDU. Además, aporta mejoras en seguridad, tanto real como percibida, ya que cada MSDU agregado en la transmisión broadcast permanecerá inaccesible para quienes no sean sus legítimos destinatarios. Recordemos que, en una transmisión broadcast estándar, todos los miembros del grupo de destino pueden acceder al mensaje.

Por tanto, en este escenario, la distribución de roles es bastante intuitiva: los usuarios y la cabecera mantienen su función, mientras que los mensajes a cifrar corresponden a las diferentes MSDU.

II-C. Modelo de cifrado broadcast para mensajes cortos

La adaptación a mensajes cortos de [2] se hace por medio de criptografía simétrica, pero utilizando aritmética modular (Teorema Chino de los Restos), que resultando ser más lenta, quedaría parcialmente amortiguada por la corta longitud de los mensajes.

- **Configuración** (λ)**:** Toma como entrada el parámetro de seguridad λ. Los parámetros globales del sistema se generan de la siguiente manera: primero se eligen n primos p_i (uno por usuario) y n enteros aleatorios $x_i \in \mathbb{Z}^*_{p_i}$, tales que $mcd(x_i, p-1) = 1$. Se define $N = \prod_{i=1}^{n} p_i$ y $EK = ((p_1, x_1), (p_2, x_2), ..., (p_n, x_n))$, donde cada par corresponde a la clave secreta de descifrado de cada usuario.
- **Extracción** (i, EK)**:** Se envía de forma segura a cada usuario, i, su clave secreta (p_i, x_i).
- **Cifrado** $(u_1, u_2, ..., u_n, EK, m_1, m_2, ..., m_n)$**:** Elige un entero aleatorio $Hdr \xleftarrow{\$} \mathbb{Z}^*_{min_j\{p_j\}}$ y define $Hdr_i = min\{g \leq Hdr \backslash g \text{ es un generador de } \mathbb{Z}^*_{p_i}\}$ para dar como salida calcular (Hdr, ET), siendo

$$ET = \sum_{i=1}^{n} [(m_i + Hdr_i^{x_i}) \text{mód } p_i] \frac{N}{p_i} \left[\left(\frac{N}{p_i} \right)^{-1} \text{mód } p_i \right] (\text{mód } N) \quad (1)$$

- **Descifrado** (Hdr, ET, i, p_i, x_i)**:** Este algoritmo da como salida $m_i = (ET - Hdr_i^{x_i})(mod p_i)$ que es consecuencia del Teorema Chino de los Restos.

Este esquema de cifrado broadcast para mensajes cortos fue mejorado y adaptado en [7] para mejorar la eficiencia y bajar el retardo de procesado.

II-D. Problemas de implementación

Pese a las mejoras efectuadas en [7], que elevaron notablemente la eficiencia, la exigencia de un tamaño mínimo de seguridad para los paquetes que inicialmente eran cortos, hacía que ya no lo fueran tanto, lastrando la eficiencia y el retardo de procesado.

El empleo de exponenciaciones modulares para el cifrado se hace muy caro a medida que crece el tamaño del paquete y el crecimiento de este gasto no es lineal. De alguna manera hay que limitar el tamaño máximo del empleo de tales operaciones.

Por otra parte, el ajuste de tamaños que provee el teorema chino de los restos, parece adecuado y no incrementa en exceso el retardo de procesado. En cambio, la variabilidad del tamaño de los mensajes hace que el uso de una clave por usuario sea ineficiente, pero el uso de una familia de claves eleva mucho la gestión de claves en ambos extremos de la comunicación.

III. Solución propuesta

Con estos problemas encontrados en este escenario descrito, plantearemos un nuevo esquema de cifrado broadcast en WiFi para paquetes cortos empleando criptografía simétrica por ser más eficiente para tamaños de paquete menos pequeños. Para ello debemos definir el esquema de cifrado simétrico con sus tres algoritmos característicos, (Gen, Enc, Dec):

- **Gen(**λ**):** Como en todos los algoritmos de generación de clave, ésta debe ser calculada a partir de una distribución

aleatoria uniforme con una longitud λ marcada por el algoritmo de cifrado en bloque que vayamos a utilizar. En nuestro caso el resultado de este algoritmo será la salida de una función Hash con un tamaño de salida adecuado, instanciando un oráculo aleatorio. Tal y como se puede ver más abajo $k \xleftarrow{\$} H(ctr||p||Hdr)$, siendo ésta la salida del algoritmo.

- **Enc**(k, M)**:** El único requisito que se le pide a este algoritmo de cifrado en bloque es que conserve la longitud, es decir que $|M| = |Enc(M)|$, condición que cumple la mayor parte de los algoritmos de este tipo de uso generalizado. Su salida sera D
- **Dec**(k, M)**:** Este algoritmo ha de ser el asociado con Enc y que sea correcto, es decir que $Dec(Enc(M)) = M$.

Un ejemplo que podría instanciar este esquema podría ser SHA3-256 [8] como función hash que tiene 256 bits de salida y los algoritmos de cifrado y descifrado AES-256 [9] en modo CTR que utiliza una clave de 256 bits para un bloque de mensaje de 128 bits. Por otra parte, solucionaremos la gestión de claves, utilizando el Teorema Chino de los Restos con más profundidad, empleando una única clave por usuario, pero pudiendo emplear potencias de este valor que aumentaría su capacidad de cifrado.

Volveremos a utilizar el esquema habitual para esta tipología de cifrado:

- **Configuración** (λ)**:** Toma como entrada el parámetro de seguridad λ y los parámetros globales del sistema, devolviendo la clave secreta de cifrado EK, consistente en dos n-tuplas: una de valores primos $P = (p_1, \ldots, p_n)$ y otra de claves iniciales $K = (k_{01}, \ldots, k_{0n})$ correspondientes a los n-usuarios para los que tiene capacidad el sistema, siendo los valores k_{0i} las claves secretas de un cifrado de bloque en el esquema de cifrado simétrico (Gen, Enc, Dec) entre i y el emisor del mensaje descrito anteriormente.
- **Extraccción** (i, EK)**:** Toma como entrada el índice i del usuario, junto con la clave de cifrado EK y suministra de manera segura sus correspondientes valores k_{0i} y p_i.
- **Cifrado** $(EK, m_1, m_2, ..., m_n)$**:** Para cada mensaje m_i, se calcula el valor e_i que resultaría ser el menor entero, tal que $p_i^x > 2^{l_i}$, donde l_i sería la longitud máxima en bits del cifrado del mensaje m_i, con la función Enc. Es decir, que si Enc es un cifrado en bloque de tamaño b bits que preserva longitud,

$$e_i = \left\lceil \log_{p_i} \left(\frac{\lceil \log_2 m_i \rceil}{b} \right) \right\rceil \tag{2}$$

Definimos

$$N_1 = \prod_{i=1}^{n} p_i \tag{3}$$

$$k_{j+1,i} = Enc_{k_{ji}}(p_i) \tag{4}$$

$$Hdr = \left(\sum_{i=1}^{n} Enc_{k_{j+1,i}}(e_i) \frac{N_1}{p_i} \left[\left(\frac{N_1}{p_i} \right)^{-1} \bmod p_i \right] \right) \bmod N_1 \tag{5}$$

$$N_2 = \prod_{i=1}^{n} p_i^{e_i} \tag{6}$$

Ahora actualizamos en EK la n-tupla K, reemplazando k_{ji} por k_{j+1i}. Calculamos

$$ET = \left(\sum_{i=1}^{n} Enc_{k_{j+1,i}}(m_i) \frac{N_2}{p_i^{e_i}} \left[\left(\frac{N_2}{p_i^{e_i}} \right)^{-1} \bmod p_i^{e_i} \right] \right) \bmod N_2 \tag{7}$$

para dar como salida (Hdr, ET).

- **Descifrado** (Hdr, ET, i, p_i)**:** El algoritmo calculará

$$k_{j+1,i} = Enc_{k_{ji}}(p_i) \tag{8}$$

$$e_i = Dec_{k_{j+1i}}[Hdr \pmod{p_i}] \tag{9}$$

Así, calcula

$$Enc_{k_{j+1,i}}(m_i) = ET \pmod{p_i^{e_i}} \tag{10}$$

La salida del algoritmo será

$$m_i = Dec_{k_{j+1i}}(Enc_{k_{j+1i}}(m_i)) \tag{11}$$

IV. Solución propuesta

Una vez descrita la propuesta, analicemos el resultado obtenido en los dos aspectos que queríamos mejorar: seguridad y eficiencia.

IV-A. Seguridad

Podemos comprobar la seguridad del sistema a partir del modelo definido en [2]. De hecho es sencillo demostrar que cumple los dos niveles de seguridad definidos, tanto para un entorno sin corrupción (Basic Selective Security) como para un entorno en la que puede haber nodos corruptos (Strong Selective Security). Del resultado extraído de [10] podemos afirmar que para demostrar la seguridad de todo el sistema, simplemente es necesario probar que uno de los canales para un usuario lo es. Podemos apreciar que la mezcla de la información de los mensajes se hace una vez que han sido cifrados con AES. Además cada clave se renueva con cada mensaje siguiendo la estrategia de seguridad posterior [11]. Suponiendo que AES es INDCPA-seguro y que los esquemas de seguridad posterior basados en cifrados INDCPA-seguros, como AES, tambien los son; podemos deducir que nuestra propuesta mantiene la seguridad ya que el canal de información de cada uno de los usuarios no comparte información en claro en ningún momento y sólo lo hacen con la información cifrada. Con esto cumplimos las premisa de [10] y en consecuencia podemos afirmar que el sistema es seguro, tanto si el resto de participantes comparten información o no.

IV-B. Eficiencia

En la versión original se proponía una exponenciación modular con una clave, p_i, que tenía un tamaño mínimo que dependía del nivel de seguridad, λ, escogido. Si el mensaje que se transmitía era de mayor tamaño, se aumentaba el tamaño de la clave, ajustando el tamaño de la clave por encima del tamaño del mensaje (que a su vez aumentaba el nivel de seguridad), lo que implicaba un aumento del nivel de procesado, un incremento de retardo y por lo tanto una disminución de la eficiencia.

Hemos eliminado el cálculo de la exponenciación modular, que incrementaba notoriamente el procesado, y por otra parte, podemos reutilizar la clave en un mismo mensaje, aunque éste

aumente de tamaño. Si reducimos el tamaño mínimo de p_i, al nivel atómico de seguridad para satisfacer el impuesto por λ, cuando se incremente el tamaño del mensaje por encima de ese nivel atómico, simplemente podremos incrementar la capacidad de esa clave elevándola al exponente que deje $p_i^{e_i}$ por encima del valor de ese mensaje que querramos transmitir. Por ejemplo, si el nivel atómico de seguridad es de 160 bits, necesitaríamos un primo p_i, de al menos 161 bits. Si tuviéramos que transmitir un mensaje de 400 bits, necesitamos al menos el triple del espacio atómico, que se correspondería con un valor de $e_i = 3$, ya que el tamaño de $p_i^{e_i}$, será e_i veces mayor que el de p_i. Con ese cambio de paradigma, seguimos cumpliendo las hipótesis del Teorema Chino de los Restos y podemos seguir aprovechando sus ventajas.

V. Conclusiones

En este trabajo hemos presentado un algoritmo de cifrado de información para un medio compartido manteniendo la confidencialidad de los usuarios. Este medio compartido se establece en el entorno WiFi defnido en el standard IEEE 802.11[1] para el tráfico de bajada y mediante tramas multicast. La propuesta presentada mejora otra anterior en eficiencia, al eliminar las operaciones modulares y dejar todo el peso de la confidencialidad a la criptografía de cifrado de bloque, sin pérdida de seguridad.

Agradecimientos

Esta investigación se ha realizado financiada por los proyectos: NEWLAN (PID2022-136476OB-100) del gobierno de España (financiado por MICIU/AEI/10.13039/501100011033/FEDER EU, ERDF/EU) y por el Departamento de Educación, Ciencia y Universidades del Gobierno de Aragón. Grupo CeNIT (Ref T31_23R).

Referencias

[1] IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std 802.11, 2016.

[2] D. H. Phan, D. Pointcheval, V. C. Trinh, "Multi-channel broadcast encryption," en *ASIACCS 2013, K. Chen, Q. Xie, W. Qiu, N. Li, and W. G. Tzeng, Eds., Hangzhou, China: ACM Press*, pp. 277–286, 2013.

[3] J. L. Salazar Riaño, J. M. Saldaña, J. Fernández-Navajas, J. Ruiz-Mas, G. Azuara Guillén, "Short Message Multichannel Broadcast Encryption," en *Advances in Intelligent Systems and Computing, vol. 1267. Cham, Switzerland: Springer*, pp. 122–131, 2021.

[4] B. Ginzburg and A. Kesselman, "Performance analysis of A-MPDU and A-MSDU aggregation in IEEE 802.11n," en *IEEE 2007 Sarnoff Symposium, Princeton, USA*, pp. 1–5, 2007.

[5] E. Goh, H. Shacham, N. Modadugu, et al., "Sirius: Securing remote untrusted storage," en *Proceedings of Network and Distributed System Security Symposium (NDSS) 2003, San Diego, USA, pp. 131–145*, 2003.

[6] M. Kallahalla, E. Riedel, R. Swaminathan, et al., "Plutus: Scalable secure file sharing on untrusted storage," en *Proceedings of USENIX Conference on File and Storage Technologies (FAST) 2003, San Francisco, USA*, pp. 29–42, 2003.

[7] J. L. Salazar, J. Fernández-Navajas J. Ruiz-Mas, G. Azuara Guillén, "Implementación de cifrado broadcast para mensajes cortos en WiFi," en *Actas de la XVII Reunión Española sobre Criptología y Seguridad Informática (RECSI XVII)*, pp. 113–117, 2014.

[8] NIST, FIPS 202: "SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions," National Institute of Standards and Technology, Gaithersburg, MD, USA, Aug. 2015.

[9] NIST, FIPS 197: "Advanced Encryption Standard (AES)," National Institute of Standards and Technology, Gaithersburg, MD, USA, Nov. 2001.

[10] D. Boneh, C. Gentry, and B. Waters, "Collusion resistant broadcast encryption with short ciphertexts and private keys," en *Advances in Cryptology – CRYPTO 2005, V. Shoup, Ed. Berlin, Germany: Springer*, pp. 258–275, 2005.

[11] M. Bellare and B. Yee, "Forward-security in private-key cryptography," en *Cryptology ePrint Archive, Report 2001/035*, 2001. [Online]. Available: https://eprint.iacr.org/2001/035.pdf

[12] Nombre autores: "Título del trabajo", en *nombre revista/conferencia*, vol. x, n. y, pp. aa-bb, año.

A Linguistic Intelligence Framework for Proactive Ransomware Detection

Muhammad Junaid iqbal
Universitat Oberta de Catalunya, Barcelona, Spain
CYBERCAT-Center for Cybersecurity Research of Catalonia
miqbal6@uoc.edu

Jordi Serra-Ruiz
Universitat Oberta de Catalunya, Barcelona, Spain
CYBERCAT-Center for Cybersecurity Research of Catalonia
jserrai@uoc.edu

***Abstract*—The dominant paradigm in ransomware defence has overwhelmingly focused on technical indicators of compromise (IOCs) within malware binaries and anomalous system behaviours. While effective to a degree, this approach is inherently reactive, struggling against polymorphism, encryption, and the rapid evolution of attacker Tactics, Techniques, and Procedures (TTPs). This paper proposes a novel, complementary framework for ransomware detection and attribution that shifts the focus from code to communication. We posit that the ransom note a persistent, human-generated artifact serves as a stable and rich source of linguistic and operational intelligence that is frequently obfuscated in the technical payload. Unlike its associated binary, the ransom note is a direct communication from the attacker, designed to instruct, persuade, and intimidate its victim. This paper introduces the Ransomware Note Data (RAND) framework, which systematizes the extraction and analysis of features from these notes, including lexicon, syntax, payment details, and psychological appeals. We argue that this methodology can achieve three primary objectives: 1) earlier detection of ransomware campaigns before robust IOCs are disseminated, 2) reliable clustering of seemingly disparate attacks to a common threat actor, and 3) forecasting emerging extortion strategies. By treating ransom notes as a critical dataset, we can decode the attacker's "unspoken signature" and build a more resilient, proactive defence posture.**

***Index Terms*—Ransomware, Threat Detection, Linguistic Analysis, Threat Intelligence, TTPs, Cybersecurity, Digital Forensics, Incident Response.**

I. Introduction

In recent years, the escalation of sophisticated cyberattacks has underscored the growing limitations of traditional security paradigms. Across sectors and jurisdictions, digital infrastructures are increasingly exposed to threats that adapt rapidly, exploit systemic vulnerabilities, and frequently outpace the defensive mechanisms designed to contain them. This expanding threat environment has forced organizations and researchers to reassess long-established assumptions about how malicious activity can be identified, mitigated, and anticipated. As adversaries diversify their methods, from covert intrusion techniques to carefully orchestrated extortion campaigns, the need for multidimensional analytical frameworks has become both evident and urgent. Within this context, understanding the strategies, artifacts, and communication patterns that surround these attacks emerges as a critical component of a more resilient cybersecurity posture.

A. The Detection Gap in the Ransomware Ecosystem

The global threat landscape continues to be dominated by ransomware, a form of cyber extortion that has evolved from simple data encryption to complex double and triple extortion schemes. The financial and operational damages inflicted on critical infrastructure, corporations, and individuals are measured in billions of dollars annually [2]. Conventional defensive mechanisms are primarily technical, relying on signature based antivirus software, heuristic analysis of binary code, and behavioural monitoring of system processes for activities indicative of encryption (e.g., rapid file renaming, mass IO requests). However, a significant gap exists in this defensive strategy. Malware authors are adept at employing techniques that render technical analysis less effective. Polymorphic and metamorphic code can alter a binary's signature with each iteration, while sophisticated encryption routines and "living off the land" techniques (using legitimate system tools) can make behavioural detection a challenge [6]. The result is a cat and mouse game where defenders are often one step behind, relying on the discovery and distribution of IOCs *after* an attack has been successful. This paper identifies and addresses this gap by proposing a paradigm shift. We argue for the formalized analysis of a non-technical, yet universally present, artifact: the ransom note. Traditionally dismissed by analysts as a simple text file to be discarded after the initial read, the ransom note is, in fact, a data rich communication. It is the one component of the attack that is explicitly designed for human consumption. As such, it is subject to different evolutionary pressures than the malware code. While the binary must evolve to evade detection, the note must remain clear, persuasive, and instructionally stable to successfully facilitate payment. This paper introduces a novel framework for leveraging this overlooked dataset. We will demonstrate how the systematic analysis of ransom notes can provide a stable set of signals for detection, attribution, and strategic forecasting, effectively making the attacker's own words a powerful weapon in the defender's arsenal.

II. Literature Review – Beyond the Binary

We will show the two main ways of identifying malware based on the techniques used for that purpose.

A. Static Analysis

Static analysis of ransomware typically involves examining the malicious code without executing it, focusing on intrinsic properties to identify threats. Noman et al.[7] (2021) developed a static analysis tool that disassembles Windows executables to extract API calls from associated DLLs, matching them against known malicious behaviour signatures. While achieving high accuracy for keyloggers and backdoors, ransomware detection was lower (66.6%), with false positives arising from

benign applications using similar APIs. This highlights the method's efficiency for known threats but also its vulnerability to obfuscation and novel malware, suggesting the need for hybrid approaches. Further refining static analysis, Johnson et al. [9] addressed the limitations of dynamic analysis against evasion techniques. They proposed an opcode-based classification enhanced with machine learning. By extracting opcode sequences via disassembly, applying TF-IDF feature selection with N-gram modelling, and training an ensemble learning classifier, their approach significantly outperformed traditional models. This demonstrates the potential of opcode-level static analysis combined with ML for faster and more reliable ransomware detection, particularly against samples designed to evade dynamic environments. İbrahim et al. [8] extended static analysis to Android malware, proposing a deep learning framework that integrates diverse features like permissions, API calls, services, opcode sequences, application size, and fuzzy hashes. Their functional API deep learning model, utilizing a GRU layer for fuzzy hashes, achieved a 99.5% F1-score in binary classification and 97% in multiclass classification on the CICMalDroid 2020 dataset. This work underscores the strength of comprehensive feature extraction combined with deep learning for countering sophisticated Android malware, applicable to the broader challenge of ransomware.

B. Dynamic Analysis

Dynamic analysis involves executing suspicious files in a controlled environment (sandbox) to observe their behaviour, which is crucial for detecting polymorphic and obfuscated threats. Gulmez et al. [10] introduced XRan, an explainable deep learning system for ransomware detection based on dynamic analysis. XRan extracts API call sequences, DLL usage, and mutex operations from executables run in a Cuckoo Sandbox. In the context of wireless sensor networks (WSNs), Bala et al.[11] addressed replication attacks by proposing an enhanced clustering protocol, NI-LEACH, integrated with an Intrusion Detection System (IDS). NI-LEACH optimizes cluster size based on node energy and range, improving energy efficiency and detection accuracy. Sharma et al.[12] proposed an integrated ransomware defence framework for IoT environments, combining dynamic analysis, honeypot traps, and machine learning. This system deploys dummy honeypot files, monitoring them for access or modification attempts by ransomware processes. Upon detection, the process is halted, and an alert is generated. Concurrently, an AES-256 encryption service protects user-selected sensitive files, with regular encrypted backups ensuring data resilience. Experiments demonstrated 100 % success in safeguarding protected files against multiple crypto-ransomware families, outperforming individual honeypot or ML-only approaches. This integrated strategy of detection, prevention, and protection is particularly suited for IoT devices, offering a holistic defence against ransomware.

III. Dataset and Exploratory Analysis

We curated two corpora: (i) attacker-written ransom notes and (ii) benign/marketing texts as a control. Exploratory analysis reveals strong lexical and stylistic divergence that underpins our framework. Fig. 1 summarizes word usage, while Figs. 2 detail top unigram.

A. Word Length Distribution

The distribution of word lengths, shown in Fig. 1, reveals a notable difference between ransomware notes and benign texts. Ransomware notes exhibit a higher frequency of shorter words, peaking around 3–4 characters. This suggests a preference for concise, direct, and imperative language, typical of instructions and threats. In contrast, benign texts show a broader distribution, with a significant number of longer words, indicative of more descriptive, elaborate, and nuanced communication. This observation supports the hypothesis that ransomware authors prioritize clarity and immediate impact over stylistic variety, making word length a useful discriminatory feature for the RAND framework's lexical and syntactic layers.

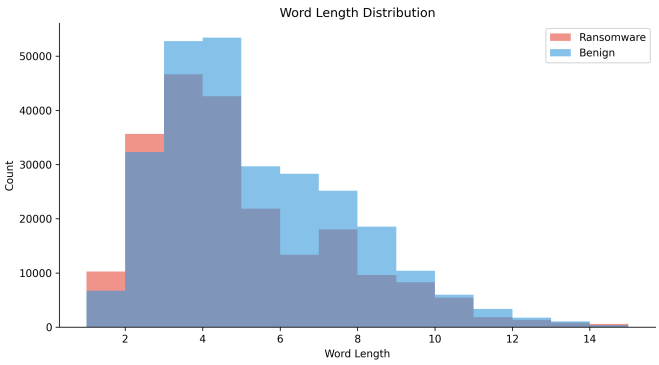


Fig. 1. Word length distribution. Ransomware notes favour short directive terms; benign texts use longer descriptive words.

B. Sentence Length Distribution

Ransomware notes display a tighter clustering of shorter sentences, with a median around 5–10 words. This confirms the use of simple, declarative, and often imperative sentence structures, designed to convey critical information and commands unambiguously. While outliers with longer sentences exist, the overall pattern indicates a focus on directness. Benign texts, however, show a wider spread of sentence lengths, reflecting more complex grammatical structures, descriptive passages, and varied communication styles. This distinction in sentence structure is a powerful indicator for differentiating between the two text types, contributing to the RAND framework's ability to identify the "unspoken signature" of ransomware.

C. Top Unigrams in Ransom Notes

Fig. 2 highlights a compact set of high-frequency tokens. These terms form a minimal vocabulary that encodes the extortion workflow: a declared condition (data made inaccessible), an offered remedy (decryption), and an explicit action (payment or contact). The narrowness of this lexicon implies low topical drift and strong priors for content filters. From a RAND perspective, this plot validates the *Lexical & Syntactic* layer as a reliable source of early indicators: even when binaries mutate, the communicator's intent forces recurrence of these tokens to maintain conversion.

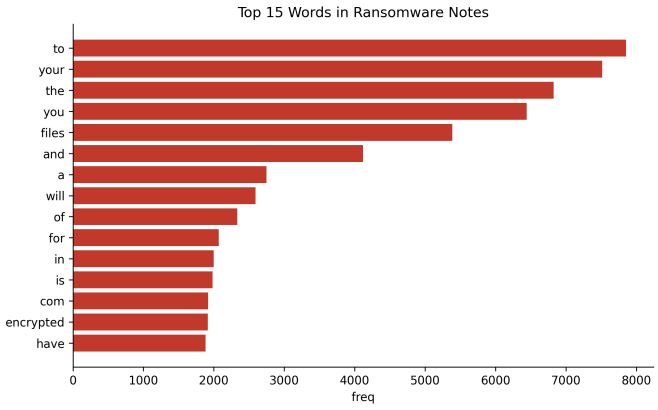


Fig. 2. Top 15 words in ransomware notes. Frequent attacker terms emphasize encryption and coercion

D. Top Discriminative TF-IDF Features (Ransom - Benign)

Figure 3 presents the top discriminative TF-IDF (Term Frequency-Inverse Document Frequency) features, illustrating the significant lexical divergence between ransomware notes and benign texts. Terms like "files", "encrypted", "decrypt", "key" and "bitcoin" show substantially higher TF-IDF scores in ransomware notes, underscoring their critical role in communicating the attack's nature and required actions. These terms directly reflect the core intent of extortion—data compromise and demand for payment. Conversely, benign texts are characterized by an entirely different vocabulary. This clear separation in TF-IDF scores confirms that the specialized lexicon of ransomware notes provides a robust, quantifiable signal for distinguishing them from ordinary communication, making it a cornerstone for effective detection within the RAND framework.

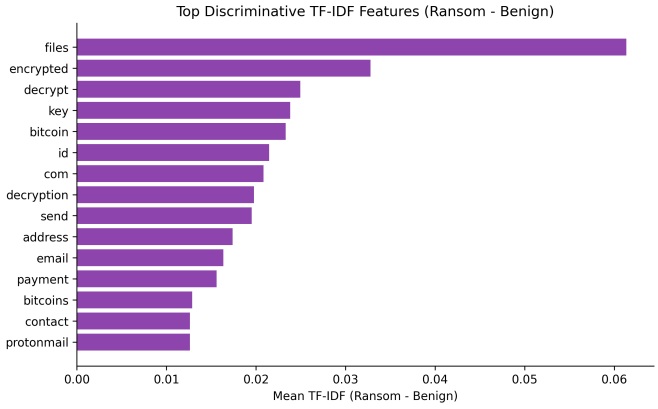


Fig. 3. Top discriminative TF-IDF features (Ransom — Benign). Clear lexical divergence between attacker intent and casual communication.

IV. Theoretical Foundation: A Stable Signal

The core thesis of this research rests on several key properties of ransom notes that make them are uniquely valuable for analysis, especially when contrasted with their associated malware.

1. The Communication Imperative vs. The Obfuscation Imperative: A malware binary's primary imperative is to execute its payload without detection. It is therefore optimized for obfuscation, mutation, and evasion. Conversely, a ransom note's primary imperative is successful communication. It must be found, read, and understood by the victim to achieve the economic goal of the attack.

2. Persistence of Style and Content: The technical components of a ransomware campaign can change rapidly. Encryption algorithms may be updated, vulnerabilities patched, and delivery mechanisms altered. However, the "voice" of the threat actor their chosen lexicon, tone, sentence structure, and the specific instructions they provide exhibits a high degree of stability. The payment addresses (cryptocurrency wallets), contact methods (TOR URLs, email addresses), and deadlines are operational necessities that remain consistent across attacks for a given period.

3. A Direct Window into TTPs and Psychology: The note is a deliberate communication that reveals the attacker's model of their victim. The language used whether threatening, apologetic, or professional reflects a psychological strategy. The specific instructions reveal operational TTPs: which communication platforms they prefer (e.g., TOX, Jabber, Telegram), which payment currencies they demand (e.g., Bitcoin, Monero), and whether they offer a "proof" service. This is human intelligence (HUMINT) delivered directly to the victim's desktop. If the malware binary is the weapon, the ransom note is the perpetrator's fingerprint left on the weapon. It is a stable, human centric artifact in an otherwise volatile, technical environment.

V. The RAND Framework: An Ontology for Analysis

To systematically analyse ransom notes, we propose the **Ransomware Note Data (RAND)** framework. This framework deconstructs a note into four analytical layers, transforming unstructured text into structured, quantifiable data.

A. Linguistic Feature Evidence

Figs. 4 illustrate recurrent phrase structures and discriminative terms that inform RAND's lexical/syntactic layer and operational instruction extraction.

B. Top 15 Bigrams in Ransomware Notes

Figure 4 presents the top 15 bigrams (two-word sequences) found in ransomware notes. This reveals highly recurrent phrases that are central to the extortion process. Bigrams such as "files encrypted", "decrypt files", "protonmail com", "recover files" and "private key" consistently appear, forming a direct narrative of compromise and resolution. These phrases are not accidental; they are carefully chosen to convey specific instructions and reinforce the attacker's demands. This reinforces the RAND framework's ability to extract stable, context-rich features from the linguistic patterns, even beyond single words.

C. Top 15 Trigrams in Ransomware Notes

Expanding on bigrams, Figure 5 displays the top 15 trigrams (three-word sequences) in ransomware notes. These longer phrases offer even richer semantic context, revealing more complete action-oriented statements. Examples like "important files encrypted", "documents photos databases", "decrypt files need" and "photos databases important" directly

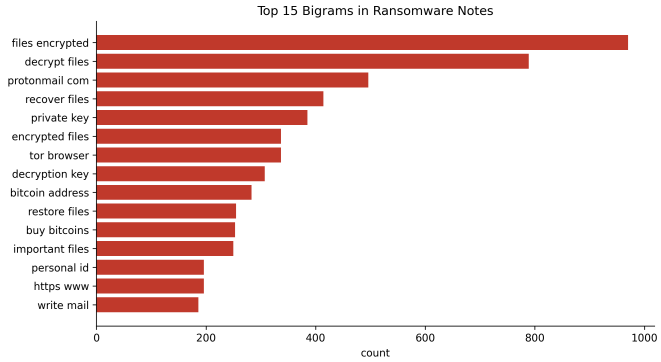


Fig. 4. Top 15 bigrams in ransomware notes. Structured threats (*files encrypted, decrypt files*) appear consistently.

describe the victim's compromised data and the necessary steps for recovery. Phrases related to communication and payment, such as "www torproject org" and "bitcoin address payment" are also prominent. The consistent recurrence of these structured threats and instructions provides strong evidence of templated language, a hallmark of ransomware campaigns. Such trigrams are invaluable for the RAND framework, as they capture more complex operational details and attacker intent, offering a deeper linguistic fingerprint for detection and attribution.

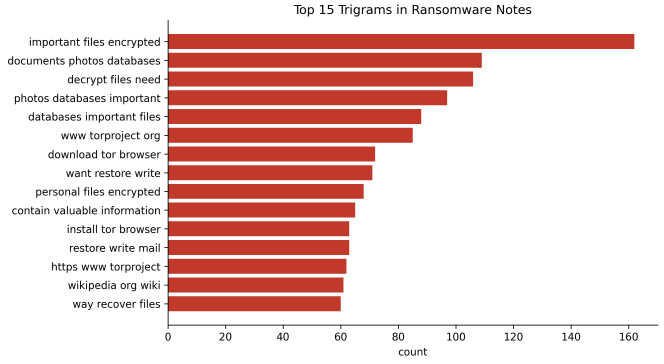


Fig. 5. Top 15 trigrams in ransomware notes. Structured threats (*your files have been encrypted, the decryption password*) appear consistently.

D. Keyword Co-occurrence Network (Ransomware Bigrams)

The co-occurrence analysis of keyword bigrams in ransomware notes reveals a tightly structured lexical system in which core concepts consistently cluster together. Central terms such as "files", "decrypt", "key", and "payment" form a dominant semantic nucleus that reflects the fundamental logic of a ransomware attack: "data are encrypted", "decryption requires a key", and "access is conditioned on payment". Additional clusters are organized around communication channels (e.g., references to email services) and cryptocurrency-based transactions (e.g., mentions of "Bitcoin", "acquisition" and "payment"). The overall density and cohesion of these clusters indicate a highly constrained and purpose-driven vocabulary. This pattern evidences a distinct "operational grammar" characteristic of ransom notes, reinforcing the RAND framework's capacity to identify and discriminate ransomware variants through their unique linguistic signatures.

VI. RAND APPLICATIONS

The proposed framework enables the extraction of stable linguistic and structural indicators from ransom notes that, by complementing traditional technical signals, support both early-stage detection and more precise attribution of malicious campaigns within the ransomware ecosystem.

A. Early Detection and Novel Campaign Identification

Early detection and the identification of emerging campaigns constitute the most immediate applications of the RAND framework. When a new ransomware variant appears, its binary may be entirely unfamiliar to existing security systems, perhaps employing an unseen encryption scheme or leveraging a zero-day vulnerability for initial access. Despite these technical innovations, the attackers are still compelled to leave a ransom note, and this unavoidable artifact provides a stable point of analysis from which early signals of a novel campaign can be extracted, even before traditional indicators become available.

B. Clustering and Attribution of Attacks

Clustering and attribution of attacks constitute another domain in which the RAND framework proves highly effective. Within threat intelligence, a recurring challenge is determining whether two seemingly unrelated incidents originate from the same threat actor. From a purely technical perspective, the samples involved may differ substantially—employing distinct packers, modified code sections, or other alterations designed to hinder classification—leading automated systems to label them as separate families. However, a detailed examination of the ransom notes can uncover revealing connections: recurring linguistic patterns, consistent communicative structures, or characteristic rhetorical choices that often remain stable even when the underlying malware has been altered. These elements enable the identification of links between campaigns and actors with a level of precision that surpasses the inherent limitations of technical analysis alone, thereby strengthening attribution efforts and contributing to a more robust foundation for strategic intelligence.

C. Forecasting Emerging Extortion Strategies

Forecasting emerging extortion strategies is another area in which the RAND framework provides clear analytical advantages. Ransomware evolves rapidly, yet its early strategic shifts are rarely visible in the technical artifacts of the malware. The move from single to double extortion, and later to triple extortion through added DDoS pressure, illustrates how threat actors refine coercive tactics while binaries and infection methods still resemble earlier variants, limiting the effectiveness of purely technical detection. Ransom notes, however, often reveal these changes at the outset. Because attackers must communicate demands and outline consequences, new layers of pressure and shifts in tone appear in these messages before they surface in code or infrastructure. By analysing such signals, the RAND framework allows early identification of emerging extortion models, giving defenders a chance to anticipate trends and strengthen their strategic readiness.

VII. Methodology for Dataset Curation and Model Building

A. Structural Signals and Class Separability

To motivate deployable detectors, we quantify structural markers (URLs/emails, punctuation and capitalization emphasis, entropy) and show linear separability in a reduced space (Figs. 6-8).

B. Presence of URLs and Emails in Notes

Figure 6 presents a bar chart comparing the presence of URLs and email addresses in ransomware notes versus benign texts, showing a markedly higher count of both elements in the malicious samples. This difference is expected, as ransomware operations require victims to reach the attackers or access a designated payment portal—often a .onion site—to recover their data or avoid further harm. Benign texts, by contrast, seldom include such direct and prescriptive contact channels. This quantitative disparity makes the presence and frequency of URLs and email addresses a highly discriminative feature within the RAND framework, offering a strong indicator for identifying malicious intent.

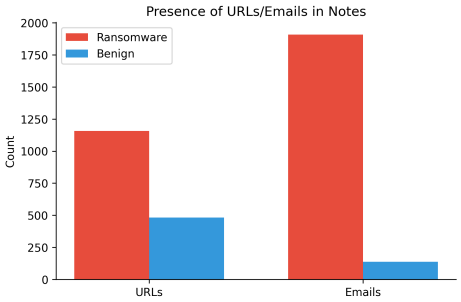


Fig. 6. Presence of URLs and emails. Ransomware notes frequently embed contact details; benign texts rarely do so.

C. ALL CAPS Word Ratio per Note

Ransomware notes exhibit a significantly higher proportion of ALL CAPS words compared to benign texts, a stylistic choice that functions as a visual emphasis mechanism. In typography and digital communication, text set entirely in uppercase often conveys heightened emotional intensity or urgency, a convention that emerged with early online etiquette and persists in message design for its attention-grabbing effect (e.g., “IMPORTANT!”, “WARNING!”, “READ THIS CAREFULLY!”). Studies in readability have shown that all-caps text alters the visual processing of words by removing distinctive ascenders and descenders, which can slow reading speed and increase perceptual effort, thereby magnifying the prominence of the emphasized content. Empirical work on message framing has also demonstrated that the presence of capitalized words—as one form of typographic emphasis—can influence recipients perceptions of urgency and importance in health communication contexts. In contrast, benign content rarely employs excessive capitalization outside proper nouns or acronyms, resulting in more naturalistic and varied sentence case usage. Within ransom notes, exaggerated ALL CAPS usage functions as a psychological and visual cue that intensifies perceived threat and command presence, and constitutes a quantifiable stylistic feature leveraged by the RAND framework to differentiate malicious extortion text from typical non-malicious writing.

D. Shannon Entropy Distribution of Notes

The Shannon entropy distribution of ransomware notes, shown in Figure 7, characterizes the extent to which their language is predictable and repetitive. Consistently lower entropy relative to benign texts indicates reduced lexical and structural variability—an effect that is expected when messages are highly scripted, task-oriented, and optimized for unambiguous instruction (e.g., payment steps, deadlines, contact channels). In practical terms, such notes concentrate probability mass on a narrow set of recurring tokens and short phrases, yielding higher redundancy and lower uncertainty per symbol than heterogeneous benign corpora. This interpretation aligns with the information-theoretic view of entropy as a measure of predictability and redundancy in natural language[13]. Consequently, entropy functions as a compact, model-agnostic statistic that complements higher-level lexical and syntactic features in the RAND framework, improving separability between extortion-driven templates and organically produced text with broader topical and stylistic dispersion.

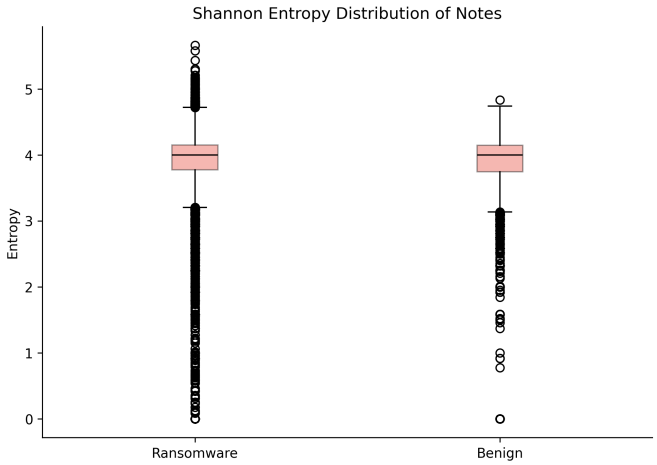


Fig. 7. Shannon entropy distribution. Ransom notes show lower entropy (formulaic repetition), while benign texts are more diverse.

E. 2D Separation (TruncatedSVD on TF-IDF)

Figure 8 illustrates the 2D separation achieved by applying TruncatedSVD (a dimensionality reduction technique similar to PCA) to TF-IDF features of ransomware and benign notes. The plot clearly shows distinct clustering, with ransomware notes (red points) forming a separate and relatively tight cluster from benign texts (blue points). This visual separation is a powerful validation of the RAND framework’s underlying premise: despite linguistic variations, ransomware notes possess a stable and unique linguistic signature that differentiates them from general text. The effectiveness of this separation in a low-dimensional space demonstrates that machine learning models can readily leverage these features for highly accurate classification. This finding is crucial for building deployable detectors that can reliably identify ransomware notes based on their textual content alone.

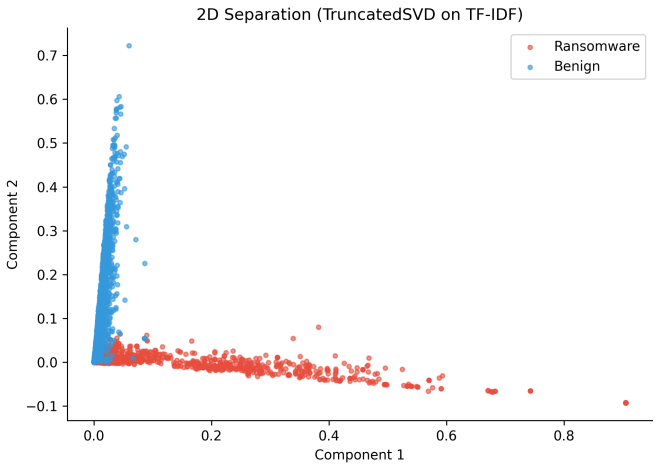


Fig. 8. 2D separation (TruncatedSVD/PCA on TF–IDF). Distinct clustering demonstrates ransom notes' stable linguistic signature suitable for ML-based detection.

VIII. Limitations and Ethical Considerations

No methodology is without limitations.

- **Copycat Notes:** Less sophisticated actors may copy the note of a successful group, leading to false attributions. However, the operational details (wallet addresses, URLs) would be different, and this itself is a useful intelligence point.
- **Translation and Localization:** Groups operating in different regions may translate their notes, potentially altering linguistic fingerprints. Analysis may need to focus on operational features for these cases.
- **Data Availability:** Building a comprehensive corpus requires collaboration between industry, academia, and law enforcement to share data responsibly.

Ethically, this research uses only publicly available or properly identified data. The purpose is purely defensive: to enhance the ability of organizations to protect themselves from criminal activity. The methodology does not involve engaging with threat actors or making payments.

IX. Conclusion and Future Work

The persistent evolution of ransomware demands a widening of our defensive outlook, compelling us to look beyond purely technical evidence. This paper has demonstrated that concentrating solely on binaries, exploits, and system behaviours overlooks one of the most consistent and revealing elements of an attack: the ransom note. As the only component crafted explicitly for human consumption, it offers an informative view into the attacker's methods, intentions, and operational culture, while remaining far less vulnerable to the forms of evasion that complicate traditional malware analysis. The RAND framework introduced here establishes a new, structured approach to converting ransom notes into a coherent source of intelligence. Its potential applications in early detection, campaign clustering, and the anticipation of evolving extortion practices provide a meaningful supplement to conventional defensive tools, enhancing both analytical depth and practical readiness. Looking ahead, future research will focus on deploying the RAND framework at scale, broadening the repository of annotated notes, and constructing production-grade machine learning models capable of integration into real-world security systems. By examining the language and communicative strategies of adversaries, defenders can move toward a more anticipatory posture, strengthening proactive defences against the continuing and pervasive threat of ransomware.

Acknowledgments

This work was supported by the Spanish Nuclear Safety Council (CSN) through the project "CIBER-CLEAR" and by the Spanish Ministry of Science and Innovation through the project PID2021-125962OB-C31 "SECURING/CYBER" and PID2024-156914OB-C41 "SAFE/CYBER". Additional funding was provided by the ARTEMISA International Chair in Cybersecurity (C057/23) and the DANGER Strategic Project of Cybersecurity (C062/23), both funded by the Spanish National Institute of Cybersecurity through the European Union NextGenerationEU and the Recovery, Transformation, and Resilience Plan.

References

[1] Scrivens, Ryan, Thomas W. Wojciechowski, Joshua D. Freilich, Steven M. Chermak, and Richard Frank. "Comparing the online posting behaviors of violent and non-violent right-wing extremists." Terrorism and political violence 35, no. 1 (2023): 192-209.

[2] Cybersecurity and Infrastructure Security Agency (CISA), *Stop Ransomware Guide*, 2023.

[3] Al-Dwairi, Mahmoud, Ahmed S. Shatnawi, Osama Al-Khaleel, and Basheer Al-Duwairi. "Ransomware-resilient self-healing XML documents." Future Internet 14, no. 4 (2022): 115.

[4] S. Homayoun et al., "Know Abnormal, Find Evil: Frequent Pattern Mining for Ransomware Threat Hunting and Intelligence," *IEEE Trans. Emerg. Topics Comput.*, vol. 8, no. 2, pp. 341–351, Apr. 2017.

[5] Haner, Murat, Melissa M. Sloan, Amanda Graham, Justin T. Pickett, and Francis T. Cullen. "Ransomware and the Robin Hood effect?: Experimental evidence on Americans' willingness to support cyber-extortion." Journal of Experimental Criminology 19, no. 4 (2023): 943-970.

[6] Berardi, Davide, Saverio Giallorenzo, Andrea Melis, Simone Melloni, Loris Onori, and Marco Prandini. "Data flooding against ransomware: Concepts and implementations." Computers & Security 131 (2023): 103295.

[7] Noman, Haitham Ameen, Qusay Al-Maatouk, and Sinan Ameen Noman. "A static analysis tool for malware detection." In 2021 International Conference on Data Analytics for Business and Industry (ICDABI), pp. 661-665. IEEE, 2021.

[8] İbrahim, Mülhem, Bayan Issa, and Muhammed Basheer Jasser. "A method for automatic android malware detection based on static analysis and deep learning." IEEE Access 10 (2022): 117334-117352.

[9] Johnson, Shanoop, R. Gowtham, and Anand R. Nair. "Ensemble model ransomware classification: A static analysis-based approach." In Inventive Computation and Information Technologies: Proceedings of ICICIT 2021, pp. 153-167. Singapore: Springer Nature Singapore, 2022.

[10] Gulmez, Sibel, Arzu Gorgulu Kakisim, and Ibrahim Sogukpinar. "XRan: Explainable deep learning-based ransomware detection using dynamic analysis." Computers & Security 139 (2024): 103703.

[11] Usharani, S., P. Manju Bala, and M. Martina Jose Mary. "Dynamic analysis on crypto-ransomware by using machine learning: Gandcrab ransomware." In Journal of Physics: Conference Series, vol. 1717, no. 1, p. 012024. IOP Publishing, 2021.

[12] Sharma, Purushottam, Shaurya Kapoor, and Richa Sharma. "Ransomware detection, prevention and protection in IoT devices using ML techniques based on dynamic analysis approach." International Journal of System Assurance Engineering and Management 14, no. 1 (2023): 287-296.

[13] Shannon, C. E. "Prediction and entropy of printed English". Bell System Technical Journal, (1951), 30(1), 50-64.

Transformers borgeanos: inyectividad y confidencialidad en modelos de lenguaje

Mariano Caruso*
Fundación I+D del Software Libre, FIDESOL
Universidad Internacional de La Rioja, UNIR
Universidad de Granada, UGR

***Resumen*—Este artículo analiza las propiedades funcionales de las arquitecturas Transformer de tipo decoder-only autorregresivas que, dada su integración ubicua y adopción masiva incluyendo entornos no especializados, presentan implicaciones críticas para la privacidad. Investigaciones recientes demuestran que, bajo ciertas hipótesis generales, la función que mapea una secuencia finita de *tokens* hacia su estado oculto en la última capa es casi siempre inyectiva. Esto implica que las representaciones internas del modelo no constituyen una reducción semántica del *prompt*, sino un rastro invertible que permite reconstruir el texto original mediante algoritmos eficientes. Esto obliga a abandonar la interpretación habitual de los modelos de lenguaje como sistemas de abstracción o compresión de la entrada, y a reconsiderarlos como mecanismos de preservación exacta de la información.**

***Index Terms*—Transformers, inyectividad, privacidad.**

I. Introducción

El proceso inicial mediante el cual un Modelo de Lenguaje Grande (LLM) toma como entrada un *prompt* y lo codifica en un estado oculto, que finalmente será utilizado para definir la salida, está constituido por múltiples etapas: tokenización, proyección a embeddings, codificación posicional y un stack de capas que combinan mecanismos de atención multicabeza y redes feedforward con conexiones residuales y normalización. Este flujo arquitectónico fue introducido y formalizado en el trabajo seminal [1], denominado genéricamente como Transformer.

Este proceso de codificación del *prompt* en el *hidden state* puede imaginarse como el recorrido de un viajero hipotético por un terreno a priori desconocido. Para ir de un lugar conocido (*prompt*) a otro cuya ubicación exacta se ignora (*hidden state*) el viajero debe pedir indicaciones repetidamente, en distintos puntos del trayecto. Las directrices que recibe pueden parecer arbitrarias o caprichosas desde fuera, y a menudo son difíciles de memorizar. Todo este viaje constituye, en sí mismo, una travesía al interior de un laberinto con múltiples entradas, tantas como *prompts* diferentes puedan construirse a partir de un vocabulario finito y una ventana de contexto finita.

Surge una pregunta para el viajero: ¿qué forma adopta este laberinto? ¿Es una estructura de ramificaciones abiertas o converge inevitablemente hacia puntos comunes? Durante mucho tiempo prevaleció la segunda interpretación: se asumía que el laberinto funcionaba como un embudo compresor, en el que múltiples caminos (correspondientes a entradas distintas) convergían hacia un mismo punto final, descartando en el trayecto los detalles irrelevantes del origen para retener únicamente la esencia semántica. Bajo esta visión, los *hidden states* se concebían como representaciones empobrecidas o altamente destiladas del *prompt*.

Recientemente se ha demostrado [2] que para una gran clase de modelos de lenguaje de uso masivo tipo Transformer la función que toma la entrada del usuario y la codifica como un estado interno es casi siempre inyectiva y, por tanto, invertible. Este resultado invita a revisar un supuesto ampliamente extendido: que los estados ocultos surgen de una abstracción que comprime, resume o destila la información de la entrada y que obtener la entrada a partir de la salida era un procedimiento imposible o ineficiente. Según este resultado, los *hidden states* no constituyen representaciones empobrecidas del *prompt*, sino "transformaciones que no olvidan", en el sentido en que podemos volver a la entrada. Este no es un artefacto añadido, sino una consecuencia directa de la propia arquitectura de tales LLMs.

Esta demostración obliga a abandonar la idea de que se trata de un sistema cuyos estados ocultos proceden de una abstracción de la entrada que resume o comprime su información y verla, bajo una lente de conceptos borgeanos: la ausencia de pérdida informacional en un mapa que aspira a ser idéntico al territorio [3], la posibilidad de desandar el camino recorrido en el laberinto para regresar a la entrada [4] y la fidelidad patológica del recuerdo del viajero [5]. Más formalmente, de procesos uno a uno, invertibles y sin pérdidas de información.

Tratar estos estados ocultos como representaciones inocuas y efímeras —como meros vectores transitorios de procesamiento— resulta, por tanto, profundamente engañoso. Su carácter casi perfectamente reconstructivo implica que cualquier información sensible introducida en un *prompt* (datos personales, secretos comerciales, información médica o jurídica) puede, en principio, ser recuperada por quien tenga acceso al estado oculto y al algoritmo de inversión adecuado. Esta memoria perfecta ramifica en consecuencias graves para la confidencialidad de los datos, el cumplimiento de normativas de protección y la viabilidad de auditorías reales de lo que los modelos *olvidan*. En última instancia, pone de manifiesto la necesidad urgente de desarrollar nuevas arquitecturas de seguridad que se integren en las arquitecturas de tales LLMs —tales como ofuscación intencional de representaciones, mecanismos de olvido controlado, encriptación homomórfica en la capa latente o límites estrictos en la accesibilidad a estados intermedios— y que mitiguen el riesgo inherente a esta invertibilidad casi universal en los despliegues actuales de LLMs. O bien concebir arquitecturas que sean seguras por diseño.

II. Propiedades funcionales

Se dará un recordatorio breve sobre algunas propiedades generales que se derivan de la inyectividad de una función general $f : X \longrightarrow Y$. Para fijar ideas sea X el conjunto discreto que contiene todas las entradas, o *prompts* que pueden construirse con un vocabulario finito y una ventana de contexto finita y sea Y el conjunto denso en el que viven las representaciones internas o *hidden states*.

II-A. Inyectividad

Una función $f : X \to Y$ es inyectiva si $\forall x_1, x_2 \in X$: $x_1 \neq x_2 \implies f(x_1) \neq f(x_2)$, dos entradas distintas no producen resultados indistinguibles. De hecho, para una función inyectiva no existen entradas distintas que generen la misma salida. Esto puede probarse por reducción al absurdo: supongamos que $x_1, x_2 \in X$ satisfacen que $x_1 \neq x_2$ y dan lugar a $f(x_1) = f(x_2)$. Por el contrarrecíproco de la definición de inyectividad esto implica $x_1 = x_2$ contradiciendo la suposición inicial, entonces $\neg\exists x_1, x_2 \in X | x_1 \neq x_2 \wedge f(x_1) = f(x_2)$. Equivalentemente, f es inyectiva si y solo si para cada $y \in \mathtt{Im}(f) \subseteq Y$ existe un único $x \in X$: $f(x) = y$.

II-B. Invertibilidad por izquierda

Una *inversa izquierda* o *retracción* de $f : X \longrightarrow Y$ es una función $g : Y \longrightarrow X$ tal que su composición por la izquierda da la función identidad: $g \circ f = \mathtt{id}_X$, es decir, si $f(x) = y$, entonces $g(y) = x$. La función g debe coincidir con la inversa de f sobre la imagen de f, pero puede diferir para los elementos de Y que no estén en la imagen de f, pues $\mathtt{Im}(f) \subseteq Y$. De hecho, en caso de que exista la inversa de f no necesariamente coincide con g, pues la composición $f \circ g$ no está garantizada que sea igual a $\mathtt{id}_Y$.

II-C. Inyectividad e invertibilidad por izquierda

Una afirmación importante que vincula los conceptos anteriores: toda función f de dominio no vacío e inyectiva admite una inversa a izquierda g. Una definición formal de g, puede hacerse tomando un $x' \in X$ de manera que: $g(y) = x$ si $y = f(x)$, pues en particular $y \in \mathtt{Im}(f)$, y $g(y) = x'$ si $y \notin \mathtt{Im}(f)$. El hecho de que f sea inyectiva garantiza que g esté bien definida como función, i.e. que para cada $y \in Y$ exista un único x tal que $g(y) = x$. Probar que efectivamente g está bien definida, no nos da una demostración constructiva de g y la inversa a izquierda no es única, incluso bajo condiciones de inyectividad de f. Dicha arbitrariedad en tales inversas proviene esencialmente del hecho de que hay elementos que están en Y y que no están en $\mathtt{Im}(f)$.

La siguiente sección pretende descomponer el proceso en etapas mediante el cual la entrada se codifica en un estado oculto, a fin de caracterizar la función f, que es objeto de este artículo.

III. Localización del problema

Es imperativo identificar cómo es que aquel viajero adquiere una forma de volver a la misma entrada del laberinto. Para ello, es necesario diseccionar la parte de la arquitectura del Transformer involucrada. Según [2], trataremos con el pipeline de los modelos decoder-only y autorregresivos desde la entrada hasta la representación oculta del último token.

III-A. Etapas involucradas

Consideremos un modelo de lenguaje como una composición de funciones que transforman una secuencia discreta de entrada en una distribución de probabilidad, pasando por un espacio continuo latente. Sea V el vocabulario finito y $X = V^{\leq K}$ el conjunto de todas las posibles secuencias de entrada (*prompts*), donde K es el tamaño máximo de la ventana de contexto.

El proceso de este tipo de modelos se divide en las siguientes etapas. La primera etapa es la **tokenización**, que consiste en un mapeo discreto $\tau : V^{\leq K} \to \{1, \ldots, |V|\}^T$, donde $T \leq K$, T denota la longitud efectiva de la secuencia x. Esta etapa fija el dominio discreto sobre el que se plantea el problema de inyectividad, pero no introduce todavía estructura continua. La segunda etapa corresponde a la **incrustación**, que incluye tanto los embeddings de tokens como las codificaciones posicionales [1]. Formalmente, esta etapa define un mapeo $\mathrm{Emb} : \{1, \ldots, |V|\}^T \to \mathbb{R}^{T \times d}$, donde d corresponde a la dimensión de las matrices de incrustación. Aquí se produce la primera inmersión del dominio discreto en un espacio vectorial continuo. La tercera etapa es la de **transformación** y está formada por la pila de bloques Transformer, cada uno compuesto por normalización [6], atención causal multi-cabeza, redes feedforward y conexiones residuales. La composición de L bloques define un mapeo $\mathcal{T} : \mathbb{R}^{T \times d} \to \mathbb{R}^{T \times d}$, que actúa sobre todos los estados ocultos. Finalmente, la cuarta etapa es la capa de salida, es la de **proyección** que toma el último estado oculto y devuelve los logits sobre el vocabulario, usando una función `softmax`. Esta etapa se puede describir como un mapeo afín seguido de una normalización, pero el artículo no la analiza, ya que el objeto de estudio es la representación oculta previa a esta capa. Denotamos por $\Theta \subseteq \mathbb{R}^p$ el espacio de parámetros del modelo (pesos y sesgos) y por $\rho(x; \theta) \in \mathbb{R}^d$ la representación oculta asociada al último token tras procesar la secuencia $x \in V^{\leq K}$ con parámetros θ. La función cuya inyectividad se estudia es, por tanto, la composición $x \longmapsto \rho(x; \theta)$, donde $\rho(x; \theta) \in \mathbb{R}^d$ se obtiene seleccionando la fila correspondiente al último token tras aplicar embeddings y bloques Transformer. Nos enfocamos en la representación oculta final del último token ρ ya que desde ella se realiza la predicción del siguiente token, las filas anteriores importan solo en la medida en que dieron forma a ρ. En otras palabras, el estado del último token es el único que ha tenido la oportunidad de atender a todos los tokens anteriores de la secuencia. Si la función $x \longmapsto \rho(x; \theta)$ resulta ser inyectiva, significa que toda la *historia del prompt* está codificada en ese único punto del espacio latente.

Veamos algunos detalles sobre la noción de inyectividad casi segura. Diremos que existe una *colisión* si existen $x \neq x'$ tales que $\rho(x; \theta) = \rho(x'; \theta)$. Definimos la función de discrepancia entre dos entradas x, x' distintas

$$h_{x,x'}(\theta) = \|\rho(x; \theta) - \rho(x'; \theta)\|_2^2. \tag{1}$$

una colisión ocurre si y solo si $h_{x,x'}(\theta)$.

Hemos comentado que los componentes estándar de un Transformer: producto punto para las capas de atención, capas feed-forward (lineales y activaciones suaves como GELU o Swish), normalizaciones tipo LayerNorm y función `softmax`, son composiciones de funciones reales analíticas salvo en

casos de activaciones a trozos, e.g. ReLU. Cuando la composición resulta analítica real, la función $h_{x,x'}(\theta)$ es analítica real respecto a θ. Por el teorema clásico sobre conjuntos de ceros de funciones analíticas, si $h_{x,x'}(\theta)$ no es idénticamente nula entonces el conjunto $\mathscr{C}_{x,x'} = \{\theta \in \Theta : h_{x,x'}(\theta) = 0\}$, tiene medida de Lebesgue nula [7], [8]. Geométricamente, $\mathscr{C}_{x,x'}$ es una subvariedad de dimensión estrictamente menor dentro de Θ: en un volumen de parámetros $p-$dimensional, los parámetros que causan colisiones ocupan una estructura de menor dimensión (una superficie, una curva, etc.), es decir, son estadísticamente insignificantes.

Si inicializamos θ mediante una distribución continua (p. ej. Gaussiana, inicialización de Xavier/Glorot o Kaiming), la probabilidad de muestrear exactamente un punto en $\mathscr{C}_{x,x'}$ es cero. Extendiendo la consideración a todos los pares $x \neq x'$, el conjunto de parámetros que provoca cualquier colisión es aun así de medida nula, por lo que el modelo es inyectivo con probabilidad uno en la inicialización. Una preocupación natural es si el entrenamiento, e.g. descenso por gradiente estocástico (SGD), podría llevar los parámetros desde una región de medida plena hacia el conjunto de colisiones. Bajo condiciones estándares, las actualizaciones de parámetros inducidas por SGD son transformaciones suaves (flujos generados por campos vectoriales diferenciables) que no colapsan volúmenes de dimensión plena en subvariedades de dimensión inferior; en términos de medida, los difeomorfismos locales preservan la nulidad de conjuntos de medida. Por tanto, si θ inicia fuera del conjunto de colisión, permanecerá fuera con probabilidad uno a lo largo del entrenamiento. La inyectividad garantiza existencia y unicidad (casi segura) de la preimagen de una representación, pero no su recuperabilidad práctica. En la siguiente sección veremos SIPIT es un algoritmo que explota la causalidad autorregresiva de modelos decoder-only y la discreción del vocabulario para recuperar la secuencia de entrada x. En resumen,

III-B. Algoritmo SIPIT

Veamos ahora los detalles del algoritmo SIPIT (Search-based Iterative Prompt Inversion Transformer) presentado en [2]. En un Transformer decoder-only la representación en la posición k, notada $\rho_k(x;\theta)$, depende exclusivamente de manera recursiva de los anteriores: $x_{\leq k}$. Esto permite descomponer la inversión global en un número T de subproblemas secuenciales. El procedimiento ingenuo que evalúa todas las secuencias posibles tiene complejidad exponencial $|V|^T$. SIPIT transforma este coste en $O(T \cdot |V|)$ aplicando búsqueda sobre tokens posición a posición y reutilizando activaciones previas mediante KV-caching.

Descripción del algoritmo.

1. **Entrada:** representación objetivo $\rho(x;\theta)$, vocabulario V, modelo con parámetros θ, longitud T (si es desconocida, usar criterio de parada).
2. **Inicialización del prefijo:** para $k = 1$, para cada $v \in V$ se calcula la representación candidata $\rho_1((v);\theta)$ (o su proyección relevante) y se selecciona

$$\hat{x}_1 = \arg\min_{v \in V} \|\rho_1((v);\theta) - \rho_1(x;\theta)\|_2.$$

   donde (v) representa la secuencia formada únicamente por el token v.

3. **Reconstrucción iterativa:** definiendo la secuencia de orden $k-1$ como $\hat{x}_{<k}=(\hat{x}_1, \cdots, \hat{x}_{k-1})$ recuperado, para cada candidato $v \in V$ se forma el candidato temporal de tamaño k: $(\hat{x}_{<k}, v) = (\hat{x}_1, \cdots, \hat{x}_{k-1}, v)$. Usando KV-cache no es necesario recomputar las activaciones de $\hat{x}_{<k}$; únicamente se evalúa el impacto del token v en la activación de la posición k, $\rho_k((\hat{x}_{<k}, v);\theta)$. Se selecciona

$$\hat{x}_k = \arg\min_{v \in V} \|\rho_k((\hat{x}_{<k}, v);\theta) - \rho_k(x;\theta)\|_2.$$

4. **Repetir hasta:** $k = T$ o hasta criterio de parada.

Aunque $O(T \cdot |V|)$ es una mejora dramática, si $|V|$ es muy grande, tomando 128K como referencia de modelos como GPT-4 y Llama 3, el coste sigue siendo significativo. Por tanto se proponen algunas mejoras:

- **Búsqueda jerárquica:** primero filtrar candidatos mediante una búsqueda rápida en el espacio de embeddings (ANN — approximate nearest neighbours) para obtener un conjunto top-k de candidatos promisorios, y luego evaluar solo esos en las capas profundas.
- **Filtrado por proximidad semántica:** aprovechar que tokens semánticamente cercanos tienen embeddings próximos para priorizar candidatos.
- **Heurísticas de poda y beam search adaptado:** combinar la inversión exacta con estrategias de poda que reduzcan branching en prefijos improbables.

Con estas mejoras se consigue una complejidad temporal $O(T \cdot \log |V|)$, permitiendo reconstruir textos largos de manera exacta y eficiente.

Los métodos de inversión que optimizan entradas continuas por gradiente y luego proyectan al vocabulario suelen fallar por la no convexidad de la superficie de pérdida, la multiplicidad de mínimos locales y los errores introducidos por la discretización final. SIPIT, al operar directamente en el dominio discreto y verificar candidatos contra la representación objetivo, evita esas fuentes de error y proporciona una recuperación exacta bajo la hipótesis de inyectividad casi segura.

En términos de amenaza, SIPIT tiene implicaciones relevantes puesto que no requiere acceso a gradientes ni a mecanismos de entrenamiento; basta con acceso a la función de evaluación (forward pass) y a las activaciones intermedias o finales. Esto lo hace viable en escenarios de *caja gris* (p. ej. APIs que devuelvan embeddings). Permite exfiltrar prompts de sistema o datos sensibles en contexto, independientemente de la longitud, siempre que se pueda obtener la representación objetivo y ejecutar forward passes con candidatos. Las mitigaciones prácticas deben considerar restringir el acceso a embeddings/activaciones, añadir ruido diferencial a las representaciones, o diseñar mecanismos de cribado que impidan el uso de la representación como un "hashreversible del prompt.

La inyectividad casi segura derivada de la analiticidad de los componentes del Transformer, junto con la preservación de esta propiedad bajo inicialización continua y entrenamiento suave, establece que las representaciones internas no son meros resúmenes con pérdida sino codificaciones reversibles en sentido probabilístico. SIPIT convierte esa observación teórica en una herramienta práctica de inversión, con complejidad tratable gracias a la causalidad autorregresiva y al KV-caching,

y con variantes optimizadas que reducen la dependencia del tamaño del vocabulario. Desde la perspectiva de privacidad y seguridad, la combinación de garantía teórica y viabilidad algorítmica exige replantear qué tipo de salidas (embeddings, activaciones) deben exponerse por API y qué mecanismos de defensa se implementan.

De esta forma, no solo se tiene la existencia de una inversa a izquierda sino una prueba constructiva a través de algoritmo SIPIT [2], que explota la estructura causal del Transformer.

Este resultado es agnóstico al entrenamiento y depende puramente de la arquitectura. Cubre la inmensa mayoría de LLMs comerciales y *open source* actuales tipo *decoder-only*, incluyendo: GPT (OpenAI): modelos autorregresivos estándar, Llama y Mistral (Meta/Mistral AI): arquitecturas que usan variantes como RMSNorm o RoPE, pero mantienen la estructura residual fundamental y la atención causal; y Gemma y Falcon: modelos con variaciones menores en la posición de la normalización (Pre-Norm/Post-Norm) que no afectan la propiedad de inyectividad casi segura.

IV. Problemas de confidencialidad

La implementación de una función inversa que puede computarse de manera eficiente por un algoritmo $\mathcal{A}$, tal que para cada estado interno $f(x)$ se cumple que $\mathcal{A}(f(x))$ devuelve x para estos LLMs, invalida la premisa tradicional de seguridad por oscuridad en el espacio latente. El mapa del laberinto ya no es una aproximación borrosa. Quien posea el mapa (los pesos del modelo) y una posición final (un estado oculto), puede trazar matemáticamente la única línea ininterrumpida que lleva al punto de partida de manera eficiente. Esto transforma radicalmente la evaluación de riesgos de confidencialidad en sistemas de inteligencia artificial. Aunque estos ataques requieren capacidades técnicas elevadas (como acceso a activaciones internas, a la arquitectura exacta del modelo o a infraestructuras compartidas), su viabilidad no es puramente teórica, como muestran tanto los resultados constructivos recientes como incidentes documentados en sistemas distribuidos de inferencia.

IV-A. Generalización sin compresión informacional

Los modelos no comprimen ni olvidan información en sus representaciones internas: la conservan con fidelidad biyectiva. La supuesta "pérdida" o "resumen" que ocurre al salir de la ventana de contexto o al no ser priorizada por la atención es solo un fenómeno de la generación de salida, no de las capas internas [9], [10].

En términos borgeanos, el Transformer padece del mal de Funes: recuerda cada detalle con precisión enfermiza. Cualquier secreto (contraseñas, claves API, datos médicos, PII, información comercial sensible) procesado por el modelo permanece latente e íntegro en sus activaciones, esperando ser recuperado de forma determinista y eficiente mediante algoritmos de inversión como SIPIT [2].

IV-B. Nuevas superficies de ataque

La inyectividad genera vectores de ataque inéditos o dramáticamente ampliados:

- **Interceptación de activaciones internas:** un atacante que logre acceder a estados ocultos (mediante ataques man-in-the-middle en APIs de inferencia, inferencia distribuida split, o side channels) puede reconstruir el *prompt* completo en texto plano.
- **Filtración en entornos multi-tenant:** en servicios de inferencia compartida (vLLM, TGI, etc.), un fallo de aislamiento de memoria GPU (rowhammer, Spectre-like, errores de scheduling) permite leer las activaciones de otros usuarios, revelando ahora conversaciones privadas completas [11].
- **Persistencia en cachés y logs:** los KV-caches, bases de datos vectoriales (RAG), logs de depuración y trazas de ingeniería que almacenan estados intermedios se convierten en repositorios de datos sensibles en formato reversible, a menudo sin las protecciones aplicadas a los datos brutos.

IV-C. Implicaciones normativas y transparencia forense

Los estados ocultos dejan de ser meras representaciones internas opacas para convertirse legalmente en datos personales como información de salud protegida. Las empresas que almacenan embeddings, activaciones o cachés sin cifrado fuerte, asumiendo que son anónimos o irreversibles, podrían estar violando masivamente normativas de privacidad [12], [13], [14]. La vectorización ya no constituye anonimización efectiva.

Irónicamente, la misma propiedad que representa un riesgo masivo de privacidad permite una auditoría forense de precisión matemática. Ante generación de contenido tóxico, alucinaciones dañinas o salidas no deseadas, es posible rastrear exactamente qué fragmento del *prompt* o contexto recuperado activó la respuesta problemática, eliminando la ambigüedad de la caja negra [15].

V. Recomendaciones

Ante la evidencia de que el laberinto es reversible y el mapa es escala 1:1, las estrategias de seguridad deben abandonar la confianza en la opacidad del espacio latente y tratar los estados internos con la misma criticidad que los datos de entrada originales. Veamos una serie de recomendaciones en tres grandes niveles.

V-A. Arquitectura y entrenamiento

- **Ofuscación por ruido (privacidad diferencial):** inyectar perturbaciones estocásticas controladas, ruido gaussiano o Laplaciano, en los *embeddings* iniciales o en estados latentes durante entrenamiento o inferencia. Esto rompe la inyectividad estricta, convirtiendo el mapeo en una función probabilística que protege la identidad del *prompt* sin degradar excesivamente la utilidad [10].
- **Compresión estructural obligatoria:** investigar e implementar cuellos de botella de información reales que fuercen la pérdida irreversible de detalles superficiales, reteniendo únicamente representaciones semánticas abstractas.

V-B. Infraestructura y despliegue

- **Equiparación de clasificación de seguridad:** los vectores de activación, *embeddings* y KV-cachés deben recibir la misma clasificación de sensibilidad que los datos de entrada en texto plano.

- **Cifrado obligatorio en reposo y tránsito:** estados internos persistentes (bases vectoriales, cachés en disco) y comunicaciones entre componentes distribuidos deben cifrarse con estándares fuertes. Utilizar computación confidencial cuando sea posible.
- **Aislamiento estricto y borrado criptográfico:** en entornos multi-tenant, garantizar aislamiento de memoria GPU entre sesiones y borrado seguro de residuos de contexto entre peticiones de distintos usuarios.
- **Control de acceso granular:** las APIs nunca deben exponer estados ocultos o *hidden states* completos en entornos de producción; solo el texto generado final, salvo en entornos de depuración estrictamente controlados.
- **Anonimización / saneamiento previo:** aplicar redacción de PII, tokenización segura y eliminación de información sensible *antes* de que el texto entre al modelo, ya que una vez procesado, la inyectividad garantiza su preservación.

V-C. *Operativo y de gobernanza*

- **Revisión de DPIA y políticas de retención:** actualizar evaluaciones de impacto en protección de datos asumiendo el peor caso: acceso total a activaciones = acceso total al *prompt*. Incluir explícitamente estados intermedios en políticas de "derecho al olvido".
- **Auditoría forense proactiva:** desarrollar y emplear herramientas basadas en algoritmos de inversión (tipo SIPIT) para *red teaming* y análisis post-incidente, convirtiendo la propiedad de invertibilidad en una ventaja defensiva verificable.
- **Gestión estricta del ciclo de vida de logs:** prohibir almacenamiento persistente innecesario de activaciones en producción; imponer TTL corto y acceso extremadamente restringido cuando sea imprescindible para depuración o cumplimiento.

VI. CONCLUSIONES

La demostración de la inyectividad obliga a abandonar la intuición de que los modelos de lenguaje operan mediante compresión semántica u olvido estructural: lo que parecía abstracción era, en realidad, una incapacidad para usar el mapa. Al igual que en *Del rigor en la ciencia* [3], el Transformer no produce un mapa simplificado del *territorio lingüístico*, sino una representación de escala 1:1, en la que todo accidente de la entrada queda rigurosamente cartografiado en el estado interno del modelo.

Esta propiedad transforma la interpretación del recorrido interno del modelo. El conjunto de capas de un LLM: tokenización, embeddings, mecanismo de atención, funciones de activación no lineales y normalizaciones, no actúan como un filtro que descarta información, sino como un intrincado laberinto que mantiene una relación de identidad unívoca con la entrada. Por complejo que sea el trayecto, siempre existe un hilo —materializado algorítmicamente en procedimientos como SIPIT— que permite desandar el camino desde el estado oculto hasta el *prompt* original. En este sentido, el Transformer se asemeja a *La casa de Asterión* [4]: un espacio del tamaño del mundo, donde el olvido no es una contingencia empírica, sino una imposibilidad técnica.

Las consecuencias son profundas: la generalización del modelo no emerge por supresión de detalles, sino por su codificación íntegra en un espacio de alta dimensión. Esta fidelidad estructural evoca inevitablemente la figura de *Funes el memorioso* [5], para quien toda diferencia es reconocible y ninguna variación resulta irrelevante. Los modelos de lenguaje modernos comparten esa carga: recuerdan incluso aquello que no necesitan.

El algoritmo SIPIT demuestra que esta propiedad no es meramente teórica. La posibilidad de reconstrucción exacta en tiempo lineal convierte a los estados ocultos en objetos informacionalmente densos, más cercanos a un registro reversible que a una abstracción segura. Esto invalida el supuesto anonimato de los *embeddings* y expone riesgos sustanciales en materia de privacidad, protección de datos y cumplimiento regulatorio en sistemas desplegados a gran escala.

Dado que este tipo de LLMs carece de mecanismos internos de olvido, la responsabilidad de garantizar la seguridad no puede recaer en la arquitectura estándar. Las estrategias de mitigación deben, por tanto, introducir el olvido de forma explícita, mediante modificaciones estructurales que rompan la reversibilidad del modelo y limiten la persistencia de la información. Desde esta perspectiva, diseñar sistemas de lenguaje seguros no implica ocultar el laberinto, sino decidir, deliberadamente, qué recorridos deben dejar de ser posibles. Algunos Transformers no olvidan.

AGRADECIMIENTOS

Esta investigación ha sido realizada en el contexto de los proyectos ECO − 20241014 | QUORUM y la Red de Excelencia Cervera para la investigación en tecnologías Quánticas para Infraestructura eXperimental Orientada a la soberanía Tecnológica en entornos Estratégicos CER − 20251020 | QUIXOTE, financiados por el Ministerio de Ciencia, Innovación y Universidades a través del CDTI.

REFERENCIAS

[1] A. Vaswani et al., "Attention is all you need", *Advances in Neural Information Processing Systems*, vol. 30, 2017.
[2] G. Nikolaou et al., "Language Models are Injective and Hence Invertible", *arXiv preprint arXiv:2510.15511*, 2025.
[3] J. L. Borges, "Del rigor en la ciencia", *El Hacedor*, 1960.
[4] J. L. Borges, "La casa de Asterión", *El Aleph*, 1949.
[5] J. L. Borges, "Funes el memorioso", *Ficciones*, 1944.
[6] J. L. Ba, J. R. Kiros y G. E. Hinton, "Layer normalization", *arXiv preprint arXiv:1607.06450*, 2016.
[7] W. Rudin, *Principles of Mathematical Analysis*, 2nd ed. New York, NY, USA: McGraw-Hill, 1964.
[8] B. S. Mityagin, "The zero set of a real analytic function", *arXiv preprint arXiv:1512.07276*, 2015.
[9] J. X. Morris et al., "Text embeddings reveal (almost) as much as text", *EMNLP*, 2023.
[10] J. Geiping et al., "Inverting gradients - how easy is it to break privacy in federated learning?", *NeurIPS*, 2020.
[11] A. Zou et al., "Universal and transferable adversarial attacks on aligned language models", *arXiv:2307.15043*, 2023.
[12] General Data Protection Regulation (GDPR), Regulation (EU) 2016679, 2016.
[13] EU Artificial Intelligence Act (EU AI Act), Regulation (EU) 20241689, 2024.
[14] Health Insurance Portability and Accountability Act (HIPAA), Pub. L. 104-191, 1996.
[15] A. Wei, N. Haghtalab y J. Steinhardt, "Jailbroken: How does llm safety training fail?", *NeurIPS*, 2023.

QZKP basada en Codificación Conjugada para la Autenticación de Identidad

Jorge Garcia-Diaz
Universidad de La Laguna
La Laguna, Tenerife
jgarcidi@ull.edu.es

Daniel Escanez-Exposito
Universidad de La Laguna
La Laguna, Tenerife
jescanez@ull.edu.es

Pino Caballero-Gil
Universidad de La Laguna
La Laguna, Tenerife
pcaballe@ull.edu.es

***Resumen*—Las Pruebas de Conocimiento Nulo fundamentadas en supuestos de dificultad computacional de ciertos problemas matemáticos constituyen primitivas esenciales para la autenticación segura de usuarios. Este artículo presenta una innovadora Prueba de Conocimiento Nulo que explota la aleatoriedad intrínseca de los bits cuánticos. En contraste con las construcciones tradicionales basados en problemas NP computacionalmente intratables, el protocolo sugerido fundamenta su seguridad en el estrato físico; en concreto, en el principio de incertidumbre de las proyecciones de estados cuánticos sobre bases de medición desalineadas. El análisis matemático demuestra las propiedades de completitud, solidez y conocimiento nulo del modelo. Asimismo, el rendimiento del protocolo se evalúa a través de un simulador cuántico bajo condiciones realistas susceptibles a fallos. Los hallazgos evidencian que el sistema es resiliente frente a participantes deshonestos, manteniéndose viable bajo recursos cuánticos restringidos, lo que proporciona un enfoque escalable para la autenticación cuántica segura.**

***Index Terms*—QZKP, QIA, QKD, ZKP**

I. Introducción

Una de las herramientas más eficaces para la autenticación de usuarios es la Prueba de Conocimiento Nulo (ZKP) [1]. Históricamente, estos mecanismos criptográficos se fundamentan en problemas matemáticos que se presumen computacionalmente intratables, catalogados habitualmente como problemas NP. No obstante, con el acelerado progreso de la computación cuántica, el horizonte de complejidad de dichas formulaciones ha sufrido alteraciones drásticas.

Ciertos dilemas, antes concebidos como altamente complejos, han experimentado una merma en su exigencia computacional a causa de algoritmos cuánticos como el algoritmo de búsqueda de Grover [2], en tanto que otros, como la factorización de enteros, han pasado a ser resolubles de manera eficiente a través del algoritmo de Shor [3], [4].

A pesar de que los esfuerzos se han orientado a explotar la computación cuántica para computar problemas matemáticos difíciles para ordenadores tradicionales [5], persiste una evidente escasez de literatura académica que indague en las capacidades de esta tecnología para concebir nuevos protocolos criptográficos. En rigor, dentro del ámbito cuántico, la inmensa mayoría de las aportaciones a la criptografía se han focalizado en la Distribución Cuántica de Claves (QKD). En particular, la bibliografía dedicada al desarrollo de variantes cuánticas de protocolos criptográficos, como la Prueba de Conocimiento Nulo Cuántica (QZKP), es reducida. Dado que la QZKP funciona al margen de presunciones matemáticas de alta exigencia computacional, inaugura una ruta fascinante para la exploración futura en esta área.

Este manuscrito plantea una inédita Prueba de Conocimiento Nulo Cuántica de Verificador Designado (DV-QZKP) para la validación de usuarios. El modelo entrelaza las propiedades físicas elementales de los cúbits con una versatilidad y resistencia estructural comparables a las fibras de rayón, aplicando principios equivalentes a los presentes en determinados protocolos QKD [6], [7].

Por una parte, el protocolo diseñado exige, como premisa, que tanto el verificador como el probador posean una clave secreta en común. Es exactamente este secreto compartido el que determina la condición de verificador designado del sistema. Por otra parte, la protección del protocolo recae esencialmente sobre la aleatoriedad intrínseca del resultado cuando un determinado estado cuántico experimenta una proyección sobre una base seleccionada de forma errónea [8]. Dicha aleatoriedad garantiza que cualquier alteración respecto a la base de proyección estipulada vulnere la aptitud para obtener datos inteligibles, preservando en consecuencia la inalterabilidad del protocolo.

II. Procotolo DV-QZKP Propuesto

El protocolo propuesto es una Prueba de Conocimiento Nulo Cuántica de Verificador Designado en la cual tanto el probador (Alice) como el verificador designado (Bob) comparten un secreto. Este protocolo permite a Bob autenticar a Alice a través de un canal cuántico, permitiéndole a ella demostrar el conocimiento del secreto compartido sin revelar ninguna información a posibles espías.

II-1. Fase de compromiso:

1. Alice y Bob precomparten una secuencia binaria aleatoria secreta $a \in \{0,1\}^n$. Dentro del algoritmo, cuando se utiliza una secuencia binaria para la codificación o medición de cúbits, cada 0 corresponde al uso de la base computacional, y cada 1 corresponde al uso de la base de Hadamard.

II-2. Fase en línea:

Entrada: Secuencia binaria aleatoria secreta compartida $a \in \{0,1\}^n$.

Salida: Aceptación o rechazo.

1. **Compromiso de Alice**: Alice genera una secuencia binaria aleatoria secreta $b \in \{0,1\}^n$ y envía $a \oplus b$ a Bob como un compromiso. A partir de esto, Bob puede derivar b usando a, gracias a que $a \oplus b \oplus a = b$.
2. **Desafío de Bob**: Bob prepara el *estado cuántico secreto* $|\psi\rangle$ codificando la secuencia binaria secreta compartida a utilizando las bases cuánticas especificadas por la

secuencia binaria secreta compartida b, de modo que $|\psi\rangle = |a\rangle\, H_{\wr b}$, es decir:

- si $b_i = 0$, usa la base computacional,
- si $b_i = 1$, usa la base de Hadamard.

Luego, Bob genera una secuencia binaria aleatoria secreta $c \in \{0,1\}^n$ y define $|\psi'\rangle = |\psi\rangle\, X_{\wr c \cap \wr \bar{b}} Z_{\wr c \cap \wr b}$, es decir, aplica puertas cuánticas sobre $|\psi\rangle$ de la siguiente manera:

- puerta X donde $c_i = 1 \ \wedge \ b_i = 0$.
- puerta Z donde $c_i = 1 \ \wedge \ b_i = 1$.

Bob envía este *estado de desafío* $|\psi'\rangle$ a Alice.

3. **Respuesta de Alice**:
   Alice prepara el *estado de prueba* $|\psi''\rangle = |\psi'\rangle\, Z_{\wr b} H_{\wr(a\oplus b)} Z_{\wr a}$, es decir, aplica puertas cuánticas sobre $|\psi'\rangle$ de la siguiente manera:
   Alice envía este *estado de respuesta* $|\psi''\rangle$ a Bob.
4. **Verificación de Bob**: Bob mide el *estado de prueba* recibido usando las bases definidas por a, obteniendo un resultado de $b \oplus c$, por lo que el efecto de b puede eliminarse mediante $b \oplus c \oplus b = c$. Por lo tanto, si Alice no está siendo honesta, Bob recuperará una secuencia $c' \neq c$. Por consiguiente, Bob compara c' con su original c calculando el número de diferencias entre ellos:

$$W(c \oplus c'),$$

donde W se refiere al peso de Hamming y $\|\cdot\|_1$ es la norma 1 del vector.

- Si $W(c \oplus c') = 0$, Alice es aceptada por Bob como honesta.
- De lo contrario, si las diferencias exceden un umbral, Alice es rechazada.

Las secuencias b y c deben ser aleatorias para cada ejecución del protocolo. Un diagrama de flujo en la Fig. 1 ilustra esta Prueba de Conocimiento Nulo (DV-QZKP).

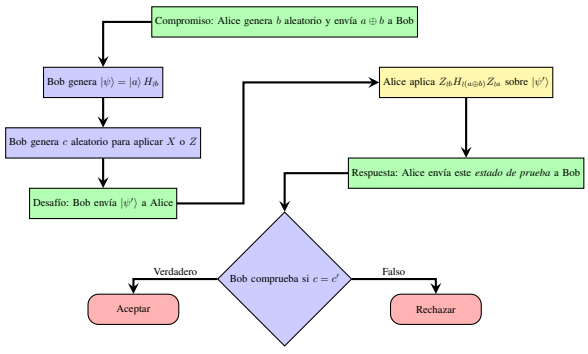


Figura 1: Diagrama de flujo del protocolo.

III. Seguridad del Protocolo

Un probador honesto tiene una probabilidad de 1 de ser aceptado bajo condiciones ideales. Sin embargo, en condiciones realistas, ocurren errores cuánticos y típicamente se caracterizan por el QBER. Por lo tanto, debido a los errores inherentes presentes en el canal cuántico, el verificador no debería rechazar cada prueba en la que c y c' no coincidan al $100\,\%$. En su lugar, deberían definir un umbral $k\,\%$ para el porcentaje mínimo de coincidencia entre c y c'.

III-A. Solidez

Siendo $X_{d\,\%}$ la variable de la tasa de coincidencia de un probador deshonesto, la probabilidad de que un probador deshonesto tenga k o más coincidencias, a saber, el porcentaje de coincidencias de $\frac{k}{n} \cdot 100$, se expresa mediante la siguiente ecuación:

$$P(X_d \geq k) = \sum_{i=k}^{n} \binom{n}{i} \frac{1}{2^n}. \tag{1}$$

III-B. Completitud

Siendo $X_{h\,\%}$ la variable de la tasa de coincidencia de un probador honesto, la probabilidad de que un probador honesto como mucho k coincidencias, a saber, un porcentaje de coincidencias de $\frac{k}{n} \cdot 100$, se expresa mediante la siguiente ecuación:

$$P(X_h \leq k) = \sum_{i=0}^{k} \binom{n}{i} (1 - QBER)^i \cdot (QBER)^{n-i}. \tag{2}$$

IV. Conocimiento Nulo

Si Eva tiene acceso a la secuencia $a \oplus b$, puede convencer a Bob de que tiene conocimiento de a y b con la probabilidad asociada al número de coincidencias entre c y su c' dada, la cual sigue una distribución binomial de la forma:

$$X_a \sim B\left(n, \frac{5 - 2QBER}{8}\right), \tag{3}$$

donde X_a es la tasa de la variable de éxito para un atacante que tiene acceso a $a \oplus b$.

V. Simulaciones del Protocolo

Se han llevado a cabo varias implementaciones utilizando el simulador cuántico Qiskit desarrollado por IBM [9]. Se realizaron dos simulaciones distintas para modelar dos tipos diferentes de errores. El código de implementación completo está disponible en el repositorio abierto [10].

V-A. Error de Volteo de Flip

Los errores de volteo de fase y amplitud se modelan utilizando operadores de Kraus definidos como:

$$\begin{aligned} E_0 &= \sqrt{1-p-q}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \\ E_1 = \sqrt{p}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, &\qquad E_2 = \sqrt{q}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned} \tag{4}$$

donde p y q representan las probabilidades de volteo de amplitud y volteo de fase, respectivamente.

V-B. Error de Amortiguamiento de Fase y Amplitud

El error de amortiguamiento de fase describe la pérdida de coherencia en un cúbit sin ninguna pérdida de energía. Se modela utilizando los siguientes operadores de Kraus:

$$E_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\lambda} \end{bmatrix}, \quad E_1 = \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{\lambda} \end{bmatrix} \tag{5}$$

donde λ puede interpretarse como la probabilidad de decoherencia de fase sin pérdida de energía.

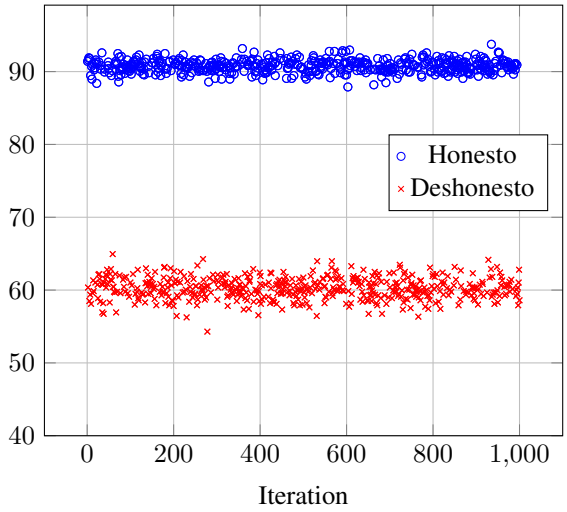


Figura 2: Error de volteo de flip donde el atacante tiene acceso a $a \oplus b$.

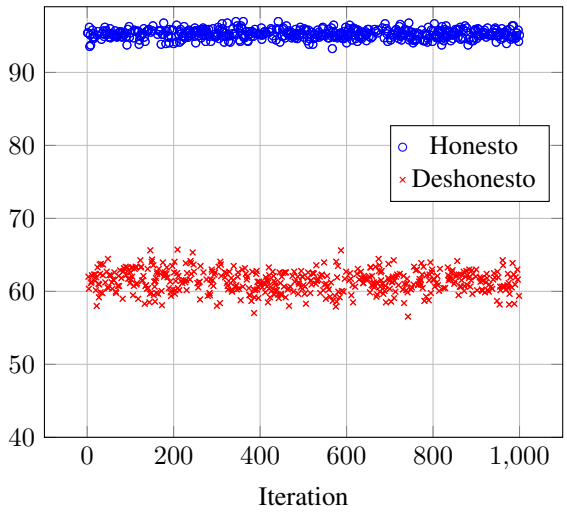


Figura 3: Error de amortiguamiento de fase y amplitud donde el atacante tiene acceso a $a \oplus b$.

El error de amortiguamiento de amplitud modela la interacción de un cúbit con un entorno térmico a temperatura cero, donde la energía del cúbit se disipa en el entorno [11]. Está representado por los siguientes operadores de Kraus:

$$\hat{E}_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{bmatrix}, \quad \hat{E}_1 = \begin{bmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{bmatrix} \tag{6}$$

donde γ representa la probabilidad de transición del estado $|1\rangle$ al estado $|0\rangle$.

VI. Conclusiones

En este trabajo se ha propuesto una nueva Prueba Cuántica de Conocimiento Nulo basada en codificación conjugada para la autenticación de usuarios sobre canales cuánticos. A diferencia de los esquemas clásicos de pruebas de conocimiento nulo, que dependen de problemas matemáticos difíciles de resolver, este enfoque se basa en principios físicos fundamentales de la mecánica cuántica, garantizando la seguridad del protocolo sin asumir la dificultad computacional de ciertos problemas.

Mediante simulaciones en Qiskit, se ha validado la viabilidad del protocolo. Los resultados muestran que el protocolo es claramente viable, manteniendo un buen rendimiento en la autenticación. Además, la flexibilidad del esquema permite ajustarlo a distintos niveles de recursos cuánticos disponibles, lo que facilita su implementación en hardware cuántico limitado.

Esta prueba de conocimiento nulo es una alternativa para la seguridad en redes cuánticas. Su solidez frente a ataques permite su integración con protocolos de Distribución Cuántica de Claves, mejorando la protección en entornos críticos como gobiernos, banca y defensa.

Como trabajo futuro, se propone su implementación en hardware cuántico real y el análisis de posibles ataques cuánticos basados en entrelazamiento contra su implementación, como por ejemplo la clonación imperfecta y el posible uso de six-state para este protocolo.

Agradecimientos

Este trabajo ha sido posible gracias a la Cátedra de Ciberseguridad C065/23 de la Universidad de La Laguna, financiada por INCIBE, con el Plan de Recuperación, Transformación y Resiliencia (Next Generation) financiado por la Unión Europea. También forma parte de los proyectos de investigación PID2022-138933OB-I00 y 2023DIG28 IACTA financiados por MCIN/AEI/10.13039/501100011033/FEDER UE, y la Fundación CajaCanarias la Caixa.

Referencias

[1] O. Goldreich, "Foundations of cryptography: volume 2, basic applications," 2001.

[2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, 1996, pp. 212–219.

[3] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134.

[4] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, p. 120–126, 1978.

[5] A. Broadbent, Z. Ji, F. Song, and J. Watrous, "Zero-knowledge proof systems for qma," in *2016 IEEE 57th Annual Symposium on Foundations of Computer Science*, 2016, pp. 31–40.

[6] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical Computer Science*, vol. 560, pp. 7–11, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304397514004241

[7] C. H. Bennett, "Quantum cryptography using any two nonorthogonal states," *Physical Review Letters*, vol. 68, pp. 3121–3124, May 1992. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.68.3121

[8] S. Wiesner, "Conjugate coding," *ACM Sigact News*, vol. 15, no. 1, pp. 78–88, 1983.

[9] A. Javadi-Abhari, M. Treinish, K. Krsulich, C. J. Wood, J. Lishman, J. Gacon, S. Martiel, P. D. Nation, L. S. Bishop, A. W. Cross, B. R. Johnson, and J. M. Gambetta, "Quantum computing with Qiskit," 2024.

[10] J. Garcia-Diaz, "GitHub - jorgegardiaz/conjugate_coding_QZKP: Scripts with authentication protocol simulations using a QZKP based on conjugate coding," https://github.com/jorgegardiaz/conjugate-coding-QZKP, 2025.

[11] V. Giovannetti and R. Fazio, "Information-capacity description of spin-chain correlations," *Physical Review A*, vol. 71, no. 3, Mar. 2005. [Online]. Available: http://dx.doi.org/10.1103/PhysRevA.71.032314


CÓDIGO
DE BARRAS
RECSI
2026
Financiado por
la Unión Europea
NextGenerationEU
GOBIERNO
DE ESPAÑA
MINISTERIO
PARA LA TRANSFORMACIÓN DIGITAL
Y DE LA FUNCIÓN PÚBLICA
SECRETARÍA DE ESTADO
DE TELECOMUNICACIONES
E INFRAESTRUCTURAS DIGITALES
Plan de
Recuperación,
Transformación
y Resiliencia
incibe_
INSTITUTO NACIONAL DE CIBERSEGURIDAD