CONFERENCE PROCEEDINGS

CRYPTOLOGY 2026

PROCEEDINGS OF THE 10TH INTERNATIONAL CRYPTOLOGY AND INFORMATION SECURITY CONFERENCE 2026

1-4 SEPTEMBER, 2026
Kota Kinabalu, Sabah, Malaysia

EDITORS

Azni Haslizan Ab Halim
Farida Hazwani Mohd Ridzuan
Rozaimi Zakaria
Che Haziqah Che Hussin

MSCR
Malaysian Society for Cryptology Research

UMS
UNIVERSITI MALAYSIA SABAH

RESEARCH. INNOVATION. SECURITY.

Proceedings of the 10th International Cryptology and Information Security Conference 2026

1st – 4th September 2026
Kota Kinabalu, Sabah,
Malaysia

Published by
Institute for Mathematical Research (INSPEM)
Universiti Putra Malaysia
43400 UPM Serdang
Selangor Darul Ehsan

©Institute for Mathematical Research (INSPEM), 2026

All rights reserved. No part of this book may be reproduced in any form without permission in writing from the publisher, except by a reviewer who wishes to quote brief passages in a review written for inclusion in a magazine or newspaper.

First Print 2026

ISSN 2716-6783

FOREWORD

Greetings,

On behalf of the Malaysian Society for Cryptology Research (MSCR), it is our great pleasure to welcome you to CRYPTOLOGY 202,6 held in Kota Kinabalu, Sabah, from 1–4 September 2026. The conference series traces its roots to the 1st and 2nd National Cryptology Conferences (NCC04 and NCC06), held in 2004 and 2006 and spearheaded by the Institute for Mathematical Research (INSPEM), Universiti Putra Malaysia (UPM). Their success led to the establishment of the Malaysian Society for Cryptology Research (MSCR) in 2007, followed by the inaugural CRYPTOLOGY conference in Kuala Lumpur in 2008. As a biennial conference, CRYPTOLOGY provides a platform for researchers, academics, practitioners, students, and industry professionals to share research, exchange ideas, and foster collaboration in cryptography and information security. Over the years, MSCR has collaborated with numerous universities and organisations to advance cryptography research and strengthen connections within Malaysia and the international community. CRYPTOLOGY 2026 marks the 10th edition of the conference, 18 years after its inaugural edition. To encourage broader participation, this edition introduces a poster session, providing researchers and students with an opportunity to share early-stage and ongoing work. The programme reflects several important developments in contemporary cryptography. Artificial Intelligence (AI) and cryptography are explored through a keynote and a workshop on deep learning in side-channel analysis, both delievered by Dr. Stjepan Picek, Professor at University of Zagreb, Croatia, and Associate Professor at Radboud University, Netherlands. The conference also examines post-quantum cryptography (PQC) migration through a keynote by Michael Osborne, CTO of IBM Quantum Safe, IBM Research Europe, addressing the practical challenges of preparing for the transition to quantum-resistant cryptography. A further keynote by Datuk Prof. Dr. Muhammad Rezal Kamel Ariffin, Director of the Malaysian Cryptology Technology and Management Centre (PTPKM), explores crypto-agility in the era of PQC and AI, including Malaysia's efforts towards strategic migration, research, capability development, and crypto-agile governance. Together, these themes: AI, side-channel analysis, PQC, cryptographic im-

plementation, and crypto-agility, highlight the breadth and dynamism of cryptography today and reinforce the importance of collaboration in addressing emerging challenges. The technical program also explores a broad range of areas including network security, cryptographic primitives, applied cryptography, privacy and cryptanalysis. We gratefully acknowledge the support of our Gold Sponsor, Jabatan Teknologi Digital Sabah (JTDI); Bronze Sponsors, Zchwantech and MSC Trustgate; strategic partners, PTPKM, Sabah Convention Bureau, and MaGraph; and supporting organisation, CyberSecurity Malaysia. Our sincere appreciation also goes to the local organising committee led by Ts. Dr. Arif Mandangan, volunteers, speakers, reviewers, authors, and all our partners and supporters whose contributions have made CRYPTOLOGY 2026 possible. We hope CRYPTOLOGY 2026 provides a stimulating environment for knowledge sharing, meaningful discussions, new collaborations, and reconnecting with colleagues and friends. We also invite you to experience the warmth and hospitality of Kota Kinabalu and the beautiful state of Sabah.

Thank you.

DR. MUHAMMAD REZA Z'ABA

Acting President,

Malaysian Society for Cryptology Research (MSCR)

WELCOMING NOTES

Assalamualaikum Warahmatullahi Wabarakatuh, Salam Sejahtera, Salam Malaysia Madani, and Selamat Datang ke Sabah. It is my great honour and pleasure to welcome all participants to CRYPTOLOGY2026, the 10th International Cryptology and Information Security Conference, and to have the privilege of bringing this prestigious conference to Sabah, with Universiti Malaysia Sabah (UMS) as the co-host. On behalf of the Organising Committee, I would like to express my sincere gratitude to the Malaysian Society for Cryptology Research (MSCR) for placing its trust in us and giving us the opportunity to host CRYPTOLOGY2026 in Sabah. I am equally grateful to UMS for its full support in providing the human resources, facilities and institutional commitment necessary to make this conference possible. Our sincere appreciation also goes to our Sponsors: Jabatan Teknologi Digital dan Inovasi Negeri Sabah (JTDI), Zchwantech and Trustgate, our Strategic Partners: Pusat Teknologi dan Pengurusan Kriptologi Malaysia, Persatuan Matematik dan Grafik Berkomputer Negeri Sabah (MAGRAPH) and Sabah Convention Bureau, and our Supporter, CyberSecurity Malaysia. I also extend my deepest appreciation to all highly committed Chairs, including our General Chair, Dr Muhammad Reza Z'aba, for their amazing dedication, commitment and teamwork. As a proud Sabahan, I am especially delighted to see the cryptology community gathered here in Sabah. I hope CRYPTOLOGY2026 will contribute towards disseminating knowledge and awareness of Post-Quantum Cryptography (PQC), encouraging greater awareness of the importance of PQC migration in Sabah, and stimulating interest among our young Sabahan talents to explore cryptology, cybersecurity and emerging technologies. I hope all participants will benefit from the academic exchange, establish meaningful collaborations, and enjoy the majestic landscapes and warm hospitality of Sabah. Finally, we sincerely apologise for any unintended shortcomings throughout the conference. Your feedback, comments and suggestions will be highly appreciated and will help us improve future editions of CRYPTOLOGY. Welcome to Sabah, and enjoy CRYPTOLOGY2026. Terima kasih.

Thank you.

TS DR. ARIF MANDANGAN
Program Chair of CRYPTOLOGY2026

EDITORIAL PREFACE

Cryptology has always evolved alongside the way society communicates, stores, and protects information. From the earliest substitution techniques associated with classical civilizations, through mechanical cipher systems, to the development of modern mathematical cryptography, the fundamental objective has remained unchanged: to protect information and establish trust between communicating parties.

The significance of cryptology became particularly evident during the twentieth century, when advances in cryptography and cryptanalysis played an important role in global communications and, subsequently, in the emergence of modern computing. The development of electronic computers transformed how information was created, processed, transmitted, and stored. As computing systems became increasingly interconnected, cryptography evolved beyond the protection of confidentiality to provide essential security properties including integrity, authentication, secure key establishment, and non-repudiation.

Today, this evolution continues at an unprecedented pace.

Digital transformation has created an environment in which information moves continuously across cloud platforms, mobile devices, Internet of Things systems, distributed infrastructures, and increasingly intelligent computing environments. Organizations are no longer protecting information within clearly defined network boundaries. Instead, security must be maintained across interconnected systems involving multiple organizations, service providers, devices, applications, and geographical locations.

At the same time, advances in artificial intelligence, large-scale data processing, quantum computing, and increasingly sophisticated cyber threats are reshaping both the opportunities and challenges facing the cryptology and information security community. Questions concerning post-quantum security, lightweight cryptography, privacy-preserving computation, secure authentication, digital identity, blockchain and distributed systems, cloud security, network security, cryptanalysis, and trustworthy artificial intelligence are becoming increasingly important.

These developments remind us that cryptology cannot be viewed as an isolated mathematical discipline. Its effectiveness ultimately depends on how cryptographic principles are designed, implemented, deployed, managed, and integrated within real-world information systems. Likewise, information security cannot be effectively achieved without a sound understanding of the cryptographic mechanisms upon which digital trust is built. The interaction between theoretical cryptology, practical cybersecurity, engineering, computing, and emerging technologies therefore remains essential.

Against this rapidly changing technological landscape, the 10th International Cryptology

and Information Security Conference 2026 (CRYPTOLOGY2026) continues its role as a platform for researchers, academics, practitioners, students, and industry professionals to exchange ideas, present research findings, and explore emerging directions in cryptology and information security.

Reaching the 10th edition of CRYPTOLOGY also represents an important milestone. Over the years, the conference has provided a dedicated forum for the discussion of theoretical developments, practical security solutions, new cryptographic constructions, cryptanalysis, security applications, and interdisciplinary research. CRYPTOLOGY2026 builds upon this foundation while encouraging the research community to address the security challenges arising from increasingly interconnected and intelligent digital environments.

The papers presented in these proceedings reflect the continuing diversity and development of the field. They demonstrate how cryptographic research continues to evolve from its mathematical foundations toward increasingly sophisticated applications in modern computing environments. Collectively, these contributions represent ongoing efforts to strengthen the confidentiality, integrity, authenticity, privacy, resilience, and trustworthiness of digital systems.

It is our hope that CRYPTOLOGY2026, held in Kota Kinabalu, Sabah, Malaysia, from 1–4 September 2026, will provide more than an opportunity to present research results. We hope that it will stimulate meaningful discussions, encourage new collaborations, inspire young researchers, and create opportunities for ideas from different areas of cryptology and information security to intersect.

As technology continues to evolve, new security challenges will inevitably emerge. At the same time, the creativity and ingenuity of the research community will continue to provide new approaches for addressing them. Through conferences such as CRYPTOLOGY2026, we strengthen the connection between fundamental research and practical security and contribute collectively towards a safer and more trustworthy digital future.

We warmly welcome all participants to CRYPTOLOGY2026 and hope that the conference will be intellectually rewarding, professionally beneficial, and personally memorable.

Thank you.

Editorial Board,

10th International Cryptology and Information Security

CRYPTOLOGY2026

Table of Contents

Foreword iii

Welcoming Notes v

Editorial Preface vii

Table of Contents ix

Committee xii

On Transaction-Structure Efficiency in Account-Based Privacy-Preserving Payments 1

Development of Identity-based Proxy Re-encryption Library 17

Evolutionary Optimisation of an AES S-Box 38

Privacy Protection Technology for Principal Component Analysis of Multi-Party Data Based on Multi-Cloud Architecture 51

Light Image Encryption via Wavelet Transform 76

A Perfect Code Based (t+1, 2t+1)-Threshold Non-Perfect Secret Sharing Scheme 89

Guess-and-Determine Cryptanalysis of the 3-Round L-Feistel 101

Low-Trace Profiling Side-Channel Attack on Kyber basemul Using Random Forest and ANOVA-Based Feature Selection 116

Certificateless Identification Schemes: A Comprehensive Survey 134

Efficient Threshold Signatures: A Runtime-Based Comparative Study of FROST Variants 149

Measuring Neural Attackability of Lightweight Authenticated Encryption: Towards Robust Cipher Design 166

VORTEX-256: A High-Diffusion ARX-Based Stream Cipher with Fast Avalanche Convergence and Enhanced Security Margins **188**

Security Analysis of a Dual-Problem Blind Signature Scheme **210**

Revisiting the LUCELG and Cramer–Shoup Analogous Cryptosystem: A Formal Security Analysis **220**

Hardware Implementation of CiliPadi Family on FPGA **232**

Security Proof for a Blind Signature Scheme under the Strong RSA Assumption **245**

Provable Security Analysis of a Dual-Hardness Digital Signature Scheme **253**

State-Dependent Vulnerability Detection via Enriched Semantic Graph Representations **265**

Performance and Efficiency Analysis for Hybrid PKI Trust Model in Multi-Cloud Environments **273**

A Scoring System for Transport Layer Security **285**

Loss Functions in GAN-Based Audio Steganography: A Systematic Review of Limitations and Performance Trade-Offs **306**

A New NTRU Variant with Composite Private Structure **320**

Mapping the Cryptographic Landscape: A Layered Abstraction Framework Illustrated via Identification Schemes without Certificates **333**

Chaos-based Double Image Encryption using 4D Henon-type Hyperchaotic Map for Colour Images **348**

An Inventory-Driven Risk Taxonomy for Post-Quantum Cryptography Migration **367**

Governance-Driven PQC Migration for National Critical Information Infrastructure: Operationalisation Challenges and Early Readiness Insights from Malaysia **380**

QRAF: A Quantum Risk Assessment Framework for Post-Quantum Cryptographic Readiness, Migration Prioritization and National Cyber Resilience **395**

A Comparative Evaluation of Wiener's Continued Fraction and Boneh–Durfee Lattice-Based Attacks on RSA under Private Exponent MSB Leakage **414**

Security Analysis of a DNA-Inspired Symmetric Lightweight Block Cipher **428**

Post-Quantum Cryptography Audit and Compliance Frameworks: A Systematic Literature Review and Implications for Malaysia's National QC Migration Governance **454**

Implementation of Dynamic Double-Lock Bézier-Menezes-Vanstone Elliptic Curve Cryptography **481**

Torus: a novel block cipher via power mappings over pseudo-Mersenne toroidal groups **496**

COMMITTEE

Advisory Committee	Datuk Prof. Dr. Muhammad Rezal Kamel Ariffin
	Prof. Ts. Gs. Dr. Rodeano Roslee
General Chair	Dr Muhammad Reza Z'aba
Program Chair	Ts Dr Arif Mandangan
International Program Committee	Professor Emeritus Dato' Dr Mohamed Ridza Wahiddi
	Prof. Dr. Abderrahmane Nitaj
	Prof. Dr. Amr M. Youssef
	Prof. Dr. Lamberto Rondoni
	Prof. Dr. Keith Martin
	Dr. Yanbin Pan
	Prof. Dr. Mohd Salmi Md Noorani
	Ts. Dr. Solahuddin Shamsuddin
	Assoc Prof. Dr. Norziana Jamil
	Dr Chin Ji-Jian
	Dr. Hai Van Dang
	Dato' Ts. Dr. Haji Amirudin Abdul Wahab
Secretary I and II	Dr Amir Hamzah Abd Ghafar
	Dr Nurul Nur Hanisah Adenan
Treasury I and II	Dr Wan Nur Aqlili Wan Mohd Ruzai
	Dr Muhammad Asyraf Asbullah
Sponsorship Chair	Prof Dr Abdullah Bade
	Dr Muhalim Mohamed Amin
	Pn Kamilah Abdullah

Logistic and Technical Chair	Mohamed Azlan Daud
	Dr Tea Boon Chian
Scientific and Publication Chair	Assoc. Prof. Dr Azni Haslizan Ab Halim
	Assoc. Prof. Dr Farida Hazwani Mohd Ridzuan
	Assoc. Prof. Dr Rozaimi Zakaria
	Dr Che Haziqah Che Hussin
Publicity and Media Chair	Muhammad Hafiz Abu Hassan
	Irfan Bin Khaidhir
	Faridatul Akhma Ishak
Workshop and Exhibition Chair	Ts. Rechard Lee
	Nor Siti Khadijah Arunah
Secretariat Chair	Norhafiza Hamzah
	Asdalifah Talibe
	Priscilla Kyle Payne
Protocol and Welfare Chair	Dr Suzelawati Zenian
	Zaturrawiah Ali Omar
Invitation and International Delegates Chair	Dr Abdullah Al-Yaari
	Nurul Amiera Sakinah Abdul Jamal
	Vaishnavi A/P Nagaraja

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 1–16.

On Transaction-Structure Efficiency in Account-Based Privacy-Preserving Payments

Haruto Yamamoto, Hao-Yi Hsu, Raylin Tso, Kai-En Cheng, Shu-Chen Yang and Tao-Hsiang Chang

National Chengchi University, Taiwan
haruto.y0327@gmail.com, 112753501@g.nccu.edu.tw,
tsoraylin@gmail.com,114753501@nccu.edu.tw113753503@nccu.edu.tw,
tchang@cs.nccu.edu.tw

Abstract. This paper revisits account-based privacy-preserving payments from a transaction-structure perspective. We observe that, in designs such as BlockMaze, direct public/private balance conversion creates an explicit structural linkage between the two balance domains. To address this issue, we propose Updated BlockMaze (UBM), which refactors public-to-private and private-to-public transfers into a unified sender-side transfer and receiver-side claim paradigm. UBM extends the original BlockMaze transaction set without introducing new cryptographic primitives or hardness assumptions. We provide a security analysis showing that UBM preserves ledger indistinguishability, transaction unlinkability, transaction non-malleability, and balance correctness. We evaluate two representative cross-boundary payment workflows using the Groth16 proof system over `ALT_BN128` on a laptop equipped with a 13th-generation Intel Core i5-13500H processor and 16 GB of RAM, running 64-bit Ubuntu 18.04.1. Although transaction-level prover cost is redistributed across phases, UBM reduces total proof generation time and cumulative serialized transaction size at the workflow level. The evaluation is limited to these two BlockMaze workflows; deployment-level evaluation and comparisons under a common implementation setting remain future work.

Keywords: Account-based Blockchain Privacy · Transaction Structure · Cross-Boundary Payments

1 Introduction

Blockchain systems such as Bitcoin and Ethereum provide decentralized and publicly verifiable ledgers, but the same transparency also exposes transaction relations, balance flows, and user interaction patterns to all participants [15, 7]. Privacy-preserving blockchain payments have therefore been studied extensively. At a high level, existing designs follow two dominant paradigms, namely the unspent transaction output (UTXO) model and the account-based model, and these paradigms differ fundamentally because they represent ownership, state, and state updates in different ways [4, 1].

The UTXO line has produced several influential privacy-preserving systems. Zerocoin and Zerocash use zero-knowledge techniques to decouple spendability from publicly visible ownership [14, 19]. Monero further develops this direction through ring signatures and confidential transactions [16, 17], while proof systems such as Bulletproofs improve proof size for confidential transaction components [6]. These designs provide strong privacy, but they rely on the expendable-note abstraction of the UTXO model and do not directly resolve privacy under persistent account-state updates [1].

Privacy on programmable and account-oriented platforms has been explored from several directions. Hawk and Zexe show that privacy-preserving computation can be embedded into smart-contract settings [13, 3]. Tornado Cash demonstrates application-layer privacy on Ethereum-style infrastructure, although it follows a note-based workflow rather than native account balances [18]. Closer to account-style balances, Zether and QuisQuis investigate confidential or anonymous payments under account-based settings [5, 9]. More recently, LedgerMaze and BlockMaze show that strong privacy can be integrated into account-model blockchains through non-interactive zero-knowledge techniques while preserving account-style state transitions [2, 12]. These works establish the feasibility of account-based privacy, but they also reveal a central challenge: privacy-preserving state transitions in the account model incur substantial cryptographic overhead.

Part of this overhead is explained by the proof system itself. Prover-side cost grows with circuit complexity even when verification is succinct, and a substantial line of work therefore focuses on proof systems and proof-related components [11, 8, 10]. However, proof-system efficiency is not the only relevant dimension. In account-based privacy-preserving payments, the way a payment is decomposed into ledger-level state transitions also determines how much cryptographic work is required. In particular, proof-system choice alone does not determine whether a logical payment is realized by one transition, two transitions, or a longer sender-side and receiver-side workflow.

This observation is especially relevant for BlockMaze [12]. In BlockMaze, private-to-private transfer already follows a sender-side transfer followed by a receiver-side claim, but public/private balance conversion remains tied to direct execution. As a result, cross-boundary payments are not handled by the same transfer-and-claim structure already used for private-to-private transfer. This leads to two related limitations. First, representative cross-boundary payment workflows may require additional ledger actions and repeated cryptographic work. Second, direct public/private conversion creates an explicit structural linkage between the public and private balance domains that is absent from the split transfer-and-claim structure used elsewhere in the system. The limitation is therefore not only a proof-system issue; it is also a transaction-structure issue.

This paper revisits BlockMaze from precisely this perspective. We propose UBM, an extension of BlockMaze that expresses public-to-private and private-to-public transfers through sender-side transfer and receiver-side claim phases. UBM does not introduce a new proof system, a new encryption primitive, or a new hardness assumption. For example, consider a public-to-private payment from a sender to a receiver. BlockMaze realizes this payment as $\mathsf{Mint} \rightarrow \mathsf{Send} \rightarrow \mathsf{Deposit}$: the sender first converts the payment amount from its public balance to its private balance, then creates a private transfer, and the receiver claims the transfer. UBM realizes the same logical payment as $\mathsf{Mint}^\star \rightarrow \mathsf{Dep\_Mi}^\star$, eliminating the intermediate update to the sender's private state. Our goal is not to claim a universal efficiency improvement for all account-based privacy systems, but to show that, within the BlockMaze design space, transaction restructuring can remove the direct structural linkage between the public and private balance domains and reduce redundancy in representative cross-boundary payment workflows.

Our contributions.

- We identify a structural limitation in the BlockMaze transaction structure: direct balance-conversion transactions create an explicit structural linkage between the public and private balance domains.
- We propose UBM, which extends BlockMaze with four additional transaction types, namely $\mathsf{Mint}^\star$, $\mathsf{Dep\_Mi}^\star$, $\mathsf{Redeem}^\star$, and $\mathsf{Dep\_Re}^\star$. These transactions refactor cross-boundary balance conversion into sender-side transfer and receiver-side claim phases while remaining compatible with the original BlockMaze transaction set.

- We provide a preservation-oriented security analysis of UBM and evaluate it on representative cross-boundary payment workflows. The results show that, in representative cross-boundary payment workflows, the transaction restructuring can reduce workflow-level proof generation time and cumulative serialized transaction size relative to the original BlockMaze workflow, without modifying the underlying proof system or introducing new cryptographic assumptions.

2 Preliminaries

Notation. Let λ be the security parameter. All algorithms in this paper are probabilistic polynomial-time unless stated otherwise. For a randomized algorithm $\mathcal{A}$, we write $y \leftarrow \mathcal{A}(x)$ for running $\mathcal{A}$ on input x and assigning its output to y. For a finite set $\mathcal{X}$, we write $x \xleftarrow{\$} \mathcal{X}$ for uniform sampling from $\mathcal{X}$. We write $[n] = \{1, \ldots, n\}$. Let $H : \{0,1\}^* \to \{0,1\}^{\ell_H}$ be a hash function. For an ordered tuple $(x_1, \ldots, x_n)$, we write $H(x_1, \ldots, x_n)$ for the hash of its canonical encoding. For a finite multiset $\mathcal{S}$ of digests, $\mathsf{Root}(\mathcal{S})$ denotes the Merkle root of the tree over $\mathcal{S}$, and $\mathsf{Path}(u, \mathcal{S})$ denotes the authentication path for leaf u.

Public-key encryption. A public-key encryption scheme Π_{enc} consists of algorithms

$$\Pi_{\mathsf{enc}} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}).$$

On input 1^λ, KeyGen outputs a key pair (pk, sk). On input pk and a message m, Enc outputs a ciphertext c. On input sk and c, Dec outputs a message or $\perp$. Correctness requires that, for every honestly generated (pk, sk) and every message m,

$$\mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) = m.$$

We assume that Π_{enc} satisfies indistinguishability under adaptive chosen-ciphertext attacks (IND-CCA security) and key indistinguishability under adaptive chosen-ciphertext attacks (IK-CCA security). IND-CCA security hides the encrypted transfer data, whereas IK-CCA security provides ciphertext key privacy under adaptive chosen-ciphertext attacks.

zk-SNARK. Let $\mathcal{R}$ be an NP relation. A zk-SNARK for $\mathcal{R}$ consists of algorithms

$$\Pi_{\mathsf{zk}} = (\mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify}).$$

Given a circuit C representing $\mathcal{R}$, $\mathsf{Setup}(1^\lambda, C)$ outputs a proving key pk_C and a verification key vk_C. For a public statement x and a witness w such that $(x, w) \in \mathcal{R}$, $\mathsf{Prove}(\mathsf{pk}_C, x, w)$ outputs a proof π. Given vk_C, x, and π, $\mathsf{Verify}(\mathsf{vk}_C, x, \pi) \in \{0,1\}$. Completeness requires that, for every $(x, w) \in \mathcal{R}$,

$$\mathsf{Verify}(\mathsf{vk}_C, x, \mathsf{Prove}(\mathsf{pk}_C, x, w)) = 1.$$

We assume that Setup is honestly executed and that its trapdoor is unavailable to the adversary. We further assume that Π_{zk} satisfies zero knowledge, knowledge soundness, and simulation extractability. Zero knowledge requires that a proof reveal no information about its witness beyond the public statement. Knowledge soundness requires that an accepted proof produced by a probabilistic polynomial-time prover admit extraction of a valid witness, except with negligible probability. Simulation extractability requires that, even after observing simulated proofs, a probabilistic polynomial-time adversary cannot produce a new accepted statement-proof pair without a corresponding extractable witness, except with negligible probability.

Hash-based digests. In the construction, H is used to derive serial numbers, authentication tags, transfer digests, private-state digests, and Merkle-tree nodes. We write

$$\mathsf{Com}(\tau) := H(\tau)$$

for a tuple τ whenever its digest is used as a commitment to the canonical encoding of τ.

We assume that H is collision resistant. Under the canonical encoding, collision resistance implies computational binding of Com. When τ contains fresh uniformly sampled randomness as specified in the construction, we assume that $\mathsf{Com}(\tau)$ is computationally hiding. When H is evaluated as $H(sk, \cdot)$ for an honestly generated secret key sk, we assume that the resulting keyed function is pseudorandom. We also assume Merkle membership integrity: for any multiset $\mathcal{S}$, no probabilistic polynomial-time adversary can produce a valid authentication path under $\mathsf{Root}(\mathcal{S})$ for a digest not contained in $\mathcal{S}$, except with negligible probability.

These cryptographic assumptions are inherited from the BlockMaze security framework [12]; UBM introduces no additional cryptographic assumption.

3 System Model

Account-based ledger. Each user runs $(pk, sk) \leftarrow \mathsf{KeyGen}(1^\lambda)$ and is identified by

$$\mathsf{addr} = H(pk).$$

An account at time t consists of a public balance and a private state. We model the public state as

$$\mathsf{State}_t = (\mathsf{Ledger}_t, \mathsf{PubBal}_t, \mathsf{PriState}_t),$$

where Ledger_t is an append-only sequence of accepted transactions, $\mathsf{PubBal}_t[\mathsf{addr}]$ is the public balance of addr, and $\mathsf{PriState}_t[\mathsf{addr}]$ is the current digest representing the private balance of addr. The mappings PubBal_t and $\mathsf{PriState}_t$ are indexed by account addresses. The private balance value and its opening are known only to the account owner.

Transaction abstraction. A transaction is a tuple

$$\mathsf{tx} = (x, \pi, \mathsf{aux}),$$

where x is a public statement, $\pi \leftarrow \mathsf{Prove}(\mathsf{pk}_C, x, w)$ is a proof for a witness w, and aux contains the public objects required by the state transition, e.g., balance digests, transfer digests, serial numbers, ciphertexts, Merkle roots, and auxiliary authentication data. A sender-side transaction may create a transfer digest for a designated receiver, while a receiver-side claim transaction later claims the corresponding transfer by proving membership of the transfer digest under a Merkle root and revealing a fresh serial number.

For a ledger prefix Ledger_t, let TCSet_t be the set of transfer digests appearing in Ledger_t, and let SNSet_t be the set of serial numbers revealed in Ledger_t.

Ledger assumptions. Following the BlockMaze ledger model [12], we abstract the consensus layer as providing a single append-only and totally ordered sequence of accepted transactions. Once a transaction is appended to Ledger_t, it cannot be removed, reordered, or modified.

The ledger deterministically applies the proof-verification, serial-number freshness, private-state consistency, Merkle-membership, public-balance sufficiency, and transaction-specific state-transition rules specified in Section 4.4. It rejects any candidate transaction that fails a validation rule without modifying the public state and applies the corresponding state update to each accepted transaction.

The adversary cannot bypass the validation rules, directly modify the public state, or rewrite an accepted ledger prefix. Attacks against the consistency or finality of the consensus layer are outside the scope of this model.

Adversarial view. The adversary $\mathcal{A}$ is a probabilistic polynomial-time algorithm with full read access to the public parameters, public ledger, public balances, and public transaction fields. It may create accounts, control the secret keys and private-state openings of those accounts, choose transaction inputs adaptively, submit arbitrary candidate transactions, and observe whether each candidate transaction is accepted.

An account is honest if its secret key and private-state opening are not controlled by $\mathcal{A}$. The adversary does not obtain the secret key, proof witness, or private-state opening of an honest account unless the corresponding information is explicitly exposed by the public transcript. The model does not provide a corruption oracle for honest accounts. The adversary can modify the public state only through transactions accepted under the ledger validation rules.

Security goals. Following the BlockMaze security framework [12], we summarize four security goals under the cryptographic assumptions of Section 2 and the ledger assumptions above. *Ledger indistinguishability* requires that, given the public transcript, $\mathcal{A}$ cannot obtain hidden private-state data beyond the information explicitly exposed by public balances and public transaction fields, except with negligible probability. *Transaction unlinkability* requires that, given the ledger transcript, $\mathcal{A}$ cannot link a receiver-side claim transaction to its corresponding sender-side transaction with non-negligible advantage. *Transaction non-malleability* requires that $\mathcal{A}$ cannot produce a distinct accepted transaction that reuses one-time public data from a previously accepted transaction with non-negligible probability. *Balance correctness* requires that $\mathcal{A}$ cannot create value, consume the same private state more than once, claim the same transfer more than once, or make the ledger accept an invalid state transition with non-negligible probability.

4 Construction

4.1 Overview

UBM retains the original BlockMaze [12] transactions

$$\mathsf{Mint},\ \mathsf{Redeem},\ \mathsf{Send},\ \mathsf{Deposit}$$

and adds two transaction pairs,

$$(\mathsf{Mint}^\star, \mathsf{Dep\_Mi}^\star) \quad \text{and} \quad (\mathsf{Redeem}^\star, \mathsf{Dep\_Re}^\star),$$

to support transfers across the public/private boundary. The pair $(\mathsf{Send}, \mathsf{Deposit})$ realizes private-to-private transfer, while $(\mathsf{Mint}^\star, \mathsf{Dep\_Mi}^\star)$ and $(\mathsf{Redeem}^\star, \mathsf{Dep\_Re}^\star)$ realize public-to-private transfer and private-to-public transfer, respectively. The original Mint and Redeem remain available for direct conversion between a user's own public balance and private balance.

Figure 1 compares the state-transition paths of the original BlockMaze and UBM cross-boundary workflows. In each direction, UBM removes the intermediate private-state update and realizes the payment through a sender-side transfer followed by a receiver-side claim.

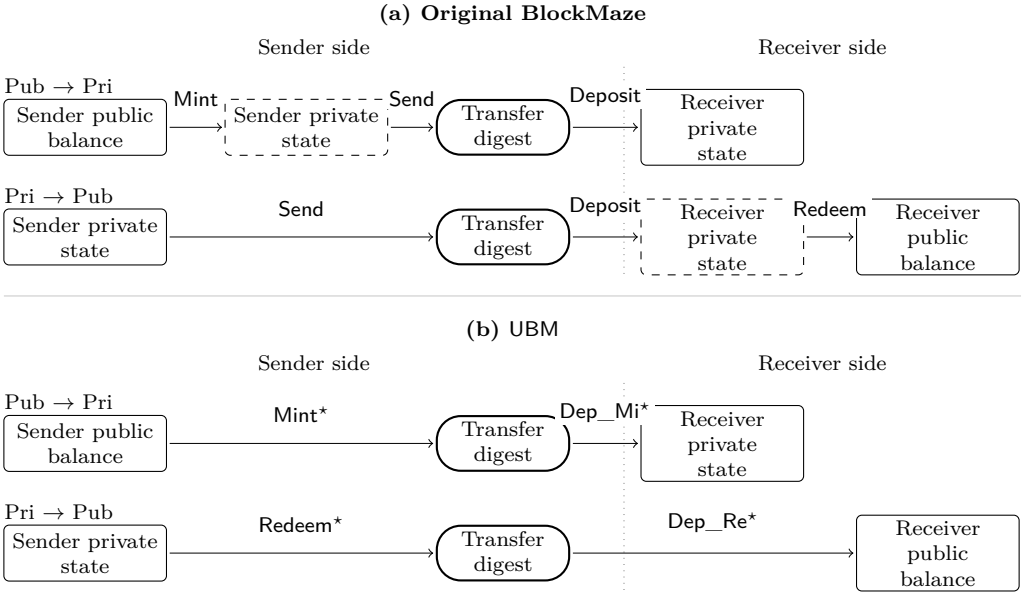


Figure 1: Comparison of cross-boundary payment workflows. Dashed nodes denote the intermediate private-state updates removed by UBM. BlockMaze composes the unchanged Send → Deposit transfer-and-claim core with a direct balance-conversion transaction, whereas UBM realizes each workflow as a sender-side transfer followed by a receiver-side claim. The private-to-private workflow remains unchanged.

The construction follows Section 3. A sender-side transaction publishes a transfer digest, a receiver ciphertext, and a proof for the corresponding state transition. A receiver-side claim transaction later proves membership of the transfer digest under a Merkle root and publishes a fresh serial number. The ledger accepts only transactions satisfying the corresponding relation and the state-transition conditions.

4.2 Setup

Let

$$\mathcal{T} = \{\mathsf{Mint}, \mathsf{Redeem}, \mathsf{Send}, \mathsf{Deposit}, \mathsf{Mint}^\star, \mathsf{Dep\_Mi}^\star, \mathsf{Redeem}^\star, \mathsf{Dep\_Re}^\star\}$$

be the set of supported transaction types. For each $T \in \mathcal{T}$, let $\mathcal{R}_T$ be the relation defining validity of T, and let C_T be a circuit for $\mathcal{R}_T$. The global setup algorithm $\mathsf{GSetup}(1^\lambda)$ invokes the zk-SNARK setup algorithm Setup for each transaction circuit. It runs

$$(\mathsf{pk}_T, \mathsf{vk}_T) \leftarrow \mathsf{Setup}(1^\lambda, C_T)$$

for every $T \in \mathcal{T}$ and publishes

$$\mathsf{pp} = \Big(H, \{(\mathsf{pk}_T, \mathsf{vk}_T)\}_{T\in\mathcal{T}}\Big).$$

Each user generates

$$(pk, sk) \leftarrow \mathsf{KeyGen}(1^\lambda), \qquad \mathsf{addr} = H(pk).$$

A private state is

$$\mathsf{st} = (\mathsf{addr}, b, \mathsf{sn}, r, \mathsf{cm}),$$

where b is the private balance, $r \xleftarrow{\$} \{0,1\}^{\ell_H}$, $\mathsf{sn} = H(sk, r)$, and

$$\mathsf{cm} = \mathsf{Com}(\mathsf{addr}, b, \mathsf{sn}, r).$$

The ledger stores only the current digest cm as $\mathsf{PriState}[\mathsf{addr}]$. For any update to balance b', the owner samples fresh r' and defines

$$\mathsf{sn}' = H(sk, r'), \qquad \mathsf{cm}' = \mathsf{Com}(\mathsf{addr}, b', \mathsf{sn}', r').$$

For a ledger prefix Ledger_t, let TCSet_t and SNSet_t be defined as in Section 3, and let

$$\mathsf{root}_t = \mathsf{Root}(\mathsf{TCSet}_t).$$

Claim transactions use root_t together with the corresponding authentication path.

4.3 Transaction Algorithms

Let

$$\mathsf{st}_S = (\mathsf{addr}_S, b_S, \mathsf{sn}_S, r_S, \mathsf{cm}_S), \qquad \mathsf{st}_R = (\mathsf{addr}_R, b_R, \mathsf{sn}_R, r_R, \mathsf{cm}_R)$$

be the sender and receiver private states, respectively, and let

$$(pk_S, sk_S), \qquad (pk_R, sk_R)$$

be the corresponding key pairs.

Direct balance conversion. For an account state $\mathsf{st} = (\mathsf{addr}, b, \mathsf{sn}, r, \mathsf{cm})$ and the corresponding secret key sk, Mint on amount $v > 0$ sets $b' = b + v$ and publishes

$$\mathsf{tx}_{\mathsf{Mint}} = (\mathsf{addr}, v, \mathsf{sn}, \mathsf{cm}', \pi_{\mathsf{mint}}),$$

where the public statement is

$$x_{\mathsf{mint}} = (\mathsf{cm}, \mathsf{addr}, \mathsf{sn}, \mathsf{cm}', v)$$

and the witness is

$$w_{\mathsf{mint}} = (b, r, sk, \mathsf{sn}', r').$$

The relation $\mathcal{R}_{\mathsf{Mint}}$ enforces validity of the current private state and correctness of the updated digest with balance $b + v$. The transaction Redeem is defined analogously with $b' = b - v$ and publishes

$$\mathsf{tx}_{\mathsf{Redeem}} = (\mathsf{addr}, v, \mathsf{sn}, \mathsf{cm}', \pi_{\mathsf{redeem}}).$$

Private-to-private transfer. In Send, the sender samples $r'_S, \rho \xleftarrow{\$} \{0,1\}^{\ell_H}$, defines

$$\tau_{\mathsf{pp}} = (\mathsf{addr}_S, v, pk_R, \rho, \mathsf{sn}_S), \qquad \mathsf{td}_{\mathsf{pp}} = \mathsf{Com}(\tau_{\mathsf{pp}}),$$

$$\mathsf{aux}_{\mathsf{pp}} = \mathsf{Enc}(pk_R, \tau_{\mathsf{pp}}), \qquad \alpha_{\mathsf{pp}} = H(sk_S, H(\mathsf{aux}_{\mathsf{pp}})),$$

sets $b'_S = b_S - v$, derives $(\mathsf{sn}'_S, \mathsf{cm}'_S)$ from the updated sender state, and publishes

$$\mathsf{tx}_{\mathsf{Send}} = (pk_S, \mathsf{sn}_S, \mathsf{cm}'_S, \mathsf{td}_{\mathsf{pp}}, \mathsf{aux}_{\mathsf{pp}}, \alpha_{\mathsf{pp}}, \pi_{\mathsf{send}}).$$

The relation $\mathcal{R}_{\mathsf{Send}}$ enforces validity of the consumed sender state, correctness of the updated sender state, and consistency of $(\mathsf{td}_{\mathsf{pp}}, \mathsf{aux}_{\mathsf{pp}}, \alpha_{\mathsf{pp}})$, including the binding of pk_R in the transfer digest and ciphertext.

In $\mathsf{Deposit}$, the receiver recovers τ_{pp} from $\mathsf{aux}_{\mathsf{pp}}$, where ρ is recovered as part of τ_{pp}, derives

$$\mathsf{sn}_v = H(sk_R, \rho),$$

sets $b'_R = b_R + v$, derives $(\mathsf{sn}'_R, \mathsf{cm}'_R)$ from the updated receiver state, and publishes

$$\mathsf{tx}_{\mathsf{Deposit}} = (\mathsf{root}_t, \mathsf{sn}_R, \mathsf{cm}'_R, \mathsf{sn}_v, pk_R, \pi_{\mathsf{deposit}}).$$

The relation $\mathcal{R}_{\mathsf{Deposit}}$ enforces consistency of the decrypted tuple and $\mathsf{td}_{\mathsf{pp}}$, membership of $\mathsf{td}_{\mathsf{pp}}$ under root_t, correctness of sn_v, and correctness of the updated receiver state.

Public-to-private transfer. In $\mathsf{Mint}^\star$, provided that the sender has sufficient public balance, the sender samples $\rho \xleftarrow{\$} \{0,1\}^{\ell_H}$, defines

$$\tau^\star_{\mathsf{mi}} = (\mathsf{addr}_S, v, pk_R, \rho), \qquad \mathsf{td}^\star_{\mathsf{mi}} = \mathsf{Com}(\tau^\star_{\mathsf{mi}}),$$

$$\mathsf{aux}^\star_{\mathsf{mi}} = \mathsf{Enc}(pk_R, (\mathsf{addr}_S, \rho)), \qquad \alpha^\star_{\mathsf{mi}} = H(sk_S, H(\mathsf{aux}^\star_{\mathsf{mi}})),$$

and publishes

$$\mathsf{tx}_{\mathsf{Mint}^\star} = (\mathsf{addr}_S, v, \mathsf{td}^\star_{\mathsf{mi}}, \mathsf{aux}^\star_{\mathsf{mi}}, \alpha^\star_{\mathsf{mi}}, \pi_{\mathsf{mint}^\star}).$$

The relation $\mathcal{R}_{\mathsf{Mint}^\star}$ enforces consistency of the transfer digest and authentication tag. In $\mathsf{Dep\_Mi}^\star$, the receiver decrypts $\mathsf{aux}^\star_{\mathsf{mi}}$ to recover (addr_S, ρ), derives

$$\mathsf{sn}_v = H(sk_R, \rho),$$

updates its private balance to $b_R + v$, derives $(\mathsf{sn}'_R, \mathsf{cm}'_R)$ from the updated receiver state, and publishes

$$\mathsf{tx}_{\mathsf{Dep\_Mi}^\star} = (\mathsf{root}_t, \mathsf{sn}_R, \mathsf{cm}'_R, \mathsf{sn}_v, pk_R, \pi_{\mathsf{depmi}^\star}).$$

The relation $\mathcal{R}_{\mathsf{Dep\_Mi}^\star}$ enforces consistency of $\mathsf{td}^\star_{\mathsf{mi}}$, membership under root_t, freshness of sn_v, and correctness of the updated receiver state.

Private-to-public transfer. In $\mathsf{Redeem}^\star$, the sender samples $r'_S, \rho \xleftarrow{\$} \{0,1\}^{\ell_H}$, defines

$$\tau^\star_{\mathsf{re}} = (\mathsf{addr}_S, v, pk_R, \rho, \mathsf{sn}_S), \qquad \mathsf{td}^\star_{\mathsf{re}} = \mathsf{Com}(\tau^\star_{\mathsf{re}}),$$

$$\mathsf{aux}^\star_{\mathsf{re}} = \mathsf{Enc}(pk_R, \tau^\star_{\mathsf{re}}), \qquad \alpha^\star_{\mathsf{re}} = H(sk_S, H(\mathsf{aux}^\star_{\mathsf{re}})),$$

sets the sender private balance to $b'_S = b_S - v$, derives $(\mathsf{sn}'_S, \mathsf{cm}'_S)$ from the updated sender state, and publishes

$$\mathsf{tx}_{\mathsf{Redeem}^\star} = (\mathsf{addr}_S, \mathsf{sn}_S, \mathsf{aux}^\star_{\mathsf{re}}, \alpha^\star_{\mathsf{re}}, \mathsf{td}^\star_{\mathsf{re}}, \mathsf{cm}'_S, \pi_{\mathsf{redeem}^\star}).$$

The relation $\mathcal{R}_{\mathsf{Redeem}^\star}$ enforces validity of the consumed sender state, correctness of the updated sender state, and consistency of the published transfer objects. In $\mathsf{Dep\_Re}^\star$, the receiver decrypts $\mathsf{aux}^\star_{\mathsf{re}}$, where ρ is recovered from the decrypted tuple, derives

$$\mathsf{sn}_v = H(sk_R, \rho),$$

and publishes

$$\mathsf{tx}_{\mathsf{Dep\_Re}^\star} = (\mathsf{root}_t, v, \mathsf{sn}_v, pk_R, \pi_{\mathsf{depre}^\star}).$$

The relation $\mathcal{R}_{\mathsf{Dep\_Re}^\star}$ enforces consistency of $\mathsf{td}^\star_{\mathsf{re}}$, membership under root_t, and correctness of the claim serial number.

4.4 State Transition Rules

Let

$$\mathsf{State}_t = (\mathsf{Ledger}_t, \mathsf{PubBal}_t, \mathsf{PriState}_t)$$

be the public state before processing a transaction tx of type $T \in \mathcal{T}$. Let x and π denote the public statement and proof extracted from tx. The ledger accepts tx only if $\mathsf{Verify}(\mathsf{vk}_T, x, \pi) = 1$, every serial number revealed by tx is fresh with respect to SNSet_t, every consumed private-state digest equals the current record in $\mathsf{PriState}_t$, every claim transaction proves membership of the transfer digest used in the claim relation under the claimed Merkle root, and every public-balance debit satisfies $\mathsf{PubBal}_t[\mathsf{addr}] \geq v$. For every accepted transaction, the ledger appends tx to Ledger_t and updates SNSet_t and TCSet_t with the serial numbers and transfer digests published by tx.

The transaction-specific updates are as follows. Mint debits $\mathsf{PubBal}[\mathsf{addr}]$ by v, replaces $\mathsf{PriState}[\mathsf{addr}]$ with cm', and publishes sn. Redeem credits $\mathsf{PubBal}[\mathsf{addr}]$ by v, replaces $\mathsf{PriState}[\mathsf{addr}]$ with cm', and publishes sn. Send replaces $\mathsf{PriState}[\mathsf{addr}_S]$ with cm'_S, publishes sn_S, and inserts $\mathsf{td}_{\mathsf{pp}}$ into the transfer-digest set. Deposit replaces $\mathsf{PriState}[\mathsf{addr}_R]$ with cm'_R and publishes $(\mathsf{sn}_R, \mathsf{sn}_v)$. $\mathsf{Mint}^\star$ debits $\mathsf{PubBal}[\mathsf{addr}_S]$ by v and inserts $\mathsf{td}^\star_{\mathsf{mi}}$ into the transfer-digest set. $\mathsf{Dep\_Mi}^\star$ replaces $\mathsf{PriState}[\mathsf{addr}_R]$ with cm'_R and publishes $(\mathsf{sn}_R, \mathsf{sn}_v)$. $\mathsf{Redeem}^\star$ replaces $\mathsf{PriState}[\mathsf{addr}_S]$ with cm'_S, publishes sn_S, and inserts $\mathsf{td}^\star_{\mathsf{re}}$ into the transfer-digest set. $\mathsf{Dep\_Re}^\star$ credits $\mathsf{PubBal}[\mathsf{addr}_R]$ by v and publishes sn_v.

5 Security Analysis

5.1 Security Scope

We analyze UBM under the cryptographic assumptions defined in Section 2 and the ledger and adversarial models defined in Section 3, following the BlockMaze security framework [12]. Because the transactions

$$\mathsf{Mint},\ \mathsf{Redeem},\ \mathsf{Send},\ \mathsf{Deposit}$$

are inherited without modification, their security analysis is unchanged. We therefore restrict the analysis to the additional transaction pairs

$$(\mathsf{Mint}^\star, \mathsf{Dep\_Mi}^\star) \quad \text{and} \quad (\mathsf{Redeem}^\star, \mathsf{Dep\_Re}^\star).$$

The analysis is preservation-oriented. For each security goal, we argue that a successful attack with non-negligible advantage or probability would require breaking a cryptographic assumption defined in Section 2 or bypassing a ledger validation rule defined in Section 3.

5.2 Ledger Indistinguishability and Transaction Unlinkability

Ledger indistinguishability. In the sender-side transactions

$$\mathsf{Mint}^\star \quad \text{and} \quad \mathsf{Redeem}^\star,$$

the public transcript contains the sender-side public fields, the transfer digest, the ciphertext, the authentication tag, and the proof. The designated receiver, transfer randomness, private-state opening, and remaining relation witnesses are not published. A transfer digest is a hash-based commitment defined in Section 4, such as $\mathsf{td}^\star_{\mathsf{mi}}$ or $\mathsf{td}^\star_{\mathsf{re}}$.

In the receiver-side claim transactions

$$\mathsf{Dep\_Mi}^\star \quad \text{and} \quad \mathsf{Dep\_Re}^\star,$$

the public transcript contains the Merkle root, the revealed serial numbers, the applicable public state-update fields, and the proof. The claimed transfer digest, its opening, and the corresponding Merkle authentication path remain part of the proof witness.

Suppose that a probabilistic polynomial-time adversary obtains hidden transfer data, identifies the receiver of a sender-side ciphertext, recovers information about a hidden digest opening, derives an honest account's secret value from a serial number, or obtains witness information from a proof with non-negligible probability. These events violate IND-CCA security, IK-CCA security, computational hiding of the hash-based commitment, pseudorandomness of the keyed hash evaluation, or zero knowledge, respectively. Therefore, under the assumptions of Section 2, UBM preserves ledger indistinguishability.

Transaction unlinkability. The cross-boundary workflows of UBM separate sender-side transfer generation from receiver-side claim. In

$$\mathsf{Mint}^{\star} \quad \text{and} \quad \mathsf{Redeem}^{\star},$$

the receiver public key and transfer randomness are bound to the transfer digest and protected by the ciphertext rather than exposed as sender-side public fields. In

$$\mathsf{Dep\_Mi}^{\star} \quad \text{and} \quad \mathsf{Dep\_Re}^{\star},$$

the receiver proves membership of the transfer digest under a Merkle root without publishing the transfer digest, its opening, or its authentication path. The claim serial number

$$\mathsf{sn}_v = H(sk_R, \rho)$$

is pseudorandom for an honest receiver and does not expose ρ or provide a public identifier shared with the sender-side transaction.

Suppose that a probabilistic polynomial-time adversary links a receiver-side claim transaction to its corresponding sender-side transaction with non-negligible advantage. Such linkage must distinguish the receiver key associated with the sender-side ciphertext, recover or test the hidden opening of the transfer digest, recover information hidden by the claim proof, or distinguish the claim serial number from a pseudorandom value. These events violate IK-CCA security, computational hiding of the hash-based commitment, zero knowledge, or pseudorandomness of the keyed hash evaluation, respectively. Therefore, UBM preserves transaction unlinkability across the public/private boundary.

5.3 Transaction Non-Malleability

Under the transaction non-malleability goal defined in Section 3, a successful adversary must produce a distinct accepted transaction that reuses one-time public data from a previously accepted transaction. For the additional transactions, the relevant one-time data include the consumed sender serial number when present, the transfer digest, the ciphertext and authentication-tag pair, and the claim serial number derived from the hidden transfer randomness.

The ledger rejects any candidate transaction that reuses a revealed sender serial number or claim serial number. Consider instead an adversary that modifies a transaction while retaining a previously published transfer digest. By simulation extractability, an accepted statement-proof pair produced by the adversary admits extraction of a valid witness, except with negligible probability. If the modified transaction binds different transfer data to the same digest, the extracted witness and the original opening provide two distinct openings of the same hash-based commitment, contradicting computational binding and collision resistance. If the modified public statement has no valid witness, its accepted proof contradicts knowledge soundness or simulation extractability.

Similarly, modifying a ciphertext while retaining a valid authentication tag requires predicting

$$H(sk_S, H(\mathsf{aux}))$$

for an honest sender without knowing sk_S, contradicting pseudorandomness of the keyed hash evaluation. Therefore, any probabilistic polynomial-time adversary succeeds in producing a malleated accepted transaction only with negligible probability, and the additional transactions preserve transaction non-malleability.

5.4 Balance Correctness

Balance correctness requires that no probabilistic polynomial-time adversary can make the ledger accept a state transition that creates value, consumes the same private state more than once, or claims the same transfer more than once.

For

$$(\mathsf{Mint}^\star, \mathsf{Dep\_Mi}^\star),$$

the ledger accepts $\mathsf{Mint}^\star$ only if the sender has sufficient public balance. The accepted transaction debits the sender public balance by v and binds the same value v to a transfer digest. An accepted $\mathsf{Dep\_Mi}^\star$ transaction proves membership of that transfer digest under a Merkle root associated with an accepted ledger prefix and updates the receiver private balance by the value bound to the digest. Knowledge soundness ensures that an accepted proof admits a witness satisfying the transaction relation, computational binding prevents the value bound to the transfer digest from being changed, and Merkle membership integrity prevents a claim for a transfer digest absent from the accepted ledger prefix.

For

$$(\mathsf{Redeem}^\star, \mathsf{Dep\_Re}^\star),$$

an accepted $\mathsf{Redeem}^\star$ transaction consumes the current sender private state, decreases the sender private balance by v, and binds the same value v to a transfer digest. An accepted $\mathsf{Dep\_Re}^\star$ transaction proves membership of that transfer digest and credits the receiver public balance by the value bound to the digest. Knowledge soundness and computational binding ensure consistency between the sender-side debit, the transfer digest, and the receiver-side credit.

The ledger prevents repeated consumption of a private state by requiring every consumed private-state digest to equal the current record in $\mathsf{PriState}_t$ and by enforcing freshness of the corresponding sender serial number. It prevents repeated claims because the same transfer opening and receiver secret key deterministically produce the same claim serial number

$$\mathsf{sn}_v = H(sk_R, \rho),$$

and the ledger accepts that serial number at most once. Producing a different accepted claim serial number for the same transfer requires a different valid opening of the transfer digest or an inconsistent proof witness, contradicting computational binding or knowledge soundness.

Consequently, each accepted sender-side transaction removes v from the sender balance and authorizes at most one receiver-side claim of the same value. Any violation of value conservation, private-state freshness, or claim uniqueness contradicts the cryptographic assumptions of Section 2 or the deterministic ledger validation rules of Section 3. Therefore, UBM preserves balance correctness.

6 Experimental Evaluation

6.1 Evaluation Setup

We use the original BlockMaze workflow as the quantitative baseline because UBM modifies its transaction structure while retaining the same account-state representation, zk-SNARK mechanism, cryptographic components, and implementation environment. We implement UBM by extending the public BlockMaze implementation[1] only with the additional transactions required by UBM. This controlled comparison isolates the effect of transaction restructuring on cross-boundary payment workflows. Systems such as Zether, QuisQuis, and LedgerMaze differ from BlockMaze in at least one of their state-transition semantics, privacy goals, proof systems, or implementation environments [5, 9, 2]. Consequently, we do not use their published performance results as quantitative baselines without a common reimplementation.

We conducted all experiments on a laptop equipped with a 13th-generation Intel Core i5-13500H processor and 16 GB of RAM, running 64-bit Ubuntu 18.04.1. Our implementation uses the Groth16 proof system [11] over `ALT_BN128` through `libsnark`. It uses SHA-256 as the hash function and the Elliptic Curve Integrated Encryption Scheme (ECIES) for public-key encryption.

We consider four metrics: the number of circuit constraints, proof generation time, the number of SHA-256 hash operations in the proof circuit, and transaction size. The number of circuit constraints and SHA-256 hash operations measure circuit-level prover cost, proof generation time measures prover-side execution time, and transaction size measures communication and ledger-storage overhead.

6.2 Evaluation Workflows

Since UBM modifies only cross-boundary payment workflows, we evaluate two representative workflows and exclude the unchanged private-to-private workflow

$$\mathsf{Send} \rightarrow \mathsf{Deposit}.$$

For public-to-private transfer, the original workflow is

$$\mathsf{Mint} \rightarrow \mathsf{Send} \rightarrow \mathsf{Deposit},$$

whereas UBM uses

$$\mathsf{Mint}^{\star} \rightarrow \mathsf{Dep\_Mi}^{\star}.$$

For private-to-public transfer, the original workflow is

$$\mathsf{Send} \rightarrow \mathsf{Deposit} \rightarrow \mathsf{Redeem},$$

whereas UBM uses

$$\mathsf{Redeem}^{\star} \rightarrow \mathsf{Dep\_Re}^{\star}.$$

Thus, in both cases, the original conversion-plus-transfer workflow is replaced by a two-phase sender-side transfer and receiver-side claim workflow.

6.3 Transaction-Level Cost Analysis

Table 1 summarizes the transaction-level circuit constraints, hash counts, and proof generation times of the modified cross-boundary workflows. For public-to-private transfer, the three columns Original/Phase 1/Phase 2 correspond to

$$\mathsf{Mint},\ \mathsf{Mint}^{\star},\ \mathsf{Dep\_Mi}^{\star},$$

[1] https://github.com/Agzs/BlockMaze/tree/paper/libsnark-vnt

while for private-to-public transfer they correspond to

$$\mathsf{Redeem},\ \mathsf{Redeem}^{\star},\ \mathsf{Dep\_Re}^{\star}.$$

The unchanged baseline transactions are Send with 252,286 constraints and 7 SHA-256 hash operations, and Deposit with 503,863 constraints and $6 + \mathsf{MT}$ SHA-256 hash operations. Their proof generation times are 5.73 seconds and 11.20 seconds, respectively.

Table 1: Transaction-level circuit constraints, hash counts, and proof generation times for the modified cross-boundary workflows, where MT denotes the Merkle-path verification term.

Metric	Flow	Original	Phase 1	Phase 2
Constraints	Pub→Pri	167,270	84,019	503,607
	Pri→Pub	167,853	252,286	336,467
Hash count	Pub→Pri	4	3	$6 + \mathsf{MT}$
	Pri→Pub	4	7	$2 + \mathsf{MT}$
Proof generation	Pub→Pri	4.32	2.27	11.35
time (s)	Pri→Pub	4.39	5.76	8.64

The results show a redistribution of transaction-level prover cost rather than a uniform reduction. In the public-to-private direction, $\mathsf{Mint}^{\star}$ incurs lower transaction-level prover cost than Mint because it omits the sender private-state update, whereas $\mathsf{Dep\_Mi}^{\star}$ performs transfer claiming, Merkle-path verification, claim-serial derivation, and receiver private-state update. In the private-to-public direction, $\mathsf{Redeem}^{\star}$ incurs higher transaction-level prover cost than Redeem because it combines the sender private-state update with transfer creation, whereas $\mathsf{Dep\_Re}^{\star}$ incurs lower transaction-level prover cost than Deposit because it credits a public balance without constructing a new private-state digest. The Merkle-path verification term remains in the receiver-side claim phase in both UBM workflows. Therefore, transaction-level cost identifies where prover work is placed but does not by itself determine the efficiency of the complete payment workflow.

6.4 Workflow-Level Efficiency

We next quantify the workflow-level proof generation time of the representative workflows defined in Section 6.2.

Figure 2 reports the total proof generation time of the two representative cross-boundary payment workflows. For public-to-private transfer, the total proof generation time decreases from 21.25 seconds for BlockMaze to 13.62 seconds for UBM, corresponding to a reduction of 7.63 seconds, or 35.9%. For private-to-public transfer, it decreases from 21.32 seconds to 14.40 seconds, corresponding to a reduction of 6.92 seconds, or 32.5%. Thus, although UBM does not uniformly reduce every individual transaction cost, it reduces total proof generation time at the workflow level in both representative workflows.

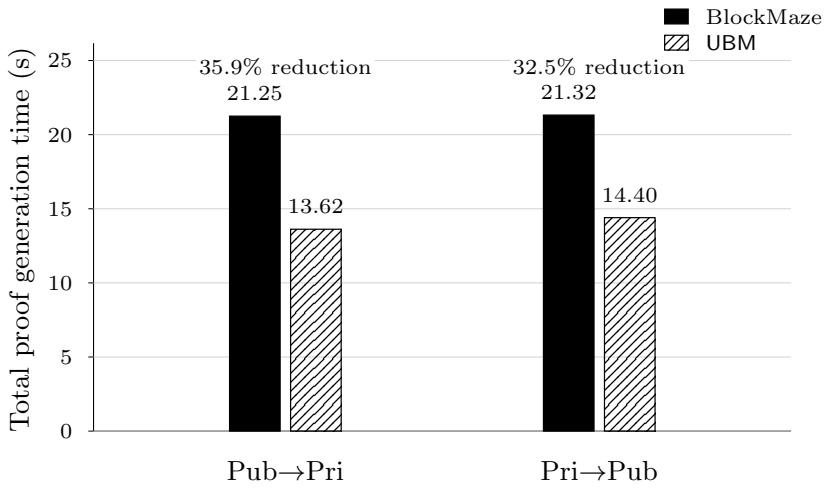


Figure 2: Total proof generation time for the representative cross-boundary workflows. The annotations report the relative reduction achieved by UBM over BlockMaze.

6.5 Communication and Storage Overhead

We finally examine whether the reduction in total proof generation time is offset by communication or ledger-storage overhead.

Table 2 reports the cumulative serialized transaction size of the same representative workflows.

Table 2: Workflow-level transaction size for the representative cross-boundary transfers. The reduction column applies only to the fixed-size component; aux and seq are variable-size terms.

Flow	BlockMaze	UBM	Fixed-size reduction
Pub→Pri	$808 + \mathsf{aux} + \mathsf{seq}$	$520 + \mathsf{aux} + \mathsf{seq}$	288 B (35.6%)
Pri→Pub	$808 + \mathsf{aux} + \mathsf{seq}$	$512 + \mathsf{aux} + \mathsf{seq}$	296 B (36.6%)

For public-to-private transfer, the fixed-size component decreases from 808 bytes to 520 bytes, a reduction of 288 bytes, or 35.6%. For private-to-public transfer, it decreases from 808 bytes to 512 bytes, a reduction of 296 bytes, or 36.6%. Within each workflow direction, the BlockMaze and UBM expressions contain the same variable-size terms aux and seq. Hence, their cumulative serialized transaction sizes differ by 288 bytes and 296 bytes, respectively; the reported percentages apply only to the fixed-size component. Thus, the reduction in total proof generation time is accompanied by a smaller cumulative serialized transaction size.

7 Conclusion

This paper revisited BlockMaze from a transaction-structure perspective and identified a structural limitation in its treatment of cross-boundary payments: direct public/private balance conversion couples the two balance domains more tightly than necessary. To address this issue, we proposed UBM, which extends BlockMaze with four transaction types and restructures the public-to-private and private-to-public workflows into sender-side transfer and receiver-side claim phases.

The security analysis showed that UBM preserves ledger indistinguishability, transaction unlinkability, transaction non-malleability, and balance correctness under the cryptographic and ledger assumptions inherited from BlockMaze. The experimental evaluation showed that, for the two representative cross-boundary payment workflows, UBM achieves lower total proof generation time and smaller cumulative serialized transaction size under the same implementation setting. These results show that transaction structure affects workflow-level efficiency in addition to proof-system design.

Limitations and future work. The reported results apply to the BlockMaze design space and do not establish a universal performance advantage over account-based privacy systems based on different state models or proof systems. The evaluation covers only two representative cross-boundary payment workflows and does not constitute a deployment-level evaluation. Deploying UBM requires validator support for the four additional transaction types and their verification keys, together with wallet support for detecting unclaimed transfers and generating receiver-side claims. Future work will implement and evaluate these components under deployment workloads and compare UBM with other account-based privacy systems under a common implementation setting.

Acknowledgments

This work was supported by the National Science and Technology Council, Taiwan, under Project No. NSTC 114-2634-F-004-002-MBK.

References

[1] Foteini Baldimtsi, Kostas Kryptos Chalkias, Varun Madathil, and Arnab Roy. SoK: Privacy-preserving transactions in blockchains. Cryptology ePrint Archive, Paper 2024/1959, 2024.

[2] Zijian Bao, Debiao He, Wei Wei, Cong Peng, and Xinyi Huang. LedgerMaze: An efficient privacy-preserving noninteractive zero-knowledge scheme over account-model blockchain. *IEEE Transactions on Computers*, 72(12):3489–3502, 2023.

[3] Sean Bowe, Alessandro Chiesa, Matthew Green, Ian Miers, Pratyush Mishra, and Howard Wu. Zexe: Enabling decentralized private computation. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 947–964. IEEE, 2020.

[4] Lars Brünjes and Murdoch J Gabbay. UTxO-vs account-based smart contract blockchain programming paradigms. In *International Symposium on Leveraging Applications of Formal Methods*, pages 73–88. Springer, 2020.

[5] Benedikt Bünz, Shashank Agrawal, Mahdi Zamani, and Dan Boneh. Zether: Towards privacy in a smart contract world. In *International Conference on Financial Cryptography and Data Security*, pages 423–443. Springer, 2020.

[6] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *2018 IEEE symposium on security and privacy (SP)*, pages 315–334. IEEE, 2018.

[7] Vitalik Buterin et al. A next-generation smart contract and decentralized application platform. *white paper*, 3(37):2–1, 2014.

[8] Alessandro Chiesa, Yuncong Hu, Mary Maller, Pratyush Mishra, Noah Vesely, and Nicholas Ward. Marlin: Preprocessing zkSNARKs with universal and updatable SRS. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 738–768. Springer, 2020.

[9] Prastudy Fauzi, Sarah Meiklejohn, Rebekah Mercer, and Claudio Orlandi. QuisQuis: A new design for anonymous cryptocurrencies. In *International conference on the theory and application of cryptology and information security*, pages 649–678. Springer, 2019.

[10] Lorenzo Grassi, Dmitry Khovratovich, Christian Rechberger, Arnab Roy, and Markus Schofnegger. Poseidon: A new hash function for Zero-Knowledge proof systems. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 519–535, 2021.

[11] Jens Groth. On the size of pairing-based non-interactive arguments. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 305–326. Springer, 2016.

[12] Zhangshuang Guan, Zhiguo Wan, Yang Yang, Yan Zhou, and Butian Huang. BlockMaze: An efficient privacy-preserving account-model blockchain based on zk-SNARKs. *IEEE Transactions on Dependable and Secure Computing*, 19(3):1446–1463, 2020.

[13] Ahmed Kosba, Andrew Miller, Elaine Shi, Zikai Wen, and Charalampos Papamanthou. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *2016 IEEE symposium on security and privacy (SP)*, pages 839–858. IEEE, 2016.

[14] Ian Miers, Christina Garman, Matthew Green, and Aviel D Rubin. Zerocoin: Anonymous distributed e-cash from Bitcoin. In *2013 IEEE symposium on security and privacy*, pages 397–411. IEEE, 2013.

[15] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[16] Shen Noether. Ring signature confidential transactions for Monero. Cryptology ePrint Archive, Paper 2015/1098, 2015.

[17] Shen Noether, Adam Mackenzie, et al. Ring confidential transactions. *Ledger*, 1:1–18, 2016.

[18] Alexey Pertsev, Roman Semenov, and Roman Storm. Tornado Cash privacy solution version 1.4. *Tornado cash privacy solution version*, 1(6), 2019.

[19] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from Bitcoin. In *2014 IEEE symposium on security and privacy*, pages 459–474. IEEE, 2014.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 17–37.

Development of Identity-based Proxy Re-encryption Library

Tang-Xing Cheang[1], Swee-Huay Heng[2] and Author CSyh-Yuan Tan[1]

[1] Faculty of Information Science and Technology, Multimedia University, Jalan Ayer Keroh Lama, 75450 Bukit Beruang, Melaka, Malaysia. {cheangtangxing,sytansy}@gmail.com

[2] Centre for Intelligent Cloud Computing, CoE for Advanced Cloud, Faculty of Information Science and Technology, Multimedia University, Jalan Ayer Keroh Lama, 75450 Bukit Beruang, Melaka, Malaysia. shheng@mmu.edu.my

Abstract. Cloud storage services have become a popular solution for storing and sharing data in today's digital environment. However, security concerns arise when users rely on semi-trusted third parties to manage and store sensitive data. Although encryption protects data confidentiality in the cloud, traditional encryption methods still face challenges in secure key sharing and access delegation. If the threats posed by semi-trusted third parties are ignored, data privacy may be compromised. Identity-Based Proxy Re-Encryption (IB-PRE) provides a practical approach to address these challenges. IB-PRE allows data owners to securely share encrypted data by enabling key delegation without exposing private keys or requiring direct interaction with the data owner. Unlike traditional encryption schemes, IB-PRE supports secure data sharing through a semi-trusted cloud proxy while ensuring that only authorised recipients can decrypt the data. In this paper, we enhance the IB-PRE scheme proposed by Green and Ateniese by converting it from Type-1 to Type-3 pairings. We employ the commit-to-point technique to address the technical challenge of preserving the algebraic relationship originally provided by symmetric hash-to-point functions. This improves implementation practicality while maintaining decryption correctness. We further demonstrate the practicality of the enhanced IB-PRE scheme through a file sharing web application. Our benchmark shows that first-level and second-level decryption each take less than 7 ms, while the overall protocol can be completed within 43 ms.

Keywords: Identity-based Proxy Re-encryption · Type-3 Pairings · File Sharing System · Cryptographic Library

1 Introduction

Cloud storage has become a common platform for storing and sharing data. However, when data is stored in a semi-trusted cloud server, users may lose direct control over their sensitive files. Although encryption protects data confidentiality, traditional encryption still faces difficulties in secure key sharing and access delegation. For example, when a data owner wants to share encrypted data with another user, the owner may need to decrypt and re-encrypt the data or manually distribute secret keys. This approach is inefficient and may expose sensitive information.

Proxy re-encryption (PRE) provides a useful solution by allowing a semi-trusted proxy to transform a ciphertext encrypted under one user's key into a ciphertext decryptable by another user, without learning the plaintext [4]. Identity-based proxy re-encryption (IB-PRE) further improves this idea by using user identities, such as email addresses, as

public keys. This removes the need for certificate management and makes IB-PRE suitable for cloud-based file sharing systems [11].

The IB-PRE scheme proposed by Green and Ateniese is an important foundation for identity-based encrypted delegation [11]. However, the original construction is based on Type-1 pairings, where both pairing inputs belong to the same group. This symmetry simplifies the construction, but Type-1 pairings are less attractive for modern implementation compared with asymmetric Type-3 pairings [20]. In Type-3 pairings, the two source groups are different and there is no efficiently computable isomorphism between them. Therefore, the original Type-1 hash-to-point structure cannot be directly applied, because it may break the algebraic relationship required for correct decryption.

Existing PRE and IB-PRE studies have improved encrypted data sharing in different directions, including conditional re-encryption, broadcast sharing, blockchain-assisted storage, IoT sharing, anonymity, and revocation [16]. However, these works do not directly solve the specific problem of converting the Green and Ateniese Type-1 pairing-based IB-PRE construction into a Type-3 pairing setting while preserving decryption correctness. This creates a clear research gap between classical IB-PRE design and practical modern Type-3 pairing implementation.

This paper addresses this gap by developing a Type-3 pairing-based IB-PRE construction. The proposed construction applies the commit-to-point technique to preserve the required algebraic consistency in the Type-3 setting [6]. We further implement the scheme as a Rust-based cryptographic library and integrate it with AES-based hybrid encryption for practical secure file sharing.

Contribution. The main contributions of this paper are summarized as follows.

1. **Type-3 IB-PRE construction.** We convert the Green and Ateniese Type-1 pairing-based IB-PRE scheme into a Type-3 pairing setting to improve its practicality for modern pairing-based implementation.

2. **Correctness preservation using commit-to-point.** We apply the commit-to-point technique [6] to address the loss of symmetry in Type-3 pairings and preserve the correctness of both first-level and second-level decryption.

3. **Rust-based cryptographic library.** We implement the proposed Type-3 IB-PRE scheme as a Rust cryptographic library using the Irmaseal curve library. The library supports setup, key generation, encryption, re-encryption, and decryption.

4. **Hybrid AES and Type-3 IB-PRE file encryption.** We combine AES with Type-3 IB-PRE, where AES encrypts the file content and Type-3 IB-PRE protects and delegates the AES key. This improves the practicality of the scheme for file sharing applications.

5. **Prototype implementation and evaluation.** We develop a secure file sharing web application and evaluate its functionality and performance through system testing and cryptographic benchmarking.

Organization. The rest of this paper is organized as follows. Section 2 reviews related work on PRE, IB-PRE, secure cloud sharing, and Type-3 pairing-based constructions. Section 3 introduces the required preliminaries, including bilinear maps, the DBDH assumption, IB-PRE algorithms, and the security model. Section 4 presents the proposed Type-3 IB-PRE construction, including the system model, assumptions, algorithms, correctness analysis, and security discussion. Section 5 describes the hybrid AES and Type-3 IB-PRE file encryption system, implementation, testing, and performance evaluation. Section 6 concludes the paper.

2 Related Work

Proxy re-encryption (PRE) was first introduced by Blaze, Bleumer and Strauss, where a semi-trusted proxy can transform a ciphertext encrypted under one public key into a ciphertext decryptable under another public key without learning the plaintext [4]. This idea provides an important foundation for delegated access control in encrypted storage systems. However, the early construction was mainly bidirectional, meaning that delegation could work in both directions between two users. Although this design is simple, it may introduce security risks when the proxy colludes with one of the users.

Ateniese et al. later proposed improved PRE schemes for secure distributed storage [3]. Their work introduced more practical properties such as unidirectional delegation, where a re-encryption key only supports transformation from the delegator to the delegatee. This makes PRE more suitable for cloud storage because encrypted data can be stored by an untrusted server and shared through ciphertext transformation. Nevertheless, traditional PRE schemes still rely on ordinary public keys and certificate management, which may be inconvenient in large cloud-based systems.

Identity-based encryption (IBE) was introduced by Shamir to simplify public key management by allowing a user's identity, such as an email address, to act as the public key [18]. Boneh and Franklin later proposed a practical pairing-based IBE scheme using bilinear pairings [5]. IBE reduces the need for certificate-based public key infrastructure, but it also introduces the key escrow problem because a trusted Private Key Generator (PKG) can generate the private key of any user.

Green and Ateniese combined the advantages of IBE and PRE by proposing the first identity-based proxy re-encryption scheme [11]. Their IB-PRE scheme allows a proxy to convert a ciphertext encrypted under one identity into a ciphertext decryptable under another identity without learning the plaintext. The scheme supports encrypted delegation while avoiding certificate management. However, the original construction is based on Type-1 pairings, where both pairing inputs belong to the same group. This symmetric structure simplifies the design, but it is less attractive for modern implementation compared with Type-3 pairings.

Recent PRE and IB-PRE studies have improved encrypted data sharing in different directions. Shao et al. proposed an identity-based conditional proxy re-encryption scheme to support fine-grained delegation based on specific conditions [19]. He et al. proposed an efficient IB-PRE scheme for blockchain-assisted decentralized storage, showing that IB-PRE can support secure data sharing in distributed environments [12]. Other recent works focus on broadcast sharing, dynamic users, revocation, anonymity, and IoT applications. For example, Hu et al. proposed identity-based broadcast PRE with dynamic functionality for flexible cloud data sharing [13], Zhang et al. applied identity-based broadcast PRE to vehicular ad hoc networks [22], Maiti et al. proposed a coalitional-game-based broadcast PRE scheme for IoT [16], Li et al. proposed a pairing-free identity-based broadcast PRE scheme for cloud environments [14], and Chen et al. introduced conditional identity-based broadcast PRE with anonymity and revocation [7].

Beyond PRE and IB-PRE, recent secure data sharing studies have also explored blockchain-enabled data governance and searchable public key encryption. Zhang and Datta studied blockchain-enabled data governance for privacy-preserved sharing of confidential data [21]. Li et al. proposed a secure data sharing scheme based on searchable public key encryption for authorised multi-receiver settings [15]. These studies show the broader research trend toward privacy-preserved and access-controlled data sharing, while our work focuses specifically on the Type-3 conversion and implementation of the Green and Ateniese IB-PRE construction.

Modern PRE research has also moved toward asymmetric pairing settings. Abe et al. studied the conversion of Type-1 pairing-based designs into Type-3 settings and showed that Type-3 pairings are important for fast and scalable pairing-based implementation [2].

Zengin et al. proposed a chosen-ciphertext secure unidirectional PRE scheme based on asymmetric pairings, further showing the practical relevance of Type-3 pairings in proxy re-encryption research [20]. However, these works do not directly address the conversion of the Green and Ateniese Type-1 IB-PRE construction into a Type-3 setting while preserving the decryption correctness of both first-level and second-level ciphertexts.

Therefore, the research gap addressed in this paper is the practical Type-3 conversion of the Green and Ateniese IB-PRE scheme [11]. Unlike works that focus mainly on blockchain integration, broadcast functionality, revocation, anonymity, or pairing-free design, this paper focuses on preserving the algebraic correctness of the original IB-PRE construction under Type-3 pairings using the commit-to-point technique [6], and then implementing the scheme as a Rust-based cryptographic library for secure file sharing.

Table 1: Comparison of Existing PRE and IB-PRE Schemes

Scheme	Main Difference	Advantage	Computational Cost	Limitation
Blaze et al. [4]	Traditional PRE	Introduces proxy-based ciphertext transformation	Public-key operations	Bidirectional and no identity-based key management
Ateniese et al. [3]	Improved public-key PRE	Supports unidirectional delegation	Public-key operations	Still requires certificate management
Green and Ateniese [11]	Type-1 IB-PRE	Removes certificate management and supports identity-based delegation	Pairing and exponentiation operations	Type-1 pairing is less practical for modern implementation
Shao et al. [19]	Conditional IB-PRE	Supports fine-grained delegation	Pairing operations with condition checking	Extra policy and condition management cost
He et al. [12]	Blockchain-assisted IB-PRE	Supports decentralized secure storage	Pairing operations plus blockchain overhead	Higher storage and management cost
Hu et al. [13]	Dynamic broadcast IB-PRE	Supports flexible cloud data sharing	Cost increases with dynamic users	More complex access management
Zhang et al. [22]	Broadcast IB-PRE for VANETs	Supports flexible sharing in vehicular networks	Cost increases with multiple receivers	Not designed for general cloud file sharing
Maiti et al. [16]	Broadcast PRE for IoT	Supports collaborative IoT data sharing	Cost increases with IoT users and broadcast setting	Does not solve Type-3 IB-PRE conversion
Zengin et al. [20]	Type-3 public-key PRE	Uses asymmetric pairings and supports CCA security	Type-3 pairing operations	Does not address Green-Ateniese IB-PRE conversion
Li et al. [14]	Pairing-free broadcast IB-PRE	Reduces pairing computation	Non-pairing cryptographic operations	Different design from Type-3 pairing conversion
Chen et al. [7]	Conditional broadcast IB-PRE	Supports anonymity and revocation	Higher cost due to extra privacy features	More complex implementation
Proposed Scheme	Type-3 IB-PRE conversion	Preserves IB-PRE delegation and improves practical implementation	Type-3 pairing operations with millisecond-level execution time	Requires trusted PKG and pairing operations

3 Preliminaries

3.1 Bilinear Map

The cryptographic basis of IB-PRE is the bilinear map, also known as a bilinear pairing. A bilinear map is a mathematical function that maps a pair of group elements to an element in a target group. Bilinear maps are widely used in pairing-based cryptography because they support security constructions based on the hardness of discrete logarithm-related problems. However, pairing operations are computationally expensive and may affect practical implementation efficiency [9]. Let G_1, G_2, and G_T be cyclic groups of prime

order q. A bilinear map is defined as

$$e : G_1 \times G_2 \to G_T.$$

It satisfies the following properties:

- **Bilinearity:** for all $u \in G_1$, $v \in G_2$, and $a, b \in Z_q^*$,

$$e(u^a, v^b) = e(u, v)^{ab}.$$

- **Non-degeneracy:** if g_1 and g_2 are generators of G_1 and G_2, respectively, then

$$e(g_1, g_2) \neq 1.$$

Type-1 pairings use the same source group for both inputs, while Type-3 pairings use two distinct source groups G_1 and G_2. Since there is no efficiently computable isomorphism between G_1 and G_2 in Type-3 pairings, Type-3 constructions require more careful notation and group assignment.

3.2 Decisional Bilinear Diffie-Hellman (DBDH)

DBDH is a common cryptographic hardness assumption based on bilinear maps. It describes the difficulty of distinguishing a real bilinear Diffie-Hellman value from a random element in the target group.

In the Type-3 pairing setting, let G_1, G_2, and G_T be cyclic groups of prime order q, and let $e : G_1 \times G_2 \to G_T$ be a bilinear map. Let $g_1 \in G_1$ and $g_2 \in G_2$ be generators. Given g_1, g_1^a, $g_1^b \in G_1$ and g_2, $g_2^c \in G_2$, where $a, b, c \leftarrow Z_q^*$, the DBDH problem is to distinguish the real value $e(g_1, g_2)^{abc}$ from a random element $T \in G_T$.

The DBDH assumption holds if no probabilistic polynomial-time adversary can distinguish $e(g_1, g_2)^{abc}$ from a random value $T \in G_T$ with non-negligible advantage. Formally, for any probabilistic polynomial-time adversary $\mathcal{A}$, the advantage is defined as

$$Adv_{\mathcal{A}}^{DBDH} = \left| Pr[\mathcal{A}(g_1, g_1^a, g_1^b, g_2, g_2^c, e(g_1, g_2)^{abc}) = 1] - Pr[\mathcal{A}(g_1, g_1^a, g_1^b, g_2, g_2^c, T) = 1] \right|.$$

The DBDH assumption states that $Adv_{\mathcal{A}}^{DBDH}$ is negligible in the security parameter k.

In Type-1 pairings, the DBDH assumption is usually written over a symmetric pairing map $e : G \times G \to G_T$. In this paper, the Type-3 form is used because the proposed construction is built over asymmetric groups G_1 and G_2.

3.3 Identity-Based Proxy Re-encryption

Identity-Based Proxy Re-Encryption (IB-PRE) supports non-interactive ciphertext transformation between users identified by their identities. In an IB-PRE scheme, a proxy can transform a ciphertext encrypted under one identity into a ciphertext decryptable under another identity, without learning the plaintext. A general non-interactive IB-PRE scheme consists of the following algorithms [11]:

- **Setup** (1^k): On input a security parameter 1^k, the system generates the public parameters $params$ and a master secret key msk. The public parameters are distributed to all users, while the master secret key is kept secret by the Private Key Generator (PKG).

- **KeyGen** $(params, msk, id)$: On input the public parameters, master secret key, and an identity id, the PKG generates a private key sk_{id} associated with that identity.

- **Encrypt** $(params, id, m)$: On input the public parameters, an identity id, and a plaintext message m, the algorithm outputs a ciphertext ct_{id} encrypted under identity id.
- **RKGen** $(params, sk_{id_1}, id_1, id_2)$: On input the public parameters, the delegator's private key sk_{id_1}, the delegator identity id_1, and the delegatee identity id_2, the algorithm outputs a re-encryption key $rk_{id_1 \to id_2}$.
- **ReEncrypt** $(params, rk_{id_1 \to id_2}, ct_{id_1})$: On input the public parameters, a re-encryption key, and a ciphertext encrypted under id_1, the proxy outputs a re-encrypted ciphertext ct_{id_2} decryptable under id_2.
- **Decrypt** $(params, sk_{id}, ct_{id})$: On input the public parameters, a private key sk_{id}, and a ciphertext ct_{id}, the algorithm outputs the plaintext m if decryption succeeds, or $\perp$ if decryption fails.

3.4 Security of Non-Interactive Identity-Based Proxy Re-Encryption

The security of non-interactive IB-PRE is commonly defined using indistinguishability under chosen-plaintext attack or chosen-ciphertext attack. In this paper, we consider the IND-prID-CCA security model, where the adversary may access private key extraction, re-encryption key extraction, decryption, and re-encryption oracles under standard restrictions. The goal is to ensure that the adversary cannot distinguish the challenge ciphertext or its valid re-encrypted forms with non-negligible advantage.

The security game is played between a challenger and a probabilistic polynomial-time adversary $\mathcal{A}$ as follows.

Setup. The challenger runs $Setup(1^k)$ to generate the public parameters $params$ and the master secret key msk. The public parameters are given to $\mathcal{A}$, while msk is kept secret.

Query Phase 1. The adversary may adaptively issue the following oracle queries.

- $Extract(id)$: The challenger returns the private key $sk_{id} = KeyGen(params, msk, id)$.
- $RKExtract(id_1, id_2)$: The challenger returns the re-encryption key $rk_{id_1 \to id_2} = RKGen(params, sk_{id_1}, id_1, id_2)$.
- $Decrypt(id, ct)$: The challenger returns the result of $Decrypt(params, sk_{id}, ct)$.
- $ReEncrypt(id_1, id_2, ct)$: The challenger returns the result of $ReEncrypt(params, rk_{id_1 \to id_2}, ct)$.

Challenge. The adversary submits a challenge identity id^* and two equal-length messages m_0 and m_1. The challenger selects a random bit $\beta \leftarrow \{0, 1\}$ and returns the challenge ciphertext $ct^* = Encrypt(params, id^*, m_\beta)$.

Query Phase 2. The adversary may continue to issue oracle queries, but it cannot make any query that trivially reveals the challenge plaintext. In particular, the adversary cannot request $Extract(id^*)$, and it cannot request decryption or re-encryption queries that directly decrypt ct^* or its valid re-encrypted forms.

Guess. The adversary outputs a guess β'. The adversary wins the game if $\beta' = \beta$. The advantage of $\mathcal{A}$ is defined as

$$Adv_{\mathcal{A}}^{IND\text{-}prID\text{-}CCA} = \left| Pr[\beta' = \beta] - \frac{1}{2} \right|.$$

An IB-PRE scheme is IND-prID-CCA secure if every probabilistic polynomial-time adversary has only negligible advantage in the above game. If decryption and re-encryption oracle access are removed, the weaker IND-prID-CPA security model is obtained.

3.5 An IB-PRE Scheme in Type-1 Pairings

We recall Green and Ateniese's IB-PRE scheme [11], in Type-1 pairings as below:

- Setup (1^k): With the input of a security parameter 1^k, select prime order q, set bilinear group $e : G \times G \to G_T$, where G and G_T are groups of prime order q, select g from generator of G, select a master secret s from $\mathbb{Z}_q^*$, set five hash functions, $H_1 : \{0,1\}^* \to G,\ H_2 : \{0,1\}^* \to G,\ H_3 : \{0,1\}^* \to G,\ H_4 : G_T \times \{0,1\}^n \to \mathbb{Z}_q^*$ and $H_5 : G_T \to \{0,1\}^n$. The output is public parameters, $params = (H_1, H_2, H_3, H_4, H_5, g, g^s)$ and master secret key $msk = s$.
- KeyGen $(params,\ s,\ id)$: With the input of $params,\ s$ and $id \in \{0,\ 1\}^*$, compute $sk_{id} = H_1(id)^s$ and set secret key of id, sk_{id}.
- Encrypt $(params,\ id,\ m)$: With the input of $params$, id, message $m \in \{0,1\}^n$, set $\sigma \in G_T$ and set $r = H_4(\sigma, m)$ and compute $A = g^r$, $B = \sigma \cdot e(g^s,\ H_1(id)^r)$, $C = m \oplus H_5(\sigma)$ and $S = H_3(id \parallel (A, B, C))^r$. The output is ciphertext $ct_{id_1} = (S, A, B, C)$.
- RKGen $(params, sk_{id_1},\ id_1,\ id_2)$: With the input of $params$, sk_{id_1}, id_1 and recipient's identity id_2, select $N \in \{0,1\}^{n(k)}$, set $K = e(sk_{id_1}, H_1(id_2))$, and compute $rk_{id_1 \to id_2} = (N, H_2(K \parallel id_1 \parallel id_2 \parallel N) \cdot sk_{id_1})$. The output is re-encryption key $rk_{id_1 \to id_2}$.
- ReEncrypt $(params,\ rk_{id_1 \to id_2}, ct_{id_1})$: With the input of $params$, $rk_{id_1 \to id_2}$ and ct_{id_1}, parse ct_{id_1} as (S, A, B, C) and $rk_{id_1 \to id_2}$ as (N, R), compute $h = H_3\,(id_1 \parallel (A, B, C))$, verify $e(g, S) = e(h, A)$, if invalid return $\perp$, select $t \in \mathbb{Z}_q^*$, compute $B^{'} = B / \frac{e(A, R \cdot h^t)}{e(g^t, S)}$ and $ct_{id_2} = (A, B^{'}, C, id_1, N)$. The output is the re-encrypted ciphertext ct_{id_2}.
- First Level Decrypt $(params,\ sk_{id}, ct_{id_1})$: Parse ct_{id_1} as (S, A, B, C), set $h = H_3\,(id_1 \parallel (A, B, C))$, compute $\sigma^{'} = B / \frac{e(A, sk_{id_1} \cdot h^t)}{e(g^t, S)}$, $m^{'} = C \oplus H_5\left(\sigma^{'}\right)$ and $r^{'} = H_4(\sigma^{'}, m^{'})$, verify $S = h^{r^{'}}, A = g^{r^{'}}$. If invalid, return $\perp$. The output is message $m^{'}$, or $\perp$ if decryption fails.
- Second Level Decrypt (params, sk_{id}, ct_{id_2}): Parse ct_{id_2} as (A, B, C, id_{src}, N), compute $K = e\,(H_1\,(id_{src})\,, sk_{id})\,,\ R = H_2\,(K \parallel id_{src} \parallel id_2 \parallel N)\,,\ \sigma^{'} = B^{'} \cdot e(A, R)$, $m^{'} = C \oplus H_5(\sigma^{'})$ and $r^{'} = H_4(\sigma^{'}, m^{'})$, verify $A = g^{r^{'}}$. If valid, output $m^{'}$, otherwise return $\perp$. The output is message $m^{'}$, or $\perp$ if decryption fails.

Theorem 1. *[11]. The IB-PRE scheme above is IND-prID-CCA secure under the Decisional Bilinear Diffie-Hellman assumption in the random oracle model.*

4 Our Enhanced Type-3 IB-PRE

4.1 System Model, Assumptions and Threat Model

The proposed scheme is designed for identity-based proxy re-encryption in a semi-trusted cloud sharing environment. The system consists of four main entities: the Private Key Generator (PKG), the delegator, the delegatee, and the proxy server.

The PKG is a trusted authority that generates the system public parameters, the master secret key, and the private key for each user based on the user's identity. The delegator is the original data owner who encrypts a message under his or her identity and generates a re-encryption key for sharing. The delegatee is the authorised recipient who receives access to the re-encrypted ciphertext. The proxy server performs the re-encryption operation

by converting a ciphertext encrypted under the delegator's identity into a ciphertext decryptable under the delegatee's identity.

The following assumptions are made in the proposed scheme. First, the PKG is fully trusted and is responsible for protecting the master secret key. Second, users are responsible for keeping their private keys secret. Third, the proxy server is semi-trusted. It follows the re-encryption protocol, but it may try to learn information from ciphertexts or re-encryption keys. Fourth, the adversary is assumed to be unable to solve the Decisional Bilinear Diffie-Hellman (DBDH) problem and the discrete logarithm problem in the selected bilinear groups.

The adversary may obtain the public parameters, observe ciphertexts, access re-encrypted ciphertexts, and obtain some re-encryption keys under the restrictions of the IB-PRE security model. The adversary may also request private keys for non-challenge identities. However, the adversary is not allowed to obtain the private key of the challenge identity or any re-encryption key that directly allows decryption of the challenge ciphertext.

The security definition follows the IND-prID-CCA model for non-interactive IB-PRE described in Section 3.4. Under this model, the adversary may adaptively query private keys, re-encryption keys, decryption oracles, and re-encryption oracles, except for queries that trivially reveal the challenge plaintext. The main security goal is ciphertext confidentiality. A ciphertext encrypted under identity id_1 should only be decrypted by the corresponding private key sk_{id_1}. After re-encryption, the second-level ciphertext should only be decrypted by the authorised delegatee using sk_{id_2}. The proxy should be able to transform the ciphertext, but it should not learn the plaintext.

4.2 Design Rationale

The original Green and Ateniese IB-PRE scheme is constructed using Type-1 pairings, where both pairing inputs belong to the same group. This symmetric setting allows the same identity-related hash-to-point value to be used on either side of the pairing operation. However, this property does not hold in Type-3 pairings because Type-3 pairings use two different source groups, G_1 and G_2, and there is no efficiently computable isomorphism between them.

Due to this difference, the original Type-1 construction cannot be directly converted into Type-3 pairings. If two independent hash-to-point functions are used for G_1 and G_2, the resulting group elements may not preserve the same algebraic relationship required for decryption correctness. This may cause the first-level and second-level decryption algorithms to fail.

To address this problem, the proposed scheme applies the commit-to-point technique. Instead of directly hashing an identity into a group element, the identity is first mapped into a scalar value. Then, the scalar value is combined with a trapdoor value b to generate related identity-based elements in both G_1 and G_2. In this scheme, the identity-related values are represented as $g_1^{H_1(id)} \cdot g_1^b \in G_1$ and $g_2^{H_1(id)} \cdot g_2^b \in G_2$. This preserves the algebraic consistency required by the Type-3 pairing map $e : G_1 \times G_2 \to G_T$.

Therefore, the main design goal of the proposed scheme is to transform the Green and Ateniese IB-PRE construction from Type-1 pairings to Type-3 pairings while preserving the correctness of first-level and second-level decryption.

4.3 Algorithm Overview

The proposed Type-3 IB-PRE scheme consists of seven algorithms: Setup, KeyGen, Encrypt, RKGen, ReEncrypt, First-Level Decrypt, and Second-Level Decrypt.

The Setup and KeyGen algorithms are performed by the PKG. The Encrypt algorithm generates a first-level ciphertext under the delegator's identity. The RKGen algorithm produces a re-encryption key from the delegator to the delegatee. The ReEncrypt algorithm

is performed by the proxy server to convert the first-level ciphertext into a second-level ciphertext. The First-Level Decrypt algorithm allows the original delegator to decrypt the original ciphertext, while the Second-Level Decrypt algorithm allows the delegatee to decrypt the re-encrypted ciphertext. The detailed construction of these algorithms is presented in the following subsection.

4.4 Proposed Type-3 IB-PRE Construction

In the proposed Type-3 construction, let G_1, G_2, and G_T be cyclic groups of prime order q, and let $e : G_1 \times G_2 \to G_T$ be a Type-3 bilinear map. Let $g_1 \in G_1$ and $g_2 \in G_2$ be generators. The master secret is denoted by $s \in Z_q^*$, and the commit-to-point trapdoor value is denoted by $b \in Z_q^*$. The hash functions H_1, H_2, and H_3 map input strings to Z_q^*, while H_4 and H_5 are used for ciphertext randomness and message masking. In the delegation setting, id_1 denotes the delegator or original data owner, and id_2 denotes the delegatee or authorised recipient. The proposed IB-PRE construction under the Type-3 pairing setting is as follows:

- **Setup** (1^k): On input a security parameter 1^k, select a prime order q and set a bilinear group $e : G_1 \times G_2 \to G_T$, where G_1, G_2, and G_T are groups of prime order q. Select g_1 as a generator of G_1 and g_2 as a generator of G_2. Select a master secret $s \leftarrow Z_q^*$ and a trapdoor value $b \leftarrow Z_q^*$. Compute $g_1^s \in G_1$, $g_1^b \in G_1$, and $g_2^b \in G_2$. Set five hash functions $H_1, H_2, H_3 : \{0,1\}^* \to Z_q^*$, $H_4 : G_T \times \{0,1\}^n \to Z_q^*$, and $H_5 : G_T \to \{0,1\}^n$. The output is $params = (G_1, G_2, G_T, e, g_1, g_2, g_1^s, g_1^b, g_2^b, H_1, H_2, H_3, H_4, H_5)$ and $msk = s$.
- **KeyGen** $(params, s, id)$: On input public parameters $params$, master secret key s, and identity $id \in \{0,1\}^*$, compute $sk_{id} = (g_2^{H_1(id)} \cdot g_2^b)^s$. The output is the secret key sk_{id}. In particular, the PKG generates sk_{id_1} for the delegator and sk_{id_2} for the delegatee using the same algorithm.
- **Encrypt** $(params, id_1, m)$: On input public parameters $params$, delegator identity id_1, and message $m \in \{0,1\}^n$, select a random value $\sigma \leftarrow G_T$ and compute $r = H_4(\sigma, m)$, $A = g_1^r$, $B = \sigma \cdot e(g_1^s, (g_2^{H_1(id_1)} \cdot g_2^b)^r)$, $C = m \oplus H_5(\sigma)$, $h = g_2^{H_3(id_1 \| A \| B \| C)}$, and $S = h^r$. The output is the first-level ciphertext $ct_{id_1}^{(1)} = (S, A, B, C)$.
- **RKGen** $(params, sk_{id_1}, id_1, id_2)$: On input public parameters $params$, delegator secret key sk_{id_1}, delegator identity id_1, and delegatee identity id_2, select a random nonce $N \leftarrow \{0,1\}^{n(k)}$. Compute $K = e(g_1^{H_1(id_2)} \cdot g_1^b, sk_{id_1})$ and $R = (g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b) \cdot sk_{id_1}$. The output is the re-encryption key $rk_{id_1 \to id_2} = (N, R)$.
- **ReEncrypt** $(params, rk_{id_1 \to id_2}, ct_{id_1}^{(1)}, id_1)$: On input public parameters $params$, re-encryption key $rk_{id_1 \to id_2} = (N, R)$, first-level ciphertext $ct_{id_1}^{(1)} = (S, A, B, C)$, and delegator identity id_1, compute $h = g_2^{H_3(id_1 \| A \| B \| C)}$ and verify $e(g_1, S) = e(A, h)$. If the verification fails, return $\perp$. Otherwise, compute $B' = B \cdot e(g_1, S)/e(A, R \cdot h)$. The output is the second-level ciphertext $ct_{id_2}^{(2)} = (A, B', C, id_1, N)$.
- **First-Level Decrypt** $(params, sk_{id_1}, id_1, ct_{id_1}^{(1)})$: Parse $ct_{id_1}^{(1)} = (S, A, B, C)$, compute $h = g_2^{H_3(id_1 \| A \| B \| C)}$, and verify $e(g_1, S) = e(A, h)$. If the verification fails, return $\perp$. Otherwise, compute $\sigma' = B/e(A, sk_{id_1})$, $m' = C \oplus H_5(\sigma')$, and $r' = H_4(\sigma', m')$. If $A = g_1^{r'}$ and $S = h^{r'}$, output m'. Otherwise, return $\perp$.
- **Second-Level Decrypt** $(params, sk_{id_2}, id_2, ct_{id_2}^{(2)})$: Parse $ct_{id_2}^{(2)} = (A, B', C, id_1, N)$. Compute $K = e(g_1^{H_1(id_1)} \cdot g_1^b, sk_{id_2})$ and $\sigma' = B' \cdot e(A, g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b)$. Then compute $m' = C \oplus H_5(\sigma')$ and $r' = H_4(\sigma', m')$. If $A = g_1^{r'}$, output m'. Otherwise, return $\perp$.

4.5 Correctness Analysis

This subsection verifies that the proposed Type-3 IB-PRE construction correctly recovers the plaintext in both decryption modes. The first-level correctness shows that a ciphertext encrypted under identity id_1 can be decrypted using sk_{id_1}. The second-level correctness shows that, after re-encryption, the ciphertext can be decrypted using the delegatee's secret key sk_{id_2}. The correctness follows from the bilinearity of the Type-3 pairing and the algebraic consistency preserved by the commit-to-point construction.

Correctness of first-level decryption. Recall that the first-level ciphertext is $ct_{id_1}^{(1)} = (S, A, B, C)$, where $A = g_1^r$, $B = \sigma \cdot e(g_1^s, (g_2^{H_1(id_1)} \cdot g_2^b)^r)$, $C = m \oplus H_5(\sigma)$, $h = g_2^{H_3(id_1 \| A \| B \| C)}$, and $S = h^r$. The secret key of the delegator identity id_1 is $sk_{id_1} = (g_2^{H_1(id_1)} \cdot g_2^b)^s$.

During first-level decryption, the decryptor first verifies the ciphertext validity by checking $e(g_1, S) = e(A, h)$. This verification is correct because $e(g_1, S) = e(g_1, h^r) = e(g_1, h)^r$ and $e(A, h) = e(g_1^r, h) = e(g_1, h)^r$. Therefore, $e(g_1, S) = e(A, h)$ holds for a valid first-level ciphertext.

After the verification succeeds, the decryptor computes $\sigma' = B/e(A, sk_{id_1})$. By substituting $A = g_1^r$ and $sk_{id_1} = (g_2^{H_1(id_1)} \cdot g_2^b)^s$, we have

$$e(A, sk_{id_1}) = e\left(g_1^r, \left(g_2^{H_1(id_1)} \cdot g_2^b\right)^s\right) = e\left(g_1^s, \left(g_2^{H_1(id_1)} \cdot g_2^b\right)^r\right).$$

Since $B = \sigma \cdot e(g_1^s, (g_2^{H_1(id_1)} \cdot g_2^b)^r)$, it follows that

$$\sigma' = \frac{\sigma \cdot e\left(g_1^s, \left(g_2^{H_1(id_1)} \cdot g_2^b\right)^r\right)}{e\left(g_1^s, \left(g_2^{H_1(id_1)} \cdot g_2^b\right)^r\right)} = \sigma.$$

The plaintext is recovered as $m' = C \oplus H_5(\sigma') = (m \oplus H_5(\sigma)) \oplus H_5(\sigma) = m$. Finally, since $\sigma' = \sigma$, $m' = m$, and $r = H_4(\sigma, m)$, we have $r' = H_4(\sigma', m') = r$. Therefore, $A = g_1^r = g_1^{r'}$ and $S = h^r = h^{r'}$. Thus, the first-level decryption algorithm correctly recovers the plaintext m.

Correctness of second-level decryption. Recall that the second-level ciphertext is $ct_{id_2}^{(2)} = (A, B', C, id_1, N)$. From the original first-level encryption, $A = g_1^r$, $B = \sigma \cdot e(g_1^s, (g_2^{H_1(id_1)} \cdot g_2^b)^r)$, $C = m \oplus H_5(\sigma)$, $h = g_2^{H_3(id_1 \| A \| B \| C)}$, and $S = h^r$.

In the re-encryption key generation algorithm, the value K is computed as

$$K = e\left(g_1^{H_1(id_2)} \cdot g_1^b, sk_{id_1}\right) = e\left(g_1^{H_1(id_2)} \cdot g_1^b, \left(g_2^{H_1(id_1)} \cdot g_2^b\right)^s\right),$$

and the re-encryption component is $R = (g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b) \cdot sk_{id_1}$.

During re-encryption, the proxy computes $B' = B \cdot e(g_1, S)/e(A, R \cdot h)$. Since the first-level ciphertext verification gives $e(g_1, S) = e(A, h)$, we obtain

$$B' = B \cdot \frac{e(A, h)}{e(A, R \cdot h)} = B \cdot \frac{e(A, h)}{e(A, R) \cdot e(A, h)} = \frac{B}{e(A, R)}.$$

By substituting $R = (g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b) \cdot sk_{id_1}$, we have

$$e(A, R) = e\left(A, g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b\right) \cdot e(A, sk_{id_1}).$$

Thus,

$$B' = \frac{B}{e\left(A, g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b\right) \cdot e(A, sk_{id_1})}.$$

From the correctness of first-level decryption, $B/e(A, sk_{id_1}) = \sigma$. Hence,

$$B' = \frac{\sigma}{e\left(A, g_2^{H_2(K\|id_1\|id_2\|N)} \cdot g_2^b\right)}.$$

During second-level decryption, the delegatee computes

$$K = e\left(g_1^{H_1(id_1)} \cdot g_1^b, sk_{id_2}\right) = e\left(g_1^{H_1(id_1)} \cdot g_1^b, \left(g_2^{H_1(id_2)} \cdot g_2^b\right)^s\right).$$

By bilinearity and the commutativity of scalar multiplication in the exponent, this value is equal to the value of K computed during re-encryption key generation:

$$e\left(g_1^{H_1(id_2)} \cdot g_1^b, \left(g_2^{H_1(id_1)} \cdot g_2^b\right)^s\right) = e\left(g_1^{H_1(id_1)} \cdot g_1^b, \left(g_2^{H_1(id_2)} \cdot g_2^b\right)^s\right).$$

Therefore, the same value $H_2(K \parallel id_1 \parallel id_2 \parallel N)$ is used in both re-encryption and second-level decryption. The delegatee then computes

$$\begin{aligned}
\sigma' &= B' \cdot e\left(A, g_2^{H_2(K\|id_1\|id_2\|N)} \cdot g_2^b\right) \\
&= \frac{\sigma}{e\left(A, g_2^{H_2(K\|id_1\|id_2\|N)} \cdot g_2^b\right)} \cdot e\left(A, g_2^{H_2(K\|id_1\|id_2\|N)} \cdot g_2^b\right) \\
&= \sigma.
\end{aligned}$$

The plaintext is recovered as $m' = C \oplus H_5(\sigma') = (m \oplus H_5(\sigma)) \oplus H_5(\sigma) = m$. Finally, since $\sigma' = \sigma$, $m' = m$, and $r = H_4(\sigma, m)$, we have $r' = H_4(\sigma', m') = r$. Therefore, $A = g_1^r = g_1^{r'}$. Thus, the second-level decryption algorithm correctly recovers the plaintext m.

4.6 Security Analysis

This subsection discusses the security of the proposed Type-3 IB-PRE construction. The security argument follows the structure of the original Green and Ateniese IB-PRE scheme, while showing that the Type-3 commit-to-point representation does not reveal additional information to the adversary. The goal is to justify that the conversion from Type-1 to Type-3 pairings preserves ciphertext confidentiality under the IND-prID-CCA model.

Theorem 2. *The enhanced IB-PRE scheme in Type-3 pairings is IND-prID-CCA secure under the Decisional Bilinear Diffie-Hellman assumption in the random oracle model, assuming that the commit-to-point transformation is computationally binding under the discrete logarithm assumption.*

Proof sketch. The proposed construction follows the security structure of the Green and Ateniese IB-PRE scheme [11], while replacing the Type-1 hash-to-point representation with a Type-3 commit-to-point representation. In the proposed scheme, id_1 denotes the delegator or original data owner, and id_2 denotes the delegatee or authorised recipient. The identity-related values for the delegator are represented as $g_1^{H_1(id_1)} \cdot g_1^b \in G_1$ and $g_2^{H_1(id_1)} \cdot g_2^b \in G_2$, while the identity-related values for the delegatee are represented as $g_1^{H_1(id_2)} \cdot g_1^b \in G_1$ and $g_2^{H_1(id_2)} \cdot g_2^b \in G_2$. These paired representations preserve the required algebraic relationship across the two source groups of the Type-3 pairing.

Under the discrete logarithm assumption, the trapdoor value b makes the commit-to-point representation computationally binding and prevents inconsistent identity openings. Therefore, replacing the Type-1 identity hash-to-point values with the Type-3 commit-to-point representation does not provide the adversary with additional information about the plaintext, private keys, or re-encryption keys. A ciphertext encrypted under id_1 remains

protected before re-encryption. After re-encryption, the transformed ciphertext can only be decrypted using the delegatee private key sk_{id_2}.

If an adversary can break the proposed Type-3 IB-PRE scheme with non-negligible advantage, then a simulator can use such an adversary to distinguish the corresponding DBDH instance in the bilinear groups. Since the hash functions H_1, H_2, H_3, H_4, and H_5 are simulated in the random oracle model as in the original IB-PRE proof, the adversary's advantage remains negligible. Therefore, the proposed Type-3 IB-PRE scheme is IND-prID-CCA secure. □

4.7 Parameter Size and Notation Discussion

The parameter sizes of the Type-3 pairing-based IB-PRE scheme using the BLS12-381 curve and the Type-1 pairing-based IB-PRE scheme using the Type-A curve are presented in Table 2. The computational cost depends on the number of bits used to compute the cryptographic components and the size and complexity of the cryptographic elements. In the case of BLS12-381 curve, group elements are based on point compression. A compressed element in G_1 takes $381 + 1 = 382$ bits (approximately 48 bytes) mathematically, values are stored in cryptographic libraries in full-byte alignment. Consequently, it is increased to 384 bits (48 bytes). The representation of a compressed G_2 element contains 96 bytes (768 bits) and that is why Table 2 reports 768 bits rather than the mathematical 764 bits. Although the theoretical size of a G_T element in BLS12-381 is computed as 12 field elements of approximately 381 bits each, it is not stored in practice with the precise mathematical bit length, a G_T element is usually stored in 4,576 bits (572 bytes). G_T elements are the result of bilinear pairings operations; they control the cost of computation and storage in the scheme [11]. In Type-1 pairings, G_1 and G_2 are on the same elliptic curve over the same base field. The point in G_1 comprises two coordinates each about 768 bits providing the uncompressed size of 1536 bits (192 bytes), and a compressed element uses one coordinate plus a sign bit (97 bytes). This target group G_T is considered over the extension field of the embedding degree ($k = 2$), with the size of a target group being 3072 bits (384 bytes).

The public parameters include several complex cryptographic components such as group elements G_1, G_2, G_T, a bilinear pairings function e, group generators g_1, g_2, exponential values g_1^s, g_1^b, g_2^b and multiple hash functions H_1, H_2, H_3, H_4, H_5. These elements are the basis for using the functionality of the scheme includes identity binding, encryption, re-encryption and decryption, but they also increase the computing cost. Although these parameters are calculated during the setup stage, they are repeatedly accessed during encryption and decryption, thereby increasing the overall computing load. The total parameter size is reduced to 5696 bits (712 bytes) by optimizing these components instead of 7782 bits (973 bytes). This decreases 2086 bits (261 bytes) memory overhead and accelerates the common access to group elements in the re-encryption and decryption operation.

Table 2: Type-1 and Type-3 IB-PRE Computation Cost

Scheme	Size (bits)						
	G_1	G_2	G_T	msk	Params	Scalar	Secret Key
IB-PRE Type-3 (BLS12-381, 128-bit)	384	768	4576	256	5696	256	1600
IB-PRE Type-1 (Type-A, 128-bit)	1536	1536	3072	256	7782	256	6400

In the key generation algorithm, the trapdoor value b is combined with the identity

hash value $H_1(id_1)$ to construct the identity-related secret key. For an identity id_1, the secret key is computed as $sk_{id_1} = (g_2^{H_1(id_1)} \cdot g_2^b)^s$. This form ensures that the secret key belongs to the correct Type-3 pairing input group and can be used as the second input of the bilinear map.

In the encryption algorithm, the ciphertext component B is computed as $B = \sigma \cdot e(g_1^s, (g_2^{H_1(id_1)} \cdot g_2^b)^r)$. The verification component is constructed by first computing $h = g_2^{H_3(id_1 \| A \| B \| C)}$ and then computing $S = h^r$. This avoids treating H_3 directly as a group element, because H_3 maps to Z_q^* in the proposed Type-3 construction.

In the re-encryption key generation algorithm, the value K is computed as $K = e(g_1^{H_1(id_2)} \cdot g_1^b, sk_{id_1})$. Then, the re-encryption component R is computed as $R = (g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b) \cdot sk_{id_1}$. This form is consistent with the Type-3 pairing setting because $g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b$ is an element of G_2, not a scalar.

In the re-encryption algorithm, the proxy first verifies the first-level ciphertext by checking $e(g_1, S) = e(A, h)$. If the ciphertext is valid, the proxy computes $B' = B \cdot e(g_1, S)/e(A, R \cdot h)$. This transformation removes the delegator secret-key-related component from the ciphertext and introduces the delegatee-recoverable masking component.

In the first-level decryption algorithm, the original data owner recovers $\sigma' = B/e(A, sk_{id_1})$. Then, the plaintext is recovered by $m' = C \oplus H_5(\sigma')$. This matches the encryption structure because $e(A, sk_{id_1}) = e(g_1^s, (g_2^{H_1(id_1)} \cdot g_2^b)^r)$.

In the second-level decryption algorithm, the delegatee first computes $K = e(g_1^{H_1(id_1)} \cdot g_1^b, sk_{id_2})$. Then, the delegatee recovers $\sigma' = B' \cdot e(A, g_2^{H_2(K \| id_1 \| id_2 \| N)} \cdot g_2^b)$. Finally, the plaintext is obtained as $m' = C \oplus H_5(\sigma')$. Therefore, the notation used in key generation, encryption, re-encryption, and decryption is consistent with the Type-3 pairing map $e : G_1 \times G_2 \to G_T$.

5 Type-3 IB-PRE for File Encryption

The IB-PRE scheme enables encrypted data sharing based on the identity of the recipient, without requiring direct public key exchange between users. Type-3 IB-PRE also supports delegation through a semi-trusted proxy, where the proxy can transform a ciphertext for another recipient without learning the plaintext. These properties make Type-3 IB-PRE suitable for cloud storage and distributed file sharing environments, where both data accessibility and privacy are important [12].

However, Type-3 IB-PRE relies on bilinear pairing operations, which are computationally expensive. Pairing-based encryption is generally less efficient than symmetric encryption for large data. Therefore, directly encrypting large files using Type-3 IB-PRE is not practical for real file sharing applications [8].

To address this limitation, the proposed system adopts a hybrid encryption approach. A random symmetric key is generated to encrypt the file content using AES, while the Type-3 IB-PRE scheme is used to encrypt and delegate the AES key. During sharing, the proxy re-encrypts the encrypted AES key for the intended recipient. The recipient then decrypts the AES key using the corresponding private key and uses it to decrypt the file content. This design combines the efficiency of AES for file encryption with the flexible delegation capability of Type-3 IB-PRE. In the file sharing prototype, AES is used for file-content encryption, while Type-3 IB-PRE is used only to protect and delegate the AES key. The AES key size is 256 bits [17]. The encrypted file content and the encrypted AES key are stored separately, so the pairing-based operation is applied only to the small symmetric key rather than the whole file. Figure 1 shows the hybrid encryption workflow.

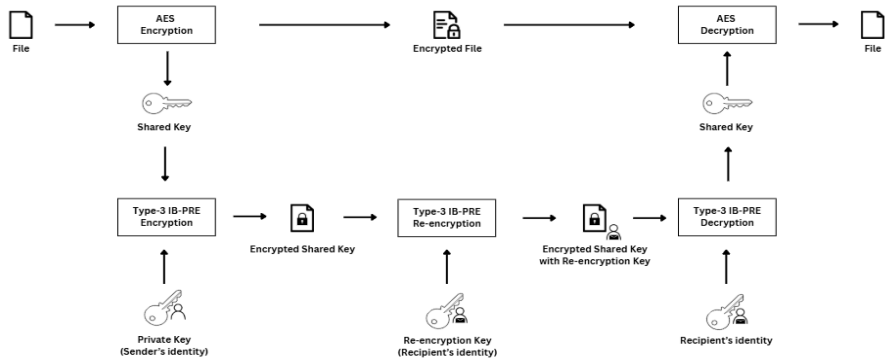


Figure 1: Hybrid Cryptography System: AES + Type-3 IB-PRE

5.1 Hash Function

Hash functions are used in the proposed system to support deterministic mapping, message masking, and file integrity verification. In the Type-3 IB-PRE construction, hash functions are used to map identities and ciphertext-related values into scalar values, and to derive masking values for ciphertext components. In the file sharing prototype, SHA-256 is used for file integrity verification because it provides a 256-bit digest and is widely supported in modern security applications [10].

To verify file integrity, the system computes and stores the SHA-256 hash value of the original file before uploading the file. When the file is retrieved, the hash value is recomputed and compared with the stored value. If both values match, the file is considered unchanged during storage or transmission.

5.2 System Architecture

Figure 2 shows the workflow of the proposed file sharing system. It is divided into three layers: system frontend, system backend, and database. At the frontend of the system, users must register and log in before they can perform actions like uploading files, sharing files, viewing files and downloading files. These actions start the process of backend management. The backend of the system handles operations such as storing user details, encrypting files with AES and Type-3 IB-PRE, re-encrypting files for sharing with new recipients, and decrypting files. The database serves as a secure repository for user information, file metadata, encryption keys, and re-encryption keys. It also helps to upload encrypted files to the cloud and retrieve file metadata and keys for decryption and sharing.

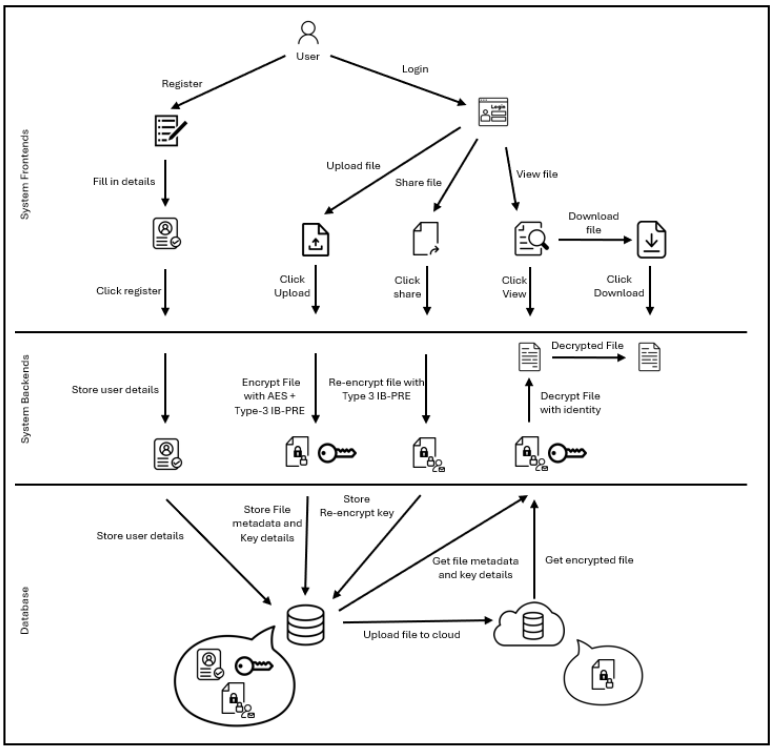


Figure 2: System Workflow of Proposed File Sharing System

5.3 User Interface and Functionality

The proposed file sharing system supports file management, file upload, file sharing, file viewing, file downloading, and file deletion. In the file management view, users can upload files through the upload function. When a file is uploaded, the system encrypts the file content using an AES key and stores the encrypted file on Google Drive. The AES key is then encrypted using Type-3 IB-PRE based on the user's email identity. After the upload is completed, the file record appears on the file management page, where users can share, view, or delete the file.

For file sharing, the user selects a recipient by searching for the recipient's username or email. Once the recipient is selected, the system performs re-encryption on the encrypted AES key using the Type-3 IB-PRE re-encryption algorithm. For file viewing, the system first decrypts the AES key using Type-3 IB-PRE. If the file belongs to the original owner, first-level decryption is used. If the file is a shared file, second-level decryption is used. The recovered AES key is then used to decrypt the encrypted file content for viewing or downloading.

5.4 Prototype Functionality Testing

Table 3 summarizes the prototype-level functionality test cases of the file sharing system.

Table 3: Prototype Functionality Test Cases

No.	Test Case	Expected Result	Result
1	Upload file under 100MB	File is encrypted using AES, and the AES key is protected by Type-3 IB-PRE.	Passed
2	Upload file above 100MB	The system rejects the file and displays the upload limit message.	Passed
3	Share file with selected user	The encrypted AES key is re-encrypted for the selected recipient.	Passed
4	Share file without recipient	The system requests the user to select a recipient.	Passed
5	View original file	The AES key is recovered using first-level IB-PRE decryption.	Passed
6	View shared file	The AES key is recovered using second-level IB-PRE decryption.	Passed
7	Download decrypted file	The file content is correctly recovered and downloaded.	Passed

5.5 Type-3 IB-PRE Library Performance Testing

To improve experimental reproducibility, the hardware, software, library dependencies, and test conditions are summarized in Table 4. The benchmark focuses on the execution time of the main Type-3 IB-PRE algorithms in the proposed Rust-based library.

Table 4: Experimental Setup and Test Conditions

Item	Description
Operating System	Windows 11 Pro, 64-bit
Processor	Intel(R) Core(TM) i5-10400 CPU @ 2.90GHz
Memory	16GB RAM
Implementation Language	Rust 2021 edition
Compiler and Build Tool	`rustc 1.89.0`, `cargo 1.89.0`
Main Libraries	`irmaseal-curve` 0.1.0, `pairing` 0.23.0, `sha2` 0.10.6, `rand_core` 0.6
Build Profile	Release profile with `opt-level = "s"` and `lto = true`
Security Level	Approximately 128-bit security using Type-3 pairings
Tested Algorithms	Setup, KeyGen, Encrypt, RKGen, ReEncrypt, First-Level Decrypt, Second-Level Decrypt
Message Sizes	32, 128, 512, 1024, and 2048 bytes
Application Test Condition	File sharing prototype tested with upload cases up to 100MB
Evaluation Metric	Execution time measured in milliseconds

Additional dependencies used for serialization, encoding, and web assembly support include `base64`, `serde`, `serde_json`, `wasm-bindgen`, `js-sys`, `web-sys`, and `getrandom`. These dependencies support data encoding, JSON processing, and web-based integration,

while the main cryptographic operations are implemented using the Rust pairing-related libraries listed in Table 4. The Rust implementation follows the algebraic structure of the proposed scheme but uses the additive notation and type representations provided by the Irmaseal curve library [1]. The scheme is described in standard multiplicative notation for clarity.

The benchmark therefore evaluates the implemented group and pairing operations provided by the library, while the mathematical construction is presented using the conventional Type-3 notation. The benchmark uses synthetic plaintext inputs rather than an external dataset. The tested plaintexts are byte arrays with sizes of 32, 128, 512, 1024, and 2048 bytes. The identities used in the benchmark are fixed test identities, such as `alice@example.com` and `bob@example.com`. Random values for the native Rust benchmark are generated using the operating-system random number generator through `OsRng`. The nonce length used in re-encryption key generation is 256 bits. SHA-256 is used in the hash-based components of the implementation.

The IB-PRE scheme using Type-3 pairings is an encryption mechanism that enables the proxy to convert ciphertext encrypted for one identity to another identity without disclosing the plaintext. The performance of the IB-PRE scheme using Type-3 pairings was analysed to evaluate the computational cost and execution time of various encryption operations. This analysis was conducted in a Rust programming environment, using a system equipped with an Intel Core i5-10400 processor and 16GB of RAM. The key operations analysed include key generation, encryption, re-encryption, and decryption. In this paper, the irmaseal curve library serves as the core encryption foundation. It is a custom pairings-friendly elliptic curve designed for the project, which focuses on privacy-protecting identities and encryption systems. This curve is optimized for IB-PRE and utilizes efficient Type-3 pairings to ensure strong security and high performance.

Table 6, Table 7, Table 8, and Figure 3 summarize the execution time of the proposed Type-3 IB-PRE algorithms. The measured operations include setup, private key generation, encryption, re-encryption key generation, re-encryption, first-level decryption, and second-level decryption. The encryption benchmark was also carried out using message sizes of 32, 128, 512, 1024, and 2048 bytes to evaluate performance stability across different plaintext sizes. The results show that the execution time remains stable across the tested message sizes. This is because the Type-3 IB-PRE component mainly encapsulates the encryption key rather than encrypting the entire file content directly. Therefore, the communication overhead remains constant at 1944 bytes in the tested message range. The **Encrypt** algorithm is the most expensive operation, taking 12.00 ms, because it requires pairing-related operations and scalar multiplications. In contrast, the one-time **KeyGen** algorithm takes 2.63 ms. The first-level and second-level decryption algorithms take 6.18 ms and 5.11 ms, respectively. The relatively low **Setup** time of 2.25 ms and **RKGen** time of 3.20 ms further show that the proposed library is practical for prototype-level secure file sharing. Existing PRE and IB-PRE schemes are evaluated under different assumptions, implementation environments, and application settings. Therefore, a direct execution-time comparison is not always fair. Accordingly, Table 5 provides a feature-level and evaluation-level comparison between the proposed scheme and representative existing schemes. The proposed scheme is then evaluated through direct Rust-based benchmarking.

Table 5: Feature and Evaluation Comparison with Existing Schemes

Scheme	Pairing Type	Identity-Based	Re-encryption	Evaluation Focus
Green and Ateniese [11]	Type-1	Yes	Yes	Theoretical IB-PRE construction and security proof
He et al. [12]	Pairing-based	Yes	Yes	Blockchain-assisted decentralized storage system
Zengin et al. [20]	Type-3	No	Yes	CCA-secure unidirectional PRE using asymmetric pairings
Li et al. [14]	Pairing-free	Yes	Yes	Broadcast PRE for cloud environments without pairings
Proposed Scheme	Type-3	Yes	Yes	Rust-based IB-PRE library and file sharing prototype benchmark

Table 6: Type-3 IB-PRE First-Level Execution Time

Scheme	Time Taken (ms)					
	Setup	$KeyGen_{Alice}$	$KeyGen_{Bob}$	*Average*	*Encrypt*	*First-Level Decrypt*
TYPE-3 IB-PRE	2.25	2.63	2.63	2.63	12.00	6.18

Table 7: Type-3 IB-PRE Second-Level Execution Time

Scheme	Time Taken (ms)		
	$RKGen_{Alice-Bob}$	*Re-Encrypt*	*Second-Level Decrypt*
TYPE-3 IB-PRE	3.20	7.19	5.11

Table 8: Performance Analysis Across Different Message Sizes

Message Size	Time Taken (ms)			
(Bytes)	*Encrypt*	*Re-Encrypt*	*First-Level Decrypt*	*Second-Level Decrypt*
32	11.85	7.10	6.40	5.18
128	11.95	7.17	6.01	5.00
512	11.84	7.22	6.09	5.15
1024	11.87	7.08	6.20	5.08
2048	12.48	7.37	6.18	5.12

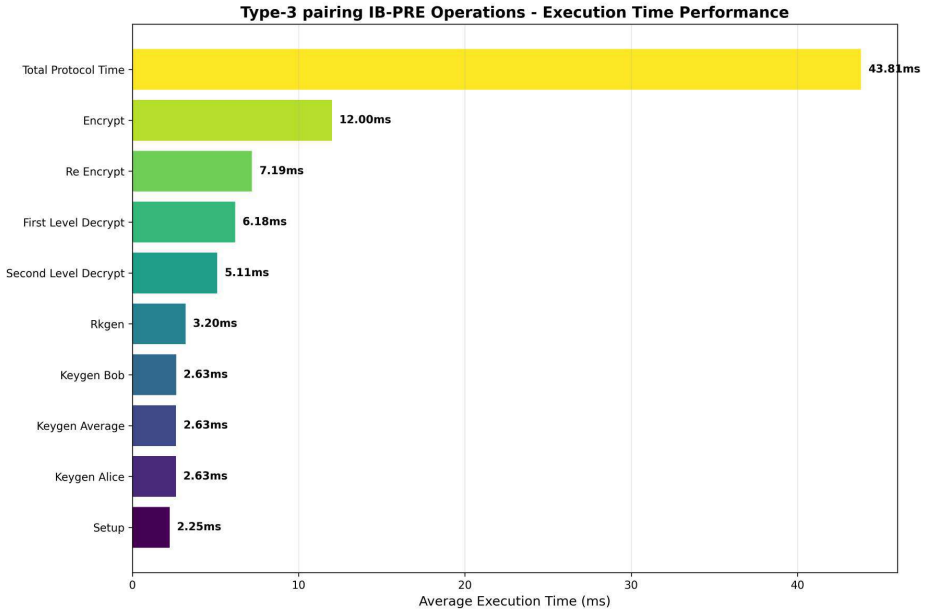


Figure 3: Type-3 IB-PRE Execution Time

6 Conclusion

This paper proposes a secure file sharing system by integrating a Type-3 IB-PRE cryptographic library with AES-based hybrid encryption. The implementation addresses security issues related to cloud storage and data sharing in environments involving semi-trusted parties. The use of Type-3 pairings improves the practicality of the Green and Ateniese IB-PRE construction for modern pairing-based implementation. The proposed system enables file owners to securely share encrypted files through key delegation without exposing private keys or requiring direct interaction with the data owner. The experimental results show that the proposed library performs key generation, encryption, re-encryption, and decryption efficiently in a prototype file sharing environment.

This work also has several limitations. First, the proposed scheme still relies on a trusted PKG, which introduces the key escrow problem commonly found in identity-based cryptography. Second, the current performance evaluation is conducted in a prototype environment and does not yet include large-scale multi-user deployment or high-concurrency cloud workloads. Third, the security argument is presented as a proof sketch based on the security structure of the Green and Ateniese scheme and the binding property of the commit-to-point transformation, rather than a complete formal reduction. Finally, although the prototype demonstrates practical functionality, the current evaluation is limited to selected message sizes and application-level file upload tests.

Future work will focus on improving the system in several directions. A distributed or threshold PKG can be investigated to reduce the key escrow problem. More advanced access control functions, such as revocation, time-limited delegation, and audit logging, can be added to improve practical deployment. The implementation can also be evaluated under larger datasets, more users, and real cloud workloads. In addition, further optimization of Type-3 pairing operations and batch re-encryption can be explored to improve performance in large-scale secure file sharing systems.

7 Acknowledgement

This work was supported by the Ministry of Higher Education of Malaysia's Fundamental Research Grant Scheme (FRGS/1/2023/ICT07/MMU/01/1).

References

[1] irmaseal_curve - rust. https://docs.rs/irmaseal-curve/latest/irmaseal_curve/. Accessed: 2026-1-30.

[2] M. Abe, F. Hoshino, and M. Ohkubo. Design in type-i, run in type-iii: Fast and scalable bilinear-type conversion using integer programming. In *Advances in Cryptology–CRYPTO 2016: 36th Annual International Cryptology Conference*, pages 387–415. Springer, 2016.

[3] G. Ateniese, K. Fu, M. Green, and S. Hohenberger. Improved proxy re-encryption schemes with applications to secure distributed storage. *ACM Transactions on Information and System Security*, 9(1):1–30, 2006.

[4] M. Blaze, G. Bleumer, and M. Strauss. Divertible protocols and atomic proxy cryptography. In *Lecture Notes in Computer Science*, pages 127–144, 1998.

[5] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In *Lecture Notes in Computer Science. Advances in Cryptology — CRYPTO 2001*, pages 213–229, 2001.

[6] K.-M. Chan, S.-H. Heng, S.-Y. Tan, and S.-C. Tan. An efficient ciphertext-policy decryptable attribute-based keyword search scheme with dynamic attribute support. *Electronics*, 14:4325, 2025.

[7] Liqing Chen, Meng Zhang, and Jiguo Li. Conditional identity-based broadcast proxy re-encryption with anonymity and revocation. *IEEE Transactions on Reliability*, 2025. Early access.

[8] H. M. Eldeeb, H. M. Dahshan, and A. E. D. R. Shehata. A new cryptographic pairing-based security solution for cloud storage. In *2018 International Japan-Africa Conference on Electronics, Communications and Computations (JAC-ECC)*, pages 73–78, 2018.

[9] S. Ema, Y. Sato, K. Emura, and T. Ohigashi. Implementation and evaluation of an identity-based encryption with security against the kgc. In *2021 Ninth International Symposium on Computing and Networking Workshops (CANDARW)*, 2021.

[10] H. Gilbert and H. Handschuh. Security analysis of sha-256 and sisters. In *Lecture Notes in Computer Science*, pages 175–193. Springer Berlin Heidelberg, 2004.

[11] M. Green and G. Ateniese. Identity-based proxy re-encryption. In *Lecture Notes in Computer Science. Applied Cryptography and Network Security*, pages 288–306, 2007.

[12] Jiayu He, Dong Zheng, Rui Guo, Yushuang Chen, Kemeng Li, and Xiaoling Tao. Efficient identity-based proxy re-encryption scheme in blockchain-assisted decentralized storage system. *International Journal of Network Security*, 23(5), 2021.

[13] Huidan Hu, Huasong Jin, and Changlu Lin. Identity-based broadcast proxy re-encryption with dynamic functionality for flexible data sharing in cloud environments. *Symmetry*, 17(7):1008, 2025.

[14] Binhan Li, Lunzhi Deng, and Yiming Mou. A pairing-free identity-based broadcast proxy re-encryption scheme for the cloud. In *2024 14th International Conference on Information Technology in Medicine and Education (ITME)*, pages 521–528, 2024.

[15] H. Li, L. Deng, Y. Gu, N. Wang, Y. Chen, and S. Li. A secure data sharing scheme based on searchable public key encryption for authorized multi-receiver. *Journal of Systems Architecture*, 167:103489, 2025.

[16] Sumana Maiti, Sudip Misra, and Anandarup Mondal. Cbp: Coalitional-game-based broadcast proxy re-encryption in iot. *IEEE Internet of Things Journal*, 10(17):15642–15651, 2023.

[17] B. Sarkar, A. Saha, D. Dutta, G. De Sarkar, and K. Karmakar. A survey on the advanced encryption standard (aes): A pillar of modern cryptography. *International Journal of Computer Science and Mobile Computing*, 13(4):68–87, 2024.

[18] A. Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology*, pages 47–53, 1985.

[19] J. Shao, G. Wei, Y. Ling, and M. Xie. Identity-based conditional proxy re-encryption. In *2011 IEEE International Conference on Communications (ICC)*, pages 1–5, 2011.

[20] Benjamin Zengin, Paulin Deupmann, Nicolas Buchmann, and Marian Margraf. Chosen-ciphertext secure unidirectional proxy re-encryption based on asymmetric pairings. *Applied Sciences*, 14(23):11322, 2024.

[21] J. Zhang and A. Datta. Blockchain-enabled data governance for privacy-preserved sharing of confidential data. *PeerJ Computer Science*, 10:e2581, 2024.

[22] Jie Zhang, Shuai Su, Hong Zhong, Jie Cui, and Debiao He. Identity-based broadcast proxy re-encryption for flexible data sharing in vanets. *IEEE Transactions on Information Forensics and Security*, 18:4830–4842, 2023.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 38–50.

Evolutionary Optimisation of an AES S-Box

Ahmed Jobaer[1], Nur Hafiza binti Zakaria[1,2], Farida Ridzuan[1,2] and A. H. Azni[1,2]

[1]Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai 71800, Negeri Sembilan, Malaysia 3241269@raudah@usim.edu.my
[2]CyberSecurity and Systems Research Unit, Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai, 71800 Negeri Sembilan, Malaysia
{mzhafiza,farida,ahazni}@usim.edu.my

Abstract. The Advanced Encryption Standard (AES) uses fixed S-Box and MixColumns transformations to provide nonlinear substitution and diffusion. This paper presents a reproducible offline Genetic Algorithm (GA) method for constructing an alternative AES-compatible S-Box and validates a separately obtained GA-derived MixColumns candidate. Each S-Box candidate is encoded by a fixed 72-bit chromosome comprising an 8×8 affine matrix and an 8-bit constant. Ten independent runs were conducted using tournament selection, two-point crossover, bit-flip mutation, elitism, and matrix repair. The selected S-Box achieved fitness 0.955225, average Strict Avalanche Criterion (SAC) 0.501709, vectorial nonlinearity 112, differential uniformity 4, maximum absolute LAT value 32, algebraic degree 7, and no fixed or opposite fixed points. The MixColumns candidate was verified as invertible, MDS, and having branch number 5. Both fixed components were integrated into AES-128. Using 100 paired ciphertext sequences of 1,000,000 bits per implementation, standard AES and GA-AES passed all 15 NIST SP 800-22 test categories and all 188 reportable test instances. The results indicate comparable ciphertext statistical randomness under the stated configuration.

Keywords: Advanced Encryption Standard (AES) · Genetic Algorithm (GA) · S-Box · MixColumns · Strict Avalanche Criterion · NIST SP 800-22

1 Introduction

The Advanced Encryption Standard (AES) is a widely adopted symmetric block cipher with a 128-bit block size and key sizes of 128, 192, or 256 bits [12, 5]. In its round structure, the S-Box provides nonlinear substitution and MixColumns provides linear diffusion. Both transformations are fixed by the standard.

The construction of alternative AES-compatible components remains a useful research direction for studying how evolutionary search can explore large combinatorial spaces while enforcing cryptographic constraints. This direction does not imply that standard AES is weak. Rather, it examines whether alternative transformations can be constructed reproducibly and evaluated using transparent criteria. Prior GA-based cryptographic studies have mainly addressed key or sequence generation, genetic operators, application-level cryptosystems, and S-Box construction [19, 17, 14, 10, 9, 13, 18, 11].

The term "dynamic S-Box" is used inconsistently in the literature and may imply that a new mapping is generated during every encryption session or round [16, 4, 11, 8]. The present study does not perform runtime regeneration. The GA operates only during the offline design stage; the selected S-Box and MixColumns matrix are then fixed throughout encryption and decryption.

Table 1: Prior GA-oriented cryptographic studies

Ref.	Primary focus	Relevance	Scope relative to this study
[19]	Symmetric-key randomness	Demonstrates GA-based search for key material with statistical evaluation.	Focuses on key randomness rather than AES round-component construction.
[17]	Cipher key-sequence generation	Shows evolutionary generation of cryptographic sequences.	Does not address affine S-Box parameters or diffusion matrices.
[14]	Genetic operators in symmetric cryptography	Examines how GA operators and entropy-based evaluation can be incorporated into a cipher.	Proposes a broader cryptosystem rather than an AES-compatible component search.
[10]	ECC key-pair generation using GA	Provides a structured GA workflow with selection, crossover, and mutation.	Concerns asymmetric-key generation rather than AES internal transformations.
[7]	Review of DNA-based cryptography	Illustrates alternative bio-inspired cryptographic design directions.	Not an evolutionary AES S-Box or MixColumns study.
[9]	GA applications in cryptography	Discusses opportunities and computational costs of evolutionary cryptographic design.	Provides limited component-specific reproducibility and validation.
[13]	Genetic algorithms in cryptography	Reviews or demonstrates GA use in cryptographic construction.	Does not provide the fixed affine encoding adopted in this work.
[18]	Cloud-data cryptosystem based on GA	Shows an application-level GA cryptosystem for data security.	Targets a complete cloud-security scheme rather than AES component validation.

This paper focuses on a fully reproducible GA optimisation of the S-Box affine parameters and on structural validation of a separately obtained GA-derived MixColumns candidate. The distinction is important: complete repeated-run optimisation settings are reported for the S-Box, whereas the MixColumns contribution in this paper is limited to deterministic reconstruction, mathematical validation, integration, and full-cipher evaluation.

The contributions are threefold. First, a fixed 72-bit chromosome directly encodes an invertible 8×8 affine matrix and an 8-bit affine constant for generating an AES-compatible S-Box. Second, the selected S-Box and the alternative MixColumns matrix are evaluated using component-appropriate criteria, including nonlinearity, differential uniformity, LAT magnitude, SAC, fixed-point conditions, invertibility, the MDS property, and branch number. Third, the fixed components are integrated into AES-128 and evaluated through correctness checks and the official NIST SP 800-22 Statistical Test Suite.

The remainder of the paper is organised as follows. Section 2 reviews related work. Section 3 presents the S-Box optimisation, MixColumns validation, full-cipher integration, and NIST testing methodology. Section 4 reports the component-level and full-cipher results. Section 5 concludes the study and outlines future work.

2 Literature Review

A broad review of cryptographic algorithms, weaknesses, and research challenges is provided in [15]. Within evolutionary cryptography, earlier studies have applied GA concepts to symmetric-key randomness, key-sequence generation, operator design, ECC key generation, and cloud-oriented cryptosystems. Table 1 summarises these studies using claims that are directly relevant to the present work.

Research on S-Box construction is more directly related to the nonlinear layer considered here. Existing work includes surveys of S-Box strategies, statistical analysis of the Rijndael S-Box, elliptic-curve and Möbius constructions, GA-generated S-Boxes, software libraries, and key-dependent designs. Table 2 summarises the principal distinction between these

Table 2: Prior S-Box design and evaluation studies

Ref.	Primary focus	Relevance	Scope relative to this study
[2]	Strategies for enhanced AES S-Box design	Summarises design criteria and alternative S-Box approaches.	Does not provide the fixed 72-bit affine-parameter GA used here.
[16]	Development of dynamic S-Box generation	Reviews dynamic and key-dependent S-Box concepts.	The present S-Box is generated offline and remains fixed during encryption.
[6]	Statistical analysis and enhancement of the Rijndael S-Box	Highlights the need for component-level statistical and cryptographic evaluation.	Uses a different enhancement procedure and does not optimize affine parameters using the reported GA.
[4]	Elliptic-curve and Möbius S-Box generation for image encryption	Demonstrates a non-GA algebraic construction and application-level evaluation.	Uses a different construction and targets image encryption.
[11]	GA-generated S-Boxes for IoT security	Directly motivates evolutionary S-Box generation and cryptographic assessment.	Uses dynamic S-Boxes, whereas this work selects one reproducible fixed mapping.
[1]	Software library combining ECC and GA-based dynamic S-Boxes	Demonstrates an accessible implementation of evolutionary S-Box generation.	Primarily a software contribution and does not evaluate the same fixed affine search space.
[8]	Dynamic key-dependent S-Box for IoT systems	Explores key-dependent substitution for constrained environments.	Regenerates or varies the substitution layer; the proposed mapping remains fixed after offline optimisation.

studies and the present offline fixed-component approach.

The review reveals three recurring issues: ambiguous use of the term "dynamic," incomplete disclosure of the chromosome-to-component mapping, and limited separation between component-level properties and full-cipher statistical behaviour. The present study addresses these issues by using an explicit fixed-length affine encoding, reporting repeated GA runs for the S-Box, validating the MixColumns candidate independently, and evaluating the integrated cipher under a paired experimental design.

3 Methodology

All evolutionary operations are performed during an offline design stage. The S-Box branch is a fully specified GA optimisation experiment. The MixColumns matrix is a separately obtained GA-derived candidate whose deterministic construction and structural validation are reported in this paper. After validation, both components are fixed and integrated into the AES-128 round structure; the GA is not executed during normal encryption or decryption.

3.1 Overall Research Design

The study follows an offline-design and runtime-encryption workflow, as illustrated in Figure 1. The left-hand side of the figure represents the evolutionary optimisation stage used to construct the final S-Box. It begins with population initialisation, followed by fitness evaluation, selection, crossover, mutation, and termination checking. These operations are repeated until either the stopping conditions are satisfied or the maximum number of generations is reached. The best valid candidate is then used to construct the final S-Box, which is fixed before being integrated into AES-128.

The right-hand side of Figure 1 presents the resulting GA-AES encryption process. The standard AES-128 state arrangement, key expansion, round-key addition, ShiftRows transformation, and final-round structure are retained. The optimised S-Box replaces the

standard round SubBytes transformation, while the validated GA-derived MixColumns matrix replaces the standard MixColumns transformation in Rounds 1–9. The final round omits MixColumns, consistent with the AES specification. After encryption, the resulting ciphertext is supplied to the NIST SP 800-22 evaluation stage.

The figure therefore separates two distinct activities. The first is the GA optimisation of the S-Box, which is fully specified and repeated across independent runs in this paper. The second is the integration of the selected S-Box together with a separately obtained and mathematically validated GA-derived MixColumns candidate. Thus, the term "GA-evolved MixColumns" in the figure denotes the origin of the candidate matrix and does not imply that a new repeated-run MixColumns optimisation experiment was conducted in the present study.

The complete study consists of five stages. First, related GA and S-Box studies are reviewed. Second, the S-Box search space, fitness function, genetic operators, and stopping conditions are defined and executed across ten independent runs. Third, the selected S-Box and MixColumns candidate are validated using component-appropriate mathematical and cryptographic criteria. Fourth, the fixed components are integrated into an AES-128 implementation and verified through known-answer and round-trip checks. Finally, paired ciphertext datasets are generated and evaluated using the official NIST SP 800-22 Statistical Test Suite. This separation between offline optimisation and runtime encryption ensures that the component properties do not vary between sessions.

3.2 GA-Based S-Box Design

3.2.1 Chromosome Representation

Each S-Box candidate is represented by a fixed 72-bit chromosome. The first 64 bits encode the eight rows of an 8×8 binary affine matrix A, while the remaining 8 bits encode the affine constant c:

$$\chi = A_0 \parallel A_1 \parallel \cdots \parallel A_7 \parallel c, \tag{1}$$

where each A_i and c is one byte. This direct encoding avoids the ambiguity of converting variable-length binary segments into an affine transformation.

3.2.2 Population Initialisation and Feasibility

The initial population is generated randomly. A candidate matrix must have rank 8 over $GF(2)$ to be invertible. Rank-deficient matrices produced during initialisation, crossover, or mutation are processed by a feasibility-repair operator. The operator replaces dependent rows with minimally different rows that increase the matrix rank. Only full-rank matrices are evaluated as valid candidates.

For each valid chromosome, the candidate S-Box is generated by

$$S(x) = A \cdot x^{-1} \oplus c, \tag{2}$$

where x^{-1} denotes the multiplicative inverse of x in $GF(2^8)$ using the AES irreducible polynomial, with 0^{-1} defined as 0. Because A is invertible, the resulting mapping is bijective and an inverse S-Box can be constructed.

3.2.3 Fitness Function

The GA fitness function focuses on properties that can vary among the valid affine-equivalent S-Boxes produced by the representation. Let P_{ij} denote the probability that output bit j changes when input bit i is complemented. The mean and maximum SAC deviations are defined as

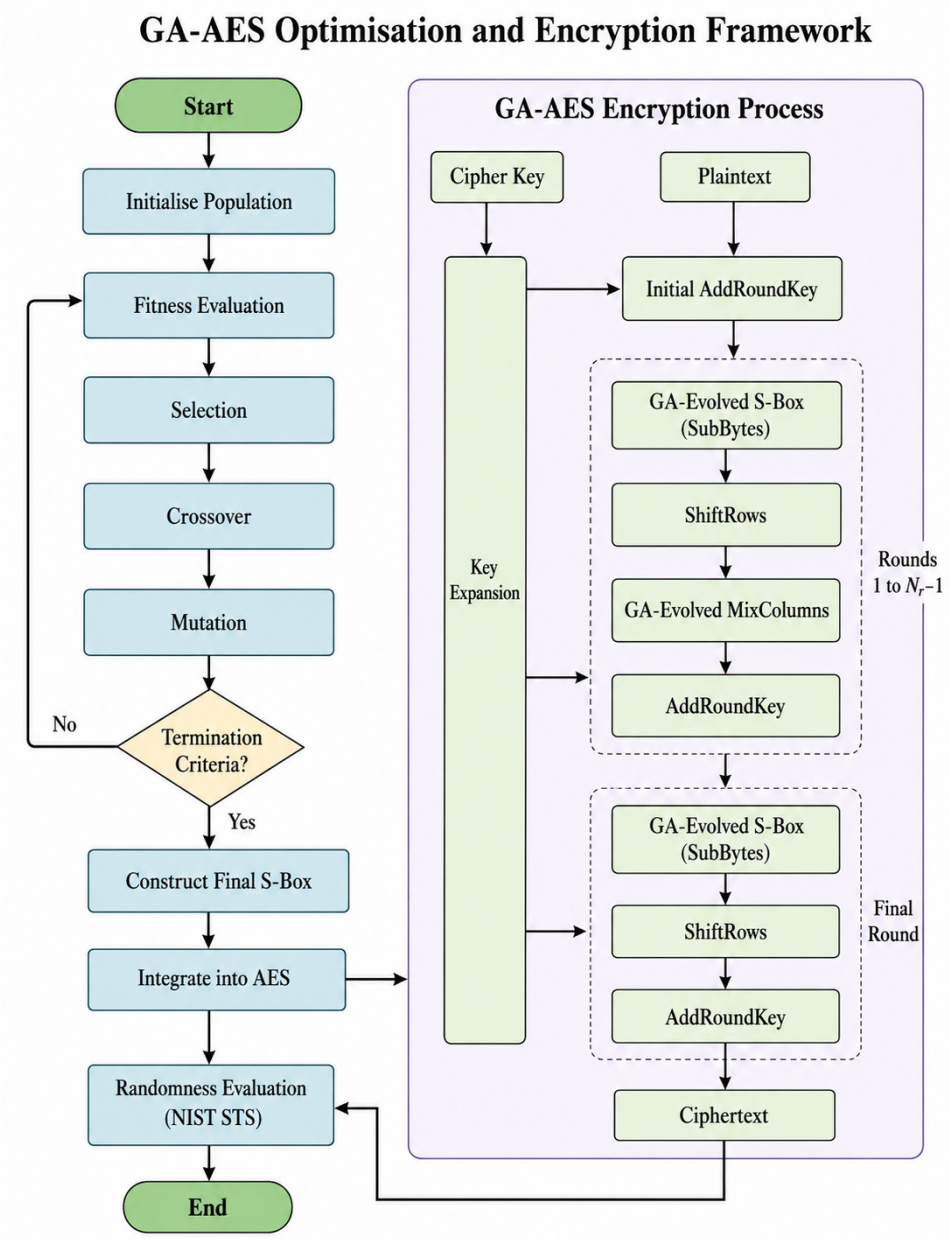


Figure 1: Overall GA-AES optimisation and encryption framework. The left branch shows the offline evolutionary optimisation of the S-Box, while the right branch shows integration of the fixed optimised S-Box and validated GA-derived MixColumns matrix into AES-128, followed by ciphertext randomness evaluation using NIST SP 800-22.

$$E_{\text{mean}} = \frac{1}{64} \sum_{i=0}^{7} \sum_{j=0}^{7} |P_{ij} - 0.5| \,, \tag{3}$$

and

$$E_{\max} = \max_{i,j} |P_{ij} - 0.5| \,. \tag{4}$$

The implemented fitness function is

$$F = 1 - 1.40E_{\text{mean}} - 0.70E_{\max} - 0.025N_{\text{FP}} - 0.025N_{\text{OFP}}, \tag{5}$$

where N_{FP} and N_{OFP} are the numbers of fixed and opposite fixed points, respectively. A larger fitness value indicates SAC values closer to 0.5 and fewer undesirable fixed-point conditions. A BIC-style pairwise avalanche correlation is calculated after optimisation as an additional evaluation metric; it is not included in the fitness function used for the reported experiment.

Nonlinearity, differential uniformity, maximum LAT magnitude, and algebraic degree are reported as cryptographic validation metrics. Under the adopted inversion followed by an invertible affine transformation, these properties are preserved by affine equivalence and therefore are not treated as quantities improved by the GA.

3.2.4 Genetic Operators

Tournament selection with a tournament size of 3 is used to select parents. Two-point crossover is applied to the 72-bit chromosomes with probability 0.90. Independent bit-flip mutation is applied with probability 1/72 per bit. After crossover and mutation, the matrix-repair operator restores full rank when necessary. Elitism retains the four best individuals in each generation.

3.2.5 Experimental Configuration and Termination

The final experiment used a population of 80 individuals, 150 maximum generations, and 10 independent runs. The pseudorandom seed for the first run was 20260731, with subsequent runs using deterministic seed offsets. A run terminates when either the maximum number of generations is reached or the best candidate satisfies all of the following conditions:

$$E_{\text{mean}} \leq 0.015, \qquad N_{\text{FP}} = 0, \qquad N_{\text{OFP}} = 0. \tag{6}$$

The SAC value of 0.015 was used as an aspirational early-termination target rather than a mandatory validity threshold. In the reported experiment, this target was not reached; consequently, the best candidate under the defined fitness function was selected after the maximum of 150 generations.

3.2.6 S-Box Evaluation

The selected S-Box is evaluated using bijectivity, matrix rank, vectorial nonlinearity, coordinate nonlinearities, maximum absolute LAT value, differential uniformity, SAC, algebraic degree, fixed points, opposite fixed points, and a BIC-style average absolute pairwise avalanche correlation. The inverse S-Box is also generated and verified by confirming that $S^{-1}(S(x)) = x$ for all $x \in [0, 255]$.

3.3 GA-Derived MixColumns Candidate and Validation

The alternative MixColumns candidate originated from two 128-bit offspring retained from an earlier GA-based component-generation stage:

$$\text{Offspring 1} = \{\texttt{D2}, \texttt{6D}, \texttt{69}, \texttt{A7}, \texttt{92}, \texttt{B9}, \texttt{E6}, \texttt{54}, \texttt{B4}, \texttt{CB}, \texttt{69}, \texttt{D2}, \texttt{B4}, \texttt{CB}, \texttt{69}, \texttt{D2}\},$$
$$\text{Offspring 2} = \{\texttt{9E}, \texttt{AA}, \texttt{95}, \texttt{6A}, \texttt{B6}, \texttt{63}, \texttt{E2}, \texttt{2B}, \texttt{E9}, \texttt{3A}, \texttt{B6}, \texttt{CF}, \texttt{AC}, \texttt{E9}, \texttt{E9}, \texttt{E9}\}.$$

Bytewise XOR gives

$$\{\texttt{4C}, \texttt{C7}, \texttt{FC}, \texttt{CD}, \texttt{24}, \texttt{DA}, \texttt{04}, \texttt{7F}, \texttt{5D}, \texttt{F1}, \texttt{DF}, \texttt{1D}, \texttt{18}, \texttt{22}, \texttt{80}, \texttt{3B}\}.$$

The bytes are arranged row-wise into

$$M = \begin{bmatrix} \texttt{4C} & \texttt{C7} & \texttt{FC} & \texttt{CD} \\ \texttt{24} & \texttt{DA} & \texttt{04} & \texttt{7F} \\ \texttt{5D} & \texttt{F1} & \texttt{DF} & \texttt{1D} \\ \texttt{18} & \texttt{22} & \texttt{80} & \texttt{3B} \end{bmatrix}. \tag{7}$$

The scope of the present paper is the deterministic reconstruction, mathematical validation, and AES integration of this candidate; a new repeated-run optimisation experiment for M is not claimed. The inverse candidate is

$$M^{-1} = \begin{bmatrix} \texttt{E7} & \texttt{F7} & \texttt{EF} & \texttt{BA} \\ \texttt{25} & \texttt{65} & \texttt{CA} & \texttt{3F} \\ \texttt{61} & \texttt{3C} & \texttt{95} & \texttt{B4} \\ \texttt{35} & \texttt{E0} & \texttt{FF} & \texttt{ED} \end{bmatrix}. \tag{8}$$

Validation was performed over $GF(2^8)$ using the AES irreducible polynomial. Invertibility was checked by finite-field Gaussian elimination and by verifying $MM^{-1} = M^{-1}M = I_4$. The determinant was computed as hexadecimal `84`. The MDS property was verified by confirming that all 69 square minors of orders 1 through 4 were nonzero. For a 4×4 MDS matrix, this establishes the maximum branch number

$$B = \min_{x \neq 0} \left(W(x) + W(Mx)\right) = 5, \tag{9}$$

where $W(\cdot)$ denotes the number of nonzero bytes.

3.4 Full-Cipher Integration and NIST STS Setup

As summarised in Figure 1, the final S-Box and MixColumns matrix were fixed before integration. The implementation retained the AES-128 state layout, ten-round structure, standard AES-128 key expansion, and omission of MixColumns in the final round. The GA-optimised S-Box replaced the round SubBytes transformation, while M replaced standard MixColumns in Rounds 1–9. The standard AES S-Box was retained in the SubWord operation of the key schedule for both implementations so that the comparison isolated the two modified round components.

Implementation correctness was checked before dataset generation. Standard AES reproduced the FIPS-197 known-answer ciphertext for key `000102030405060708090A0B0C0D0E0F` and plaintext `00112233445566778899AABBCCDDEEFF`. In addition, the first encrypted block of each of the 100 generated sequences was decrypted with the corresponding implementation and compared with its original plaintext. All standard-AES and GA-AES checks passed. The S-Box was also verified to contain 256 unique output values, and the MixColumns inverse was verified as described above.

For sequence index i and block counter j, each 128-bit plaintext block was generated as

$$P_{i,j} = \text{uint64}_{\text{BE}}(i) \parallel \text{uint64}_{\text{BE}}(j). \tag{10}$$

A distinct AES-128 key was derived for each sequence from the first 16 bytes of the SHA-256 digest of the ASCII label `paired-sequence-key-{index:03d}`. Identical plaintext blocks and keys were supplied to standard AES and GA-AES, producing paired ciphertext sequences under controlled inputs.

For each algorithm, 100 sequences of 1,000,000 bits were generated. Each sequence contained 125,000 raw ciphertext bytes. Because this length is not divisible by the 16-byte

AES block size, 7,813 blocks were encrypted and the concatenated output was truncated to the first 125,000 bytes. Each complete dataset therefore contained 100,000,000 bits, or 12,500,000 bytes, stored in raw binary form.

The ciphertext was evaluated using NIST STS version 2.1.2 with all 15 SP 800-22 Rev. 1a test categories enabled, a sequence length of 1,000,000 bits, 100 bitstreams, and binary input mode [3]. The uniformity criterion was a P-value greater than 0.0001. For tests involving all 100 sequences, the official report indicated a minimum acceptable passing proportion of approximately 96/100. Random Excursions and Random Excursions Variant used only sequences meeting the cycle requirement, resulting in 58 applicable standard-AES sequences and 61 applicable GA-AES sequences.

For reproducibility, the SHA-256 digests of the final input files were retained in the experiment manifest:

standard AES: `3479c259916b7f55e559087d5a55e142ad45ac85033845468482956462303 92f`,

GA-AES: `aa90c3c85500c6f8ab82e194ed945755 9c5121d992ff45271e255ffbdef6107d`.

4 Results and Discussion

4.1 GA Convergence and Reproducibility

The left branch of Figure 1 corresponds to the repeated S-Box optimisation experiment reported in this subsection. Table 3 summarises the configuration used in the final S-Box optimisation experiment.

Table 3: GA configuration for the final S-Box experiment

Parameter	Value
Population size	80
Maximum generations	150
Independent runs	10
Crossover probability	0.90
Mutation probability	1/72 per bit
Elitism size	4
Tournament size	3
Target SAC mean error	0.015

The final-generation results for the ten independent runs are shown in Table 4. All runs produced full-rank affine matrices, bijective S-Boxes, and zero fixed and opposite fixed points.

The best fitness value of 0.955225 was obtained in Runs 5 and 8. The selected final candidate was taken from Run 5. Although its SAC mean error of 0.016357 did not satisfy the target value of 0.015, it was the best candidate obtained after the maximum of 150 generations.

4.2 Final GA-Optimised S-Box

The selected affine parameters were

$$A = \{\texttt{2F}, \texttt{BD}, \texttt{BE}, \texttt{DA}, \texttt{DB}, \texttt{97}, \texttt{CB}, \texttt{7B}\}, \tag{11}$$

and

Table 4: Final results across ten independent GA runs

Run	Fitness	SAC mean error
1	0.954883	0.016602
2	0.952832	0.018066
3	0.942578	0.017578
4	0.953516	0.017578
5	0.955225	0.016357
6	0.943604	0.016846
7	0.954199	0.017090
8	0.955225	0.016357
9	0.954199	0.017090
10	0.942920	0.017334
Mean	0.950918	0.017090
Standard deviation	0.005495	0.000564

Table 5: Final S-Box generated from the optimised GA-derived affine parameters

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	C3	30	B6	86	B4	58	F3	28	3D	ED	68	65	84	11	36	4C
1	05	A3	18	CA	2F	95	9C	53	7A	0C	57	9D	AA	1B	E7	79
2	D8	99	87	AF	71	10	E5	49	96	E1	DC	A9	FA	3C	C7	98
3	BC	CF	FE	64	52	D6	46	E6	A1	2E	44	6A	0B	55	89	51
4	76	59	D3	B5	4F	09	88	B0	13	BB	EB	73	82	90	CE	EF
5	E9	5D	17	47	E3	63	94	72	A2	FF	B9	5C	56	1A	6F	54
6	A7	A0	CC	D5	37	E0	20	15	7B	32	F0	2D	F4	0D	B7	2B
7	0E	AD	2A	69	7D	50	A6	02	5E	22	DB	19	C9	7E	4E	40
8	B3	27	D4	BD	D9	2C	81	7F	D2	06	D7	F7	75	3E	A8	1D
9	DE	BE	07	97	04	9F	9A	DF	66	5A	F5	62	70	23	91	21
A	8D	DA	41	6B	4B	5B	48	C4	9E	BA	80	39	60	CB	26	61
B	3B	E2	8B	14	8E	F6	FD	D1	EE	B8	F8	F1	8F	7C	67	8C
C	12	9B	B2	34	F9	FC	C5	35	16	6D	AB	E4	AC	A4	C2	38
D	00	EA	3F	92	B1	0A	1F	C0	24	6C	42	0F	08	83	03	4D
E	77	C8	01	33	BF	C6	93	31	1C	1E	F2	EC	AE	A5	78	8A
F	74	4A	25	DD	43	C1	CD	45	D0	FB	29	3A	6E	5F	E8	85

$$c = \texttt{C3}. \tag{12}$$

The matrix achieved rank 8 over $GF(2)$, and the resulting S-Box contained 256 unique output values. Table 5 presents the final mapping. The row and column headings correspond to the high and low hexadecimal nibbles of the input byte, respectively.

Table 6 summarises the cryptographic evaluation. The average SAC value was 0.501709, with individual SAC entries between 0.46875 and 0.53125. The final S-Box had no fixed or opposite fixed points. Its BIC-style average absolute correlation was 0.056707.

The nonlinearity, differential uniformity, maximum LAT magnitude, and algebraic degree are consistent with the inversion-based affine construction. The contribution of the GA is therefore not an improvement of these affine-invariant properties. Instead, the GA searches for a valid affine transformation with favourable SAC behaviour and without fixed or opposite fixed points.

4.3 MixColumns Matrix Evaluation

The finite-field validation confirmed that M is invertible, with determinant `84`, and that multiplication by the stated inverse produces the identity matrix in both orders. All 69 square minors were nonzero; hence, M is MDS. It therefore attains branch number 5, the maximum possible for a 4×4 byte-oriented diffusion matrix. These findings establish the structural suitability of the candidate for integration as a diffusion layer. They should be interpreted as validation of this specific matrix rather than evidence of a newly repeated and fully documented GA optimisation campaign for MixColumns.

4.4 Verification Using the NIST Statistical Test Suite

The final evaluation stage shown in Figure 1 was carried out using the official NIST SP 800-22 Statistical Test Suite. Table 7 presents the official NIST SP 800-22 results for standard AES and GA-AES. For categories with multiple outputs, the table reports the observed range of uniformity P-values and passing proportions. These include 148 Non-overlapping Template instances, eight Random Excursions states, 18 Random Excursions Variant states, two Cumulative Sums directions, and two Serial outputs.

Both implementations passed all 15 NIST SP 800-22 test categories and all 188 reportable test instances. The minimum observed uniformity P-values were 0.006661 for standard AES and 0.003712 for GA-AES, both above 0.0001. For tests involving 100 sequences, the lowest passing proportion for each implementation was 96/100, satisfying the official minimum reported for the experiment.

For the excursion-based tests, the lowest passing proportions were 56/58 for standard AES and 60/61 for GA-AES, exceeding their respective minimum acceptable proportions of 55/58 and 58/61. The proposed GA-AES variant therefore exhibited ciphertext statistical behaviour comparable to standard AES under the same paired-input configuration. These results assess observable statistical properties only and do not independently prove resistance to cryptanalysis or superiority over standard AES.

Table 6: Cryptographic properties of the final GA-optimised S-Box

Property	Result
Bijective	Yes
Affine matrix rank	8
Vectorial nonlinearity	112
Coordinate nonlinearities	112 for all 8 components
Maximum absolute LAT	32
Differential uniformity	4
Average SAC	0.501709
SAC range	0.46875–0.53125
SAC mean error	0.016357
Maximum SAC error	0.03125
BIC-style correlation	0.056707
Algebraic degree	7
Fixed points	0
Opposite fixed points	0

Table 7: Official NIST SP 800-22 results for standard AES and GA-AES

Statistical Test	AES P-value	AES Proportion	GA-AES P-value	GA-AES Proportion
Frequency	0.350485	99/100	0.006661	99/100
Block Frequency	0.319084	99/100	0.383827	98/100
Cumulative Sums (Forward)	0.474986	99/100	0.574903	100/100
Cumulative Sums (Reverse)	0.016717	100/100	0.042808	98/100
Runs	0.224821	99/100	0.048716	99/100
Longest Run of Ones	0.657933	99/100	0.085587	99/100
Binary Matrix Rank	0.924076	99/100	0.798139	100/100
Discrete Fourier Transform	0.983453	99/100	0.616305	99/100
Non-overlapping Template[a]	0.006661–0.987896	96/100–100/100	0.003712–0.987896	96/100–100/100
Overlapping Template	0.867692	98/100	0.437274	98/100
Maurer's Universal	0.867692	98/100	0.350485	97/100
Approximate Entropy	0.080519	99/100	0.455937	100/100
Random Excursions[b]	0.075719–0.779188	56/58–58/58	0.063482–0.723129	60/61–61/61
Random Excursions Variant[c]	0.096578–0.935716	56/58–58/58	0.063482–0.970538	60/61–61/61
Serial (P_1)	0.657933	100/100	0.145326	99/100
Serial (P_2)	0.657933	100/100	0.437274	99/100
Linear Complexity	0.798139	98/100	0.019188	97/100

[a]Range across 148 Non-overlapping Template instances. [b]Range across eight Random Excursions states.
[c]Range across 18 Random Excursions Variant states.

5 Conclusion

This study presented a reproducible offline GA optimisation of an AES-compatible S-Box and structural validation of a separately obtained GA-derived MixColumns candidate. The fixed 72-bit S-Box chromosome directly represented an invertible affine matrix and affine constant. Across ten independent runs, the best candidate achieved fitness 0.955225 and produced a bijective S-Box with average SAC 0.501709, vectorial nonlinearity 112, differential uniformity 4, maximum absolute LAT value 32, algebraic degree 7, and no fixed or opposite fixed points.

The MixColumns candidate was verified as invertible with determinant `84`; its inverse was confirmed by finite-field matrix multiplication, all 69 square minors were nonzero, and the matrix attained branch number 5. Both fixed components were integrated into AES-128. Standard AES reproduced the FIPS-197 known-answer ciphertext, and the sampled round-trip checks for both implementations recovered the original plaintext.

The official NIST SP 800-22 experiment evaluated 100 paired ciphertext sequences of 1,000,000 bits per implementation. Standard AES and GA-AES both passed all 15 test categories and all 188 reportable test instances. The findings support the correctness and acceptable ciphertext statistical behaviour of the implementation under the stated configuration, but they do not independently establish full cryptographic security or superiority over standard AES.

5.1 Future Work

Future work should perform systematic plaintext- and key-avalanche experiments, key-sensitivity analysis, execution-time and memory measurements, and cryptanalytic evaluation of the integrated variant. The MixColumns search should also be repeated as a fully documented optimisation experiment with explicit encoding, fitness criteria, operators, parameters, independent runs, and comparison against standard AES and other MDS matrices. Additional ciphertext campaigns using different plaintext distributions and independently generated keys would further strengthen the empirical evaluation.

Acknowledgements

This research was funded by the Ministry of Higher Education (MOHE), Malaysia, under the Fundamental Research Grant Scheme (FRGS/1/2024/ICT04/USIM/02/1). The authors would like to express their gratitude to Universiti Sains Islam Malaysia (USIM) and MOHE for the support and facilities provided.

References

[1] Salman Ali, Md Asdaque Hussain, Faisal Anwer, Ahwar Khan, and M. I. Alam. pycastx: A python library for lightweight data encryption in IoT using ECC and metaheuristics-based high-entropy keys and dynamic S-Box. *SoftwareX*, 31:102321, 2025.

[2] Sura Nabil Hamed Alsweedy and Sufyan Salim Aldabbagh. Enhancing AES security through advanced S-Box design: Strategies and solutions. *International Research Journal of Innovations in Engineering and Technology*, 8(8):182–192, 2024.

[3] Lawrence E. Bassham, Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, Elaine Barker, Stefan Leigh, Mark Levenson, Mark Vangel, David Banks, Alan Heckert, James Dray, and San Vo. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical Report NIST Special Publication 800-22 Revision 1a, National Institute of Standards and Technology, 2010.

[4] Muhammad Bilal, Ghulam Murtaza, Bilal Demir, Miguel D. Bustamante, and Umar Hayat. An efficient algorithm to generate dynamic substitution-boxes and its applications in image encryption. *Alexandria Engineering Journal*, 116:214–231, 2025.

[5] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer, Berlin, Heidelberg, 2002.

[6] Juliet N. Gaithuru and Majid M. Bakhtiari. Statistical analysis of S-Box in rijndael-AES algorithm and formulation of an enhanced S-Box. *Journal of Information Assurance and Security*, 9:213–221, 2014.

[7] B. T. Hammad, A. M. Sagheer, I. T. Ahmed, and N. Jamil. A comparative review on symmetric and asymmetric DNA-based cryptography. *Bulletin of Electrical Engineering and Informatics*, 9(6):2484–2491, 2020.

[8] Divya James and TKS Lakshmi Priya. An innovative approach for dynamic key-dependent S-Box to enhance security of IoT systems. *Measurement: Sensors*, 30:100923, 2023.

[9] Rajat Jhingran, Vikas Thada, and Shivali Dhaka. A study on cryptography using genetic algorithm. *International Journal of Computer Applications*, 118(20):10–14, 2015.

[10] Sanjay Kumar and Deepmala Sharma. Key generation in cryptography using elliptic-curve cryptography and genetic algorithm. *Engineering Proceedings*, 59(1):59, 2023.

[11] Amina Msolli, Imen Hagui, and Abdelhamid Helali. Dynamic S-Boxes generation for IoT security enhancement: A genetic algorithm approach. *Ain Shams Engineering Journal*, 15(11):103049, 2024.

[12] National Institute of Standards and Technology, Morris J. Dworkin, Meltem Sonmez Turan, and Nicky Mouha. Advanced encryption standard (AES). Technical Report FIPS 197-upd1, National Institute of Standards and Technology, 2023.

[13] Peter Pekarčík, Eva Chovancová, Martin Chovanec, and Martin Štancel. Application of genetics algorithms in cryptography. In *2024 IEEE 28th International Conference on Intelligent Engineering Systems (INES)*, 2024.

[14] Jefferson Rodríguez, Brayan Corredor, and César Suárez. Genetic operators applied to symmetric cryptography. *International Journal of Interactive Multimedia and Artificial Intelligence*, 5(7):39–49, 2019.

[15] Yashar Salami, Vahid Khajehvand, and Esmaeil Zeinali. Cryptographic algorithms: A review of the literature, weaknesses, and open challenges. *Journal of Computer & Robotics*, 16(2):63–115, 2023.

[16] Amandeep Singh, Praveen Agarwal, and Mehar Chand. Analysis of development of dynamic S-Box generation. *Computer Science and Information Technology*, 5(5):154–163, 2017.

[17] K. B. Sudeepa, G. Aithal, V. Rajinikanth, and S. C. Satapathy. Genetic algorithm based key sequence generation for cipher system. *Pattern Recognition Letters*, 133:341–348, 2020.

[18] Muhammad Tahir, Muhammad Sardaraz, Zahid Mehmood, and Shakoor Muhammad. CryptoGA: A cryptosystem based on genetic algorithm for cloud data security. *Cluster Computing*, 24(2):739–752, 2021.

[19] A. Z. Zakaria, S. N. Ramli, C. W. Chuah, C. F. M. Foozy, P. S. S. Palaniappan, and N. F. Othman. Enhancing the randomness of symmetric key using genetic algorithm. *International Journal of Innovative Technology and Exploring Engineering*, 8(8S):327–330, 2019.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 51–75.

Privacy Protection Technology for Principal Component Analysis of Multi-Party Data Based on Multi-Cloud Architecture

Yu En Chen[1], Kai-En Cheng[2], Raylin Tso[1,2], Hao-Yi Hsu[2], Tao-Hsiang Chang[2] and Shu-Chen Yang[2]

[1] Master Program in Information Security, National Chengchi University, Taipei, Taiwan
112791005@nccu.edu.tw,raylin@cs.nccu.edu.tw
[2] Department of Computer Science, National Chengchi University, Taipei, Taiwan
{114753501,112753501,111753501,113753503}@nccu.edu.tw

Abstract. As the demand for privacy protection in data analytics and machine learning continues to grow, secure collaborative computation over multi-party data has emerged as a critical research challenge. We propose a privacy-preserving Principal Component Analysis (PCA) framework based on a multi-cloud architecture, enabling distributed feature extraction without aggregating plaintext data at a cloud server. The proposed approach integrates threshold secret sharing, a similarity transformation, and collaborative computing protocols so that, under the stated semi-honest and non-collusion assumptions, no individual cloud server receives a reconstruction set for the protected statistics. In this framework, data providers exchange secret-shared statistics, while cloud servers compute eigenvalue and eigenvector shares. Providers reconstruct the prescribed eigenpairs, form local reduced-dimensional data, and deliver the intended projection to the data consumer. Experiments on two structured numerical datasets evaluate numerical fidelity, computation time, and behavior as the feature dimension increases. The results support the feasibility of the prototype within the stated deployment and threat model.

Keywords: Privacy-preserving · Secret Sharing · Principal Component Analysis · Multi-party Computation

1 Introduction

In recent years, artificial intelligence (AI) has been widely deployed in domains such as smart homes, autonomous driving, healthcare, and finance [12, 26]. These data-driven systems rely on large-scale data analysis to support automated inference and intelligent decision-making.

In multi-party settings, data holders may be unable or unwilling to share raw data because of privacy, legal, or competitive concerns, creating a strong need for privacy-preserving protocols [7, 20]. However, conventional approaches such as encryption and anonymization are often insufficient for secure multi-party data integration in cloud environments [24]. Public clouds further introduce risks such as external attacks and honest-but-curious servers [15], while the decentralized and dynamic nature of modern data systems complicates traditional protection strategies [1].

To address these issues, Privacy-Enhancing Technologies (PETs) such as homomorphic encryption, differential privacy, zero-knowledge proofs, and secret sharing have been developed [13, 4]. Among them, Shamir's (t, N)-threshold secret sharing supports secure data partitioning and basic homomorphic operations without exposing raw values [30, 17].

In parallel, Principal Component Analysis (PCA) remains a standard dimensionality-reduction technique, but it typically requires access to raw data or the covariance matrix, which creates substantial privacy risks in cloud settings [19].

Existing PET-based solutions still face trade-offs between efficiency and coordination cost, especially in high-dimensional or intermittent computing environments [25, 3].

1.1 Motivation

Applying PCA to privacy-sensitive cloud data can expose raw values to leakage, inference, or misuse [5, 29]. HE-based PCA avoids plaintext disclosure but remains computationally expensive [27]; SS/MPC avoids ciphertext expansion but requires interactive protocols for nonlinear matrix operations [10]. Yen's Shamir-based construction uses two non-colluding clouds, but their collusion permits reconstruction [33]. MSSPCA instead targets multiple providers and clouds, keeps covariance-related values secret-shared, making it suitable for scalable secure computation in decentralized environments.

1.2 Contributions

In this paper, we propose a privacy-preserving principal component analysis (PCA) framework based on threshold secret sharing, specifically designed for multi-party data collaboration in a semi-honest, non-colluding multi-cloud environment. The main contributions are summarized as follows:

1. We construct a PCA computation architecture in which covariance-related statistics remain secret-shared at the cloud layer and are reconstructible only from a valid threshold set.

2. We employ a similarity transformation that preserves eigenvalues and maps transformed eigenvectors back to the original basis, allowing the cloud-side computation to operate on transformed covariance shares.

3. We implement the framework and evaluate its numerical fidelity, computation time, and behavior under increasing feature dimensions. The algebraic analysis establishes the similarity mapping, while the experiments evaluate the numerical prototype.

4. The framework separates the reconstruction threshold t from the number of cloud servers s, allowing different deployment sizes within the same non-collusion threat model.

The remainder of this paper is organized as follows. Section 2 reviews related work on PCA techniques. Section 3 introduces the preliminaries used in this paper. Section 4 presents our system model and threat assumptions. Section 5 describes the proposed framework and computation protocols. Section 6 analyzes the security and correctness of our system. Section 7 provides implementation details and evaluates performance. Finally, Section 8 concludes the paper and discusses future work.

2 Related Work

Principal Component Analysis (PCA) has been widely studied in privacy-preserving data analytics [16]. Existing privacy-preserving PCA approaches mainly fall into three categories: homomorphic encryption, secret sharing/MPC, and dual-cloud architectures. Liu et al. [21] further highlighted the need to protect both input data and intermediate results during outsourced PCA computation.

2.1 Homomorphic Encryption-Based Approaches

Homomorphic encryption (HE) enables computation on encrypted data and is therefore attractive for privacy-preserving outsourcing [2]. Panda et al. [27] implemented PCA using CKKS encryption, but ciphertext expansion and the overhead of encrypted matrix operations limit scalability in large-scale or time-sensitive settings. Ma et al. [23] improved the efficiency of HE-based PCA through optimized homomorphic matrix multiplication; nevertheless, practical constraints remain, including limited multiplicative depth, large switching/rotation-key storage, and memory overhead caused by intermediate ciphertexts during encrypted matrix operations. Existing HE-based PCA schemes also mainly target image-oriented scenarios, whereas our work focuses on structured numerical data.

2.2 Secret Sharing and MPC-Based Approaches

Secret sharing (SS) and secure multiparty computation (MPC) provide an alternative design space by distributing randomized shares across computing parties [30, 11]. Xia et al. [32] used additive secret sharing to support privacy-preserving image retrieval, including secure dimension-reduction operations. Because their application, workload, and system model differ from those of MSSPCA, we do not infer a direct efficiency disadvantage without a common implementation or cost model. Liu et al. [22] proposed both HE- and SS-based PCA frameworks; their SS-based design reconstructs intermediate results, including the covariance matrix, at specified stages. Fan et al. [10] developed an MPC-based PCA system with dedicated covariance and eigendecomposition protocols. These studies demonstrate the feasibility of secret-sharing-based PCA, while MSSPCA focuses on coordinating multiple independent data providers and multiple cloud servers under the deployment assumptions stated in Section 4.2.

2.3 Secret Sharing-Based Dual-Cloud Approaches

Yen [33] proposed a dual-cloud PCA framework in which a single data owner splits the dataset and random vectors into two shares and sends them to two non-colluding cloud servers for joint PCA computation without plaintext reconstruction. Its deployment is centered on one data owner and requires both cloud servers to participate. MSSPCA retains a non-collusion assumption but addresses the distinct setting in which multiple independent providers contribute secret-shared statistics to a multi-cloud workflow.

3 Preliminaries

3.1 Secret Sharing

The proposed construction adopts Shamir's Secret Sharing scheme for secure multiparty PCA. By using polynomial interpolation over a finite field, it supports efficient secret distribution and recovery, as well as homomorphic operations on shares, which are essential for privacy-preserving computation without revealing intermediate values.

Homomorphic Properties

Shamir's secret sharing supports linear homomorphic operations, allowing computation directly on shares without reconstructing the underlying secret.

Let m be a secret, and let $[m]_i$ denote the share of m held by party i. The supported operations include:

- **Addition on shares of two secrets:**

$$[m_1]_i + [m_2]_i = [m_1 + m_2]_i \tag{1}$$

This property holds only when the shares have the same x-coordinate in the Shamir secret sharing scheme.

- **Addition of a constant to a share:**

$$[m]_i + \gamma = [m + \gamma]_i \tag{2}$$

Each party can add a public constant γ directly to their share, resulting in a sharing of the updated secret $m + \gamma$.

- **Multiplication of a share by a constant:**

$$\gamma \cdot [m]_i = [\gamma \cdot m]_i \tag{3}$$

Each party can multiply their share by a public constant γ, resulting in a sharing of the scaled secret $\gamma \cdot m$.

In our framework, multiplication of two secret-shared values is realized using Beaver triples, which provide lower online and communication cost than alternative approaches.

3.2 Beaver Triples

In threshold secret sharing, direct multiplication of two secret shares increases the polynomial degree beyond the reconstruction threshold. To address this issue, we adopt the Beaver triple protocol [6], which enables secure multiplication using pre-shared random values without revealing the underlying secrets.

Let x and y be two secret values held by the parties in the form of shares $[x]_i$ and $[y]_i$. A random Beaver triple (a, b, c) is generated in advance, where a and b are random secrets, and $c = a \cdot b$. Each party receives corresponding shares $[a]_i$, $[b]_i$, and $[c]_i$.

To securely compute the product $z = x \cdot y$, each party locally computes:

$$[\alpha]_i = [x]_i - [a]_i, \quad [\beta]_i = [y]_i - [b]_i. \tag{4}$$

Each party broadcasts $[\alpha]_i$ and $[\beta]_i$ to the other participants, allowing all parties to reconstruct

$$\alpha = x - a, \quad \beta = y - b.$$

Since a and b are random and independent of x and y, revealing α and β does not leak private information.

Each party then computes the final share of $z = x \cdot y$ as:

$$[z]_i = \alpha\beta + \alpha[b]_i + \beta[a]_i + [c]_i. \tag{5}$$

This protocol computes shares of $z = x \cdot y$ under the stated semi-honest assumptions.

Figure 1 illustrates the Beaver-triple-based secure multiplication workflow used in our framework.

3.3 Similarity Transformation

To conceal the structure of the covariance matrix while preserving its eigenvalue-related properties, we apply a similarity transformation. Two square matrices A and A' are similar if there exists an invertible matrix P such that

$$A' = P^{-1}AP \tag{6}$$

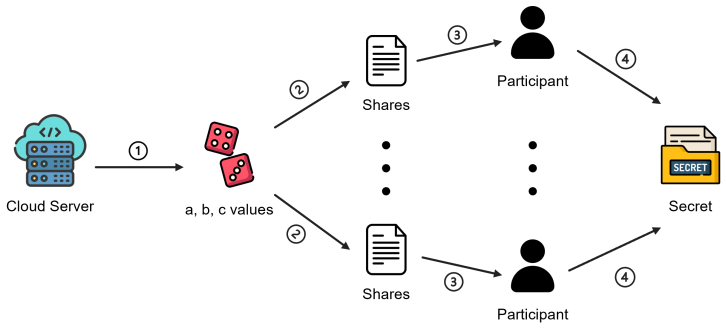


Figure 1: Workflow of Beaver triple-based secure multiplication.

This transformation preserves eigenvalues, allowing eigenvalue computation to be performed on the similar matrix [14].

In our framework, this property is used to obfuscate the global covariance matrix C. Specifically, we compute

$$C' = P^{-1}CP \tag{7}$$

and perform eigen-decomposition on C'. If $C'v'_j = \lambda_j v'_j$, then $v_j = Pv'_j$ satisfies $Cv_j = \lambda_j v_j$ [14]. Thus, the transformation preserves the eigenvalues and changes the coordinate representation of the covariance matrix and its eigenvectors.

The similarity transformation changes the coordinate representation but preserves the eigenvalue spectrum required by PCA. In MSSPCA, the transformed covariance remains secret-shared during cloud-side computation, while the data providers retain P for restoring the eigenvectors.

3.4 Inverse of Square Root

In our framework, all computations are performed on shared values, but shared arithmetic does not directly support non-linear operations such as square roots. To normalize vectors during iterative procedures such as the Power Method, we therefore approximate the inverse square root $\frac{1}{\sqrt{x}}$ using Taylor-series initialization followed by Newton's method refinement.

Taylor Expansion. To obtain an initial approximation of $\frac{1}{\sqrt{x}}$, we expand $f(x) = x^{-1/2}$ at a reference point x_r and truncate the series at order o [18]:

$$T(x) = \sum_{k=0}^{o} \frac{f^{(k)}(x_r)}{k!}(x - x_r)^k.$$

This yields an initial estimate $y_0 \approx \frac{1}{\sqrt{x}}$. For better convergence, x_r should be close to x.

Newton's Method. We refine the initial estimate y_0 by Newton iteration:

$$y_i = \frac{1}{2} y_{i-1}(3 - x \cdot y_{i-1}^2).$$

After τ iterations, we obtain $y_\tau \approx \frac{1}{\sqrt{x}}$ [9].

4 System Architecture

This section describes the participating entities, their interactions, and the corresponding threat model.

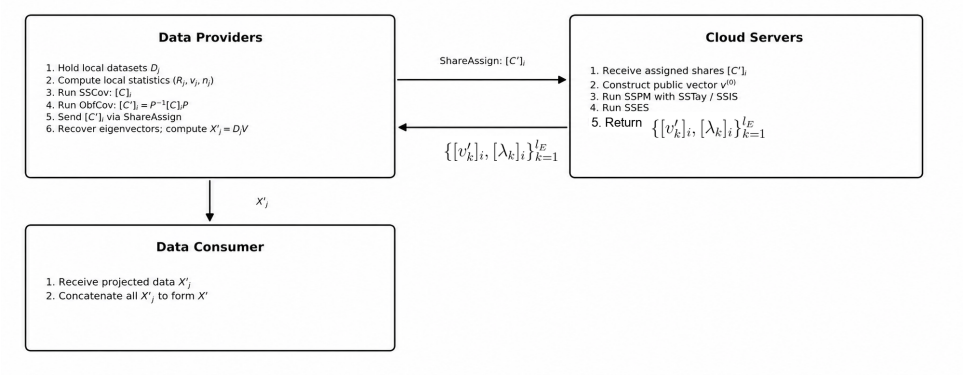


Figure 2: Detailed workflow of the proposed MSSPCA framework.

4.1 System Model

Figure 2 illustrates the overall workflow of our privacy-preserving PCA framework. The system consists of multiple data providers, multiple non-colluding cloud servers, and a data consumer. Raw datasets remain with their providers; the consumer receives the intended reduced-dimensional output.

1. **Data Providers:** Data providers are semi-honest entities, each holding a private local dataset. They compute local statistics, generate secret shares of covariance-related values, collaboratively construct the obfuscation matrix, and send the resulting obfuscated covariance shares to designated cloud servers via `ShareAssign`. After cloud-side computation, each provider reconstructs the transformed eigenpairs, restores the original-basis eigenvectors, and performs dimensionality reduction locally.

2. **Cloud Servers:** The framework involves multiple semi-honest, non-colluding cloud servers. They jointly construct a public initialization vector, receive obfuscated covariance shares from the data providers via `ShareAssign`, and execute the secure eigendecomposition procedures to compute the top-l_E transformed eigenvector shares and eigenvalue shares of the obfuscated covariance matrix. Under the sub-threshold assignment condition, no individual server reconstructs the protected matrix or eigenpairs.

3. **Data Consumer:** After dimensionality reduction, the data consumer receives the reduced-dimensional data in plaintext from the providers, combines the results, and uses them for downstream tasks such as model training, evaluation, or predictive analysis.

Remark. The temporary variable S used in several protocols denotes intermediate storage for shares during the local arithmetic steps.

4.2 Threat Model

In this framework, all data providers and cloud servers are assumed to be semi-honest: they follow the prescribed protocol but may attempt to infer sensitive information from the data they process or receive. Cloud servers are additionally assumed to be non-colluding and sufficiently available for share reconstruction. The data consumer is an authorized recipient of the final reduced-dimensional data and is not a share-holding party in the analyzed protocols. The threat model considers interception, limited provider collusion, and one passively curious cloud server combining its view with a few colluding providers.

For each secret-shared value, the adversarial view is assumed to remain below threshold t. Protocol messages that carry secret shares are transmitted over secure channels that provide confidentiality and integrity.

5 Proposed Protocols

Table 1 summarizes the main notation used throughout the proposed framework.

Table 1: Notation

Notation	Description
D_j	Plaintext dataset of size $n_j \times d$.
R_j, v_j, n_j	Outer product, sum vector, and sample count of D_j.
N	Number of data providers.
$n = \sum_j n_j$	Total number of samples across all providers.
s	Number of cloud servers.
t	Reconstruction threshold in the Shamir sharing scheme.
P	Provider-side random invertible matrix jointly derived by data providers; not disclosed to cloud servers.
$v^{(0)}$	Public random initial vector.
$[C]_i$	Secret share of the global covariance matrix.
$[C']_i$	Share of the obfuscated covariance matrix.
$[C'_k]_i$	Share of the shifted covariance matrix.
$[\lambda_k]_i$	Secret share of the k-th eigenvalue.
$[v'_k]_i$	Secret share of the transformed k-th eigenvector computed from C'.
v_k	Restored k-th eigenvector in the original basis, computed as $v_k = Pv'_k$.
$[x'_0]_i$	Initial InvSqrt approximation from `SSTay`.
$[x''_0]_i$	Refined InvSqrt result from `SSIS`.
l_E	Number of eigenpairs extracted for dimensionality reduction.
l_p	Number of SSPM iterations.
o	Taylor truncation order.
B	Upper bound for InvSqrt approximation.
τ	Number of Newton iterations in `SSIS`.
X'	Dimensionality-reduced matrix.

5.1 Sub-Protocol Description

The following protocols use the Beaver multiplication defined in Section 3 and are composed in Protocol 9. Unless one operand is public, products of secret-shared values are evaluated through this multiplication protocol.

Covariance Matrix. We use a secret-sharing-based protocol (Protocol 1) to compute the covariance matrix without exposing raw data or intermediate statistics and without involving cloud servers.

Obfuscated Covariance Matrix. Protocol 2 computes an obfuscated covariance matrix via a similarity transformation. All providers learn the matrices exchanged within the provider set, whereas the non-colluding cloud servers receive only assigned shares and are not sent P. The transformation preserves eigenvalues and changes the matrix representation.

Protocol 1 Secret Sharing-based Covariance Matrix (SSCov)

DP$_j$ **Input:** Local dataset $D_j = \{x_1, \ldots, x_{n_j}\}$
DP$_i$ **Output:** Share of the global covariance matrix $[C]_i$

1. Each DP$_j$ computes $R_j = \sum_{i=1}^{n_j} x_i x_i^\top$, $v_j = \sum_{i=1}^{n_j} x_i$, and the sample count n_j.
2. Each DP$_j$ generates N shares $[R_j]_1, \ldots, [R_j]_N$ and $[v_j]_1, \ldots, [v_j]_N$ via secret sharing, then sends each share to the other providers; n_j is sent to all providers directly.
3. Each DP$_i$ aggregates the received shares: $[R]_i = \sum_j [R_j]_i, \quad [v]_i = \sum_j [v_j]_i, \quad n = \sum_j n_j$
4. Each DP$_i$ locally computes its share of the global covariance matrix as:

$$[C]_i = \frac{1}{n-1}[R]_i - \frac{1}{n(n-1)}[v]_i[v]_i^\top$$

where the outer product of the shared vector is evaluated using Beaver multiplication.

Protocol 2 Obfuscated Covariance (ObfCov)

DP$_i$ **Input:** Share of global covariance matrix $[C]_i$
DP$_i$ **Output:** Share of the obfuscated covariance matrix $[C']_i$

1. Each data provider DP$_j$ independently generates a random invertible matrix $P_j \in \mathbb{R}^{d \times d}$ and broadcasts it to all other providers.
2. All data providers receive the set $\{P_1, P_2, \ldots, P_N\}$ and compute the product: $P = P_1 P_2 \ldots P_N$
3. Each DP$_i$ locally computes the inverse matrix P^{-1}.
4. Each DP$_i$ performs the similarity transformation:

$$[C']_i = P^{-1}[C]_i P$$

Protocol 3 Modulo-Based Share Assignment (ShareAssign)

Input: Number of cloud servers s, total list of data providers $\mathcal{P}$ with indices p
Output: Each CS$_r$ receives an assigned subset of obfuscated covariance matrix shares $\{[C']_i\}$ without changing their Shamir coordinates

1. Label each data provider in $\mathcal{P}$ with a unique index $p \in \{0, 1, \ldots, |\mathcal{P}| - 1\}$.
2. Let the cloud servers be $\{\mathrm{CS}_0, \mathrm{CS}_1, \ldots, \mathrm{CS}_{s-1}\}$.
3. For each data provider p, compute $r = p \bmod s$.
4. Assign the coordinate share held by data provider p to cloud server CS$_r$ according to:

$$\mathrm{DataProvider}_p \longrightarrow \mathrm{CS}_{(p \bmod s)}$$

5. Each data provider transmits its obfuscated covariance matrix share $[C']_i$ to its assigned cloud server; the share keeps coordinate i after routing.

Protocol 4 Secret Sharing Taylor Expansion (SSTay)

CS$_i$ **Input:** $[x_0]_i$, B, o
CS$_i$ **Output:** $[x'_0]_i$

1. CS$_i$ uses the Taylor expansion

$$T(x_0) = \sum_{k=0}^{o} \frac{f^{(k)}(x)}{k!}(x_0 - x)^k$$

to compute the approximation $\left[\frac{1}{\sqrt{x_0}}\right]_i$ at point $x = B/2 + 1$, truncated to order o.

Protocol 5 Secret Sharing InvSqrt (SSIS)

CS$_i$ **Input:** $[x_0]_i$, $[x'_0]_i$, τ
CS$_i$ **Output:** $[x''_0]_i$

1. Set $[y_0]_i \leftarrow [x'_0]_i$ and create a temporary variable S to store the calculated share.
2. **for** $j = 1$ to $\tau - 1$ **do**
3. Calculate $[y_{j-1}]_i \cdot [y_{j-1}]_i \cdot [x_0]_i$ and store the result in S
4. Calculate $\frac{1}{2}[y_{j-1}]_i \cdot (3 - S)$ and store the result in $[y_j]_i$
5. **end for**
6. Set $[x''_0]_i \leftarrow [y_{\tau-1}]_i$
7. **return** $[x''_0]_i$

Provider Selection and Distribution. Protocol 3 assigns data providers to cloud servers. Once the participation threshold is met, the selected providers send their obfuscated covariance shares to the assigned servers, enabling secure eigendecomposition on secret-shared data.

Inverse Square Root. Because the Power Method requires vector normalization while square-root operations are not directly supported on shares, Protocols 4 and 5 jointly provide a secure approximation of the reciprocal square root.

Eigenvalues and Eigenvectors. To compute multiple eigenpairs, the framework alternates between the Secret Sharing Power Method (Protocol 6), which extracts the dominant eigenpair, and Secret Sharing Eigenshift (Protocol 7), which removes the contribution of previously extracted components.

Dimensionality Reduction. Protocol 8 enables each provider to recover the original eigenvectors using P, project its local dataset into the reduced subspace, and send the resulting reduced data to the consumer.

5.2 MSSPCA

We integrate the above sub-protocols into a complete secret-sharing-based PCA framework, termed **MSSPCA**, which coordinates data providers, cloud servers, and a data consumer to perform dimensionality reduction without exposing raw data. The full procedure is given in Protocol 9.

Cloud servers do not receive provider plaintext data, while data providers participate in the initial sharing/obfuscation phase and the final reconstruction/projection phase.

Protocol 6 Secret Sharing Power Method (SSPM)

CS$_i$ **Input:** $[C']_i$, $v^{(0)}$, l_p, B, o, τ
CS$_i$ **Output:** $[v'_j]_i$, $[\lambda_j]_i$

1. Create a temporary variable S to store intermediate values.
2. **for** $j = 1$ to l_p **do**
3. **if** $j = 1$ **then**
4. $[v^{(j)}]_i \leftarrow [C']_i \cdot v^{(0)}$
5. **else**
6. $[v^{(j)}]_i \leftarrow [C']_i \cdot [v^{(j-1)}]_i$
7. **end if**
8. $S \leftarrow \mathrm{SSTay}([v^{(j)}]_i, B, o)$
9. $S \leftarrow \mathrm{SSIS}([v^{(j)}]_i, S, \tau)$
10. $[v^{(j)}]_i \leftarrow [v^{(j)}]_i / S$
11. **end for**
12. $[v'_j]_i \leftarrow [v^{(l_p)}]_i$
13. $[\lambda_j]_i \leftarrow [v'_j]_i^\top \cdot ([C']_i \cdot [v'_j]_i)$
14. **return** $[v'_j]_i, [\lambda_j]_i$

Protocol 7 Secret Sharing Eigenshift (SSES)

CS$_i$ **Input:** $[C']_i$, $[v'_j]_i$, $[\lambda_j]_i$
CS$_i$ **Output:** $[C'_k]_i$

1. **for** $j = 1$ to $k - 1$ **do**
2. $[C'_{j+1}]_i = [C'_j]_i - [\lambda_j]_i \cdot [v'_j]_i \cdot [v'_j]_i^\top$
3. **end for**
4. Each CS$_i$ outputs the shifted covariance matrix $[C'_k]_i$ for calculating the subsequent eigenvalues and eigenvectors.

Protocol 8 Dimensionality Projection (DPN)

DP$_j$ **Input:** D_j, threshold reconstruction sets of $\{[\lambda_k]_i\}_{k=1}^{l_E}$ and $\{[v'_k]_i\}_{k=1}^{l_E}$, P
Consumer Output: X'

1. Each DP$_j$ reconstructs all transformed eigenpairs (λ_k, v'_k) from threshold reconstruction sets of shares $\{[\lambda_k]_i\}_{k=1}^{l_E}$ and $\{[v'_k]_i\}_{k=1}^{l_E}$.
2. Each DP$_j$ restores all eigenvectors $\{v_k\}_{k=1}^{l_E}$ by computing: $v_k \leftarrow P v'_k$ for all $k = 1$ to l_E.
3. Each DP$_j$ constructs the eigenvector matrix $V \in \mathbb{R}^{d \times l_E}$ from the l_E selected eigenvectors.
4. Each DP$_j$ computes the projected dataset: $X'_j = D_j V$.
5. The consumer forms the full reduced matrix by concatenation:

$$X' = \begin{pmatrix} X'_1 \\ \vdots \\ X'_N \end{pmatrix}$$

Protocol 9 Multi-Data Secret Sharing Principal Component Analysis (MSSPCA)

DP$_j$ Input: D_j
CS$_i$ Input: l_E, l_p, o, B, τ
Data Consumer Output: X'

1. Each DP$_j$ computes its local sufficient statistics and the DPs execute **SSCov**; each DP$_i$ obtains its coordinate share $[C]_i$ of the global covariance matrix.
2. Each DP$_j$ generates a local random invertible matrix and exchanges it with other providers. All DPs multiply the N matrices in fixed order to derive a shared matrix P.
3. Each DP$_i$ uses matrix P to compute its obfuscated covariance share $[C']_i$ via the **ObfCov** protocol.
4. Each CS$_i$ generates a local random initial vector, and all CS exchange and add their vectors to derive a common public vector $v^{(0)}$.
5. The system executes **ShareAssign** to route the coordinate shares $[C']_i$ to cloud servers while maintaining the sub-threshold assignment condition.
6. **for** $k = 1$ to l_E **do**
7. Each CS$_i$ executes **SSPM** using $v^{(0)}$ and the current covariance share (initially $[C']_i$) to compute transformed eigenvector shares $[v'_k]_i$ and eigenvalue shares $[\lambda_k]_i$. If $k < l_E$, it then executes **SSES** to update the covariance share used in the next iteration.
8. **end for**
9. Each CS$_i$ sends shares of all transformed eigenpairs $\{[v'_k]_i, [\lambda_k]_i\}_{k=1}^{l_E}$ to every DP$_j$.
10. Each DP$_j$ reconstructs transformed eigenpairs, recovers original-basis eigenvectors via P, and runs **Dimensionality Projection** to compute X'_j, then sends X'_j to the consumer.
11. The data consumer forms X' by concatenating all received X'_j.

Under the stated sub-threshold and non-collusion assumptions, the protected intermediate statistics remain secret-shared at the cloud layer.

6 Security Analysis

In this section, we analyze MSSPCA under the semi-honest model of Section 4.2. The adversary obtains at most $t-1$ shares for each protected value, while the cloud servers remain non-colluding. Each component functionality captures the behavior of a protocol component, and the corresponding simulator argument reproduces the view of the corrupted participants from their inputs, observable messages, and prescribed outputs. The detailed component functionalities and simulator arguments are provided in Appendix A.

6.1 Obfuscation Mechanism Analysis

We now state the privacy boundary of the similarity-transform layer. Recall from Protocol 2 that the obfuscated covariance is

$$C' = P^{-1}CP, \qquad P = P_1 P_2 \cdots P_N,$$

where each provider contributes an independently sampled invertible factor P_j according to the prototype described in Section 5.

Because C' is similar to C, the transformation preserves the eigenvalues and changes the coordinate representation of the covariance matrix and its eigenvectors. The data providers derive P from the exchanged matrices $P_1, \ldots, P_N$ and use it to restore the

eigenvectors after reconstruction. The cloud servers receive assigned shares of C' through `ShareAssign`; P is not included in this transfer.

The preserved eigenvalue spectrum contains the variance information required by PCA and may also reveal variance magnitude, concentration, or multiplicity if reconstructed. In MSSPCA, covariance and eigenpair values remain secret-shared during cloud-side computation. Their confidentiality therefore follows from Shamir secret sharing and the semi-honest, non-collusion assumptions rather than from the similarity transformation alone [19].

6.2 Security Proof Sketch

We summarize the security argument by protocol type. The purpose of this subsection is to show how the component simulators cover the real protocol views; the detailed functionalities and simulator constructions are deferred to Appendix A.

Theorem 1 (Main security theorem)**.** *Under the stated semi-honest, non-collusion, and threshold-bounded assumptions, the corrupted participants' views are simulatable from their own inputs, public parameters, observable protocol messages, assigned shares, and prescribed outputs. The component simulator arguments therefore support the intended privacy-preserving PCA functionality of MSSPCA.*

Lemma 1 (Beaver multiplication)**.** *The Beaver-triple-based multiplication sub-protocol does not reveal the multiplied secrets beyond the prescribed product shares under the semi-honest model.*

Proof sketch. A corrupted participant observes its input shares, Beaver triple shares, the opened masked differences, and its output share. The opened values are

$$\alpha = x - a, \qquad \beta = y - b,$$

where a and b are random and independent of the operands. Hence α and β can be simulated without revealing x or y. Given the corrupted inputs and output shares, the simulator samples compatible triple shares and honest-party broadcasts as in the Beaver multiplication argument of Appendix A. Thus the corrupted view is reproducible from the prescribed information. □

Lemma 2 (Provider-side covariance, obfuscation, and share routing)**.** *The protocols **SSCov**, **ObfCov**, and **ShareAssign** reveal no honest provider's raw dataset or global covariance value beyond sub-threshold shares and prescribed outputs.*

Proof sketch. In **SSCov**, each provider shares only local sufficient statistics, and any corrupted set below the reconstruction threshold sees fewer than t shares of each protected value. In **ObfCov**, the providers jointly derive the invertible matrix P and locally compute $[C']_i = P^{-1}[C]_i P$; the cloud servers receive assigned shares of C' but are not sent P. In **ShareAssign**, the routing rule depends only on public indices and the number of cloud servers, and routing does not change the Shamir coordinate of a share. Therefore, the simulator can reproduce the corrupted view from corrupted inputs, public routing information, assigned shares, and prescribed outputs. □

Lemma 3 (Cloud-side eigenpair computation)**.** *The protocols **SSTay**, **SSIS**, **SSPM**, and **SSES** reveal only public parameters, local shares, and prescribed output shares to semi-honest corrupted cloud servers.*

Proof sketch. In **SSTay** and **SSIS**, each cloud server evaluates deterministic approximation routines on its own input shares and public parameters. In **SSPM**, the server uses its current covariance share, the public initialization vector, and local intermediate shares

to compute transformed eigenvector shares and eigenvalue shares. In **SSES**, the server updates its shifted covariance share using the transformed eigenvector and eigenvalue shares. Products of shared values are handled through the Beaver multiplication sub-protocol. Since no protected value is reconstructed by a sub-threshold corrupted set, the simulator can reproduce the corrupted cloud-server view from the corrupted shares, public parameters, and prescribed output shares. □

Lemma 4 (Dimensionality projection)**.** *The **DPN** protocol reveals to a corrupted provider only its own dataset, the reconstructed prescribed eigenpairs, the transformation matrix P, and its own projected output.*

Proof sketch. After receiving threshold reconstruction sets of eigenvalue shares and transformed eigenvector shares, each provider reconstructs the transformed eigenpairs, restores the original-basis eigenvectors via $v_k = Pv'_k$, and computes $X'_j = D_j V$ locally. A corrupted provider's view is therefore determined by its own dataset, the received reconstruction shares, P, and its own projected output. The consumer receives the projected matrices and forms the final reduced matrix by concatenation. No additional honest provider raw dataset is reconstructed in this phase. □

Combining these lemmas, each protocol component exposes only the information described by its inputs, observable messages, and prescribed outputs under the stated threshold and non-collusion assumptions. The detailed component functionalities and simulator arguments in Appendix A instantiate this reasoning for Beaver multiplication, SSCov, ObfCov, ShareAssign, SSTay, SSIS, SSPM, SSES, and DPN, which together establish Theorem 1.

7 Experiments and Results

This section evaluates the proposed **MSSPCA** framework in terms of numerical correctness, computation time, and scalability in the prototype setting. Communication and network latency are outside the reported measurements.

7.1 Experimental Setup

We evaluate MSSPCA on two public structured numerical datasets: the Wine Quality dataset [8] and the Insurance Company Benchmark (COIL 2000) dataset [31]. These datasets are selected because they provide complementary evaluation settings for privacy-preserving PCA. The Wine Quality dataset contains physicochemical measurements of red and white vinho verde wine samples and is suitable for validating whether MSSPCA preserves the principal-component variance structure of plaintext PCA on continuous numerical data. In our experiments, the target quality label is removed, and the remaining 11 numerical attributes are used for PCA. The Insurance Company Benchmark dataset contains customer information from an insurance company, including socio-demographic and product-usage attributes. The processed feature set used in our experiments contains 9,822 records and 85 input attributes after target removal. This dataset provides a higher-dimensional and privacy-sensitive customer-analytics scenario, which is useful for evaluating the scalability of secure covariance computation and eigendecomposition.

All experiments are conducted on a single machine that simulates four non-colluding servers under a Shamir secret-sharing setup with four shares and reconstruction threshold $t = 2$, using modulus $2^{31} - 1$. Beaver triples (a, b, c) are generated with dimensions matching the operands in each dataset, and the secret-sharing polynomials $f(x)$ are freshly sampled whenever new shares are created. The implementation is written in Python and executed on a machine with a 12-core 2.3 GHz Intel Core i7 CPU, 32 GB RAM, and an NVIDIA GeForce RTX 3060 Laptop GPU.

Metric	Value
Mean Precision Error	1.039632×10^{-4}
Reference threshold [10]	1.0×10^{-3}

Figure 3: MPE result computed from EVR differences between MSSPCA and scikit-learn PCA.

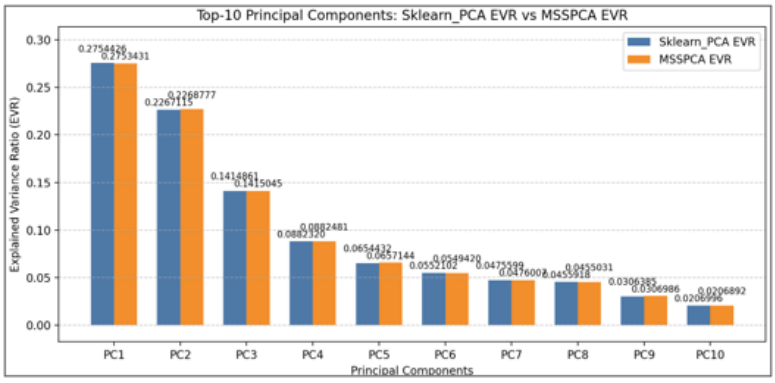


Figure 4: Bar chart of top-10 EVR values from MSSPCA and scikit-learn PCA.

7.2 Evaluation

We evaluate MSSPCA from three empirical perspectives: numerical agreement, computation time, and scaling with feature dimension. For numerical agreement, we compare the explained variance ratios (EVRs) of the first 10 principal components obtained by MSSPCA with those produced by plaintext PCA in Scikit-learn [28].

Following PPCA [10], we quantify the deviation using mean precision error (MPE). The EVR of the i-th principal component is defined as

$$\mathrm{EVR}(\lambda_i) = \frac{\lambda_i}{\sum_{j=1}^{d} \lambda_j}, \tag{8}$$

and the corresponding MPE is computed as

$$\mathrm{MPE} = \frac{1}{k} \sum_{i=1}^{k} \left| EVR_A^{(i)} - EVR_B^{(i)} \right|, \tag{9}$$

where $EVR_A^{(i)}$ and $EVR_B^{(i)}$ denote the EVRs produced by MSSPCA and plaintext PCA, respectively, and $k = 10$ in our evaluation.

On the Wine dataset, MSSPCA achieves an MPE of 1.09×10^{-4}, well below the 1×10^{-3} reference threshold reported in PPCA [10]. This result indicates that MSSPCA closely matches plaintext PCA in principal-component variance preservation.

Figure 3 reports the overall MPE result, and Figure 4 compares the top-10 EVR values from MSSPCA and plaintext PCA. Their close alignment further supports the numerical correctness of the proposed framework.

For efficiency, we measure the runtime of the complete MSSPCA pipeline, including covariance construction, eigen-decomposition, and data projection, following PPCA [10]. To ensure comparability, we compute all eigenvalues and eigenvectors rather than only the top-k components, and exclude communication overhead so that the reported results reflect pure computational cost.

For scalability, we compare the runtime of several eigen-decomposition methods under increasing feature dimensions. As shown in Table 3, the Power Method/Eigenshift approach

Table 2: Execution time breakdown of PPCA and MSSPCA across different phases.

Name	Phase	PPCA (s)	MSSPCA (s)
Wine ($6{,}497 \times 11$)	Covariance Matrix	0.62	0.40
	Eigen Decomposition	1.10	0.75
	Data Projection	0.46	0.31
	Total	**2.18**	**1.46**
Insurance (processed $9{,}822 \times 85$)	Covariance Matrix	18.50	17.20
	Eigen Decomposition	36.70	34.50
	Data Projection	10.82	10.43
	Total	**66.02**	**62.13**

Table 3: Runtime comparison of eigen-decomposition methods under increasing feature dimensions (s).

Method	50	70	100	120
QR Shift	550.14	863.56	1638.07	2531.26
Jacobi	73.69	115.23	228.57	371.45
Jacobi w/ EO-reduction	40.61	76.53	92.24	157.22
Power Method/Eigenshift	12.26	31.04	84.68	139.69

consistently achieves the lowest runtime, indicating better scalability for high-dimensional secure PCA.

To identify practical parameter settings, we examine the impact of iteration depth and approximation precision on numerical error. Our results suggest that about 20 Power Method iterations are typically sufficient for stable behavior, while 5–10 Newton iterations and Taylor truncation to 2–8 terms provide a reasonable accuracy–efficiency tradeoff.

7.3 Comparison and Advantages of the Proposed Multi-Cloud Design

Beyond the quantitative results, we compare MSSPCA with the closest dual-cloud construction [33] in terms of data owners, cloud organization, assumptions, and cloud inputs.

Both constructions rely on non-colluding cloud servers. Yen's design begins with one data owner and two cloud servers, whereas MSSPCA coordinates multiple independent data providers, jointly forms secret-shared global statistics, and assigns the resulting transformed covariance shares across multiple cloud servers. Under a (t, N) Shamir scheme, fewer than t corrupted participants cannot reconstruct the protected value; in the evaluated four-share, threshold-2 configuration, this bound is one corrupted participant. The distinction of MSSPCA is therefore its multi-provider and multi-cloud organization rather than a different non-collusion assumption.

Table 4: Comparison with the closest dual-cloud PCA construction.

Scheme	Data owners	Cloud setting	Core assumption	Cloud input
Dual-cloud [33]	single	two servers	non-collusion	two shares
MSSPCA (ours)	**multiple**	**multiple servers**	**non-collusion; sub-threshold view**	**assigned shares of C'**

8 Conclusion

In this paper, we proposed **MSSPCA**, a privacy-preserving framework for principal component analysis under a multi-cloud architecture. The design enables multiple data providers to collaboratively perform dimensionality reduction on distributed datasets without exposing a provider's raw data to cloud servers or other providers. Our framework uses threshold secret sharing and distributed computation across non-colluding cloud servers. The number of servers and the reconstruction threshold are configurable within the semi-honest threat model. Experiments on the Wine and Insurance datasets show close numerical agreement with plaintext PCA and report the computation time of the prototype. The results demonstrate the feasibility of coordinating multiple data providers and cloud servers for secret-sharing-based PCA. Overall, MSSPCA provides a multi-provider architecture for distributed PCA computation using threshold secret sharing and a similarity transformation under semi-honest and non-collusion assumptions.

Acknowledgments

This work was supported by the National Science and Technology Council, Taiwan, under Project No. NSTC 114-2634-F-004-002-MBK.

A Component Functionalities and Security Proof Sketches

This appendix provides component functionalities and simulator-based proof sketches for the sub-protocols used in MSSPCA under the semi-honest model of Section 4.2.

A.1 Simulation Paradigm and Proof Strategy

We use standard real/ideal simulation notation under semi-honest corruption. Let Π be a protocol among an active execution set of parties with security parameter κ, and let $\mathcal{A}$ corrupt a set $\mathcal{I}$ whose view remains below the reconstruction threshold. For inputs $\vec{x}$, write $\mathrm{REAL}_{\Pi,\mathcal{A},\mathcal{Z}}(\kappa,\vec{x})$ for the joint distribution of the environment $\mathcal{Z}$'s output together with the view of the corrupted parties in a real execution of Π, where a party's view is the tuple

$$\mathrm{view}_i = \big(x_i,\ r_i,\ m_{i,1}, m_{i,2}, \dots\big)$$

consisting of its input, its random tape r_i, and every message $m_{i,\ell}$ it receives. Let $\mathrm{IDEAL}_{\mathcal{F},\mathcal{S},\mathcal{Z}}(\kappa,\vec{x})$ be the corresponding distribution in the ideal world, where the parties interact only with the ideal functionality $\mathcal{F}$ and a simulator $\mathcal{S}$ produces the corrupted parties' simulated views from the inputs and outputs of the corrupted set. For each component, the proof sketch compares the ideal and real distributions through the relation

$$\mathrm{IDEAL}_{\mathcal{F},\mathcal{S},\mathcal{Z}} \overset{\mathrm{c}}{\approx} \mathrm{REAL}_{\Pi,\mathcal{A},\mathcal{Z}}.$$

The simulation arguments use the privacy of at most $t-1$ Shamir shares and the masking of the opened Beaver differences $\alpha = x - a$ and $\beta = y - b$.

Each simulator reproduces the corrupted participants' inputs, observable messages, and outputs according to the corresponding functionality.

A.2 Beaver-Triple-Based Multiplication

Functionality 1 $\mathcal{F}_{\text{BeaverTriple}}$

Description: $\mathcal{F}_{\text{BeaverTriple}}$ interacts with the participating parties and the simulator $\mathcal{S}$.

1. A designated generator produces a random Beaver triple (a, b, c) with $c = a \cdot b$, and secret-shares each value into shares indexed by the active execution set.
2. The generator distributes the shares $[a]_i, [b]_i, [c]_i$ to every participating party, and sends a copy of the triple to $\mathcal{S}$.
3. Each party computes

$$[\alpha]_i = [x]_i - [a]_i, \qquad [\beta]_i = [y]_i - [b]_i,$$

   broadcasts $[\alpha]_i, [\beta]_i$ to the other parties, and delivers the same messages to $\mathcal{S}$.
4. Upon receiving all broadcasts, each party reconstructs the clear values α and β. The reconstructed values (α, β) are also provided to $\mathcal{S}$.
5. Each party locally computes its share of the product:

$$[z]_i = \alpha\beta + \alpha[b]_i + \beta[a]_i + [c]_i.$$

Theorem 2. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for $\Pi_{BeaverTriple}$ relative to $\mathcal{F}_{BeaverTriple}$ under the stated assumptions.*

Proof. Let $\mathcal{A}$ be a semi-honest adversary corrupting the set $\mathcal{I}$ with $|\mathcal{I}| \leq t-1$. In the real execution each corrupted party $i \in \mathcal{I}$ holds its input shares $([x]_i, [y]_i)$ and triple shares $([a]_i, [b]_i, [c]_i)$, computes $([\alpha]_i, [\beta]_i) = ([x]_i - [a]_i, [y]_i - [b]_i)$, broadcasts them, receives the broadcasts $\{[\alpha]_j, [\beta]_j\}_{j \notin \mathcal{I}}$ from the honest parties, reconstructs (α, β), and outputs $[z]_i = \alpha\beta + \alpha[b]_i + \beta[a]_i + [c]_i$. Thus

$$\text{view}_{\mathcal{A}}^{\text{real}} = \Big(\{[x]_i, [y]_i, [a]_i, [b]_i, [c]_i\}_{i \in \mathcal{I}},\ \{[\alpha]_j, [\beta]_j\}_{j \notin \mathcal{I}}\Big),$$

from which all other quantities are computed deterministically.

We construct an explicit simulator $\mathcal{S}$ that, given only the corrupted inputs $\{[x]_i, [y]_i\}_{i \in \mathcal{I}}$ and the corrupted output shares $\{[z]_i\}_{i \in \mathcal{I}}$ returned by $\mathcal{F}_{\text{BeaverTriple}}$, generates a view distributed identically to $\text{view}_{\mathcal{A}}^{\text{real}}$:

(1) *Triple shares.* For each $i \in \mathcal{I}$, sample $[a]_i, [b]_i \xleftarrow{\$} \mathbb{F}$ uniformly. After sampling the opened values below, set $[c]_i = [z]_i - \alpha\beta - \alpha[b]_i - \beta[a]_i$ so that the simulated view is consistent with the corrupted output share.

(2) *Opened values.* Sample $\alpha, \beta \xleftarrow{\$} \mathbb{F}$ uniformly.

(3) *Honest broadcasts.* Using the corrupted differences $[\alpha]_i = [x]_i - [a]_i$ (resp. $[\beta]_i$) together with the target opened value α (resp. β), sample a degree-$(t-1)$ sharing polynomial uniformly subject to those constraints and define every honest broadcast as an evaluation of that polynomial.

Indistinguishability. Any set of at most $t-1$ Shamir shares is independent of the shared secret, while $\alpha = x - a$ and $\beta = y - b$ are masked by the random values a and b. Conditioned on these opened values, the sampled sharing polynomials generate broadcasts with the same distribution as in the real execution. Defining the corrupted c-shares from the prescribed output shares makes the multiplication relation consistent, so the simulated and real views are indistinguishable. □

Remark. The two instantiations used in the main text—DP-side multiplication and CS-side multiplication—are symmetric specializations of $\Pi_{\text{BeaverTriple}}$. The same simulator argument applies in both cases, with only the role of the designated triple generator changed.

A.3 Covariance, Obfuscation, and Share Distribution

Functionality 2 $\mathcal{F}_{\text{SSCov}}$

Description: $\mathcal{F}_{\text{SSCov}}$ interacts with the data providers and the simulator $\mathcal{S}$.

1. Each DP_j locally computes its statistics (R_j, v_j, n_j) from its private dataset D_j.
2. Upon computing the statistics, DP_j secret-shares R_j and v_j into N shares, and sends $([R_j]_i, [v_j]_i, n_j)$ to every DP_i and the simulator $\mathcal{S}$.
3. Upon receiving all shares, DP_i aggregates them into $[R]_i$, $[v]_i$, and n, then computes its share $[C]_i$ of the global covariance matrix.
4. Upon computing $[C]_i$, DP_i sends this share to the simulator $\mathcal{S}$.

Theorem 3. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for Π_{SSCov} relative to $\mathcal{F}_{SSCov}$ under the stated assumptions.*

Proof. Consider a semi-honest adversary $\mathcal{A}$ corrupting fewer than t data providers. In the real execution, each corrupted DP_j computes local statistics (R_j, v_j, n_j) from its dataset D_j, generates secret shares $([R_j]_i, [v_j]_i)$, sends them together with n_j, and then computes $[C]_i$ from the aggregated shares.

The view of $\mathcal{A}$ contains the corrupted datasets and statistics, the statistic shares received from honest providers, the public counts n_j, the aggregate shares, and $[C]_i$. The simulator forwards corrupted statistics and samples the shares delivered by each honest provider according to the Shamir sharing distribution. It then computes aggregate and output shares by the protocol's deterministic rule. With fewer than t corrupted data providers, the simulated view is indistinguishable from the real view. □

Functionality 3 $\mathcal{F}_{\text{ObfCov}}$

Description: $\mathcal{F}_{\text{ObfCov}}$ interacts with the data providers and the simulator $\mathcal{S}$.

1. Each DP_i holds its local share $[C]_i$ of the global covariance matrix C.
2. Each DP_j locally generates a random invertible matrix $P_j \in \mathbb{R}^{d\times d}$, broadcasts P_j to all other providers, and sends P_j to the simulator $\mathcal{S}$.
3. All DP_j receive the set $\{P_1, P_2, \ldots, P_N\}$ and compute the global obfuscation matrix:

$$P = P_1 P_2 \ldots P_N.$$

4. Each DP_i locally computes the inverse P^{-1} and performs the similarity transformation:

$$[C']_i = P^{-1}[C]_i P.$$

5. Each DP_i sends $[C']_i$ to the simulator $\mathcal{S}$.

Theorem 4. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for Π_{ObfCov} relative to $\mathcal{F}_{ObfCov}$ under the stated assumptions.*

Proof. Consider a semi-honest adversary $\mathcal{A}$ corrupting a set $\mathcal{I}$ of data providers with $|\mathcal{I}| \leq t-1$. In the real execution, each corrupted DP_j, $j \in \mathcal{I}$, holds its covariance share $[C]_j$, samples its own invertible matrix P_j according to the protocol's matrix-generation procedure, receives the matrices $\{P_\ell\}_{\ell \notin \mathcal{I}}$ broadcast by the honest providers, forms $P = P_1 P_2 \cdots P_N$, and locally derives $[C']_j = P^{-1}[C]_j P$. Hence

$$\mathrm{view}_{\mathcal{A}}^{\mathrm{real}} = \Big(\{[C]_j\}_{j \in \mathcal{I}},\ \{P_\ell\}_{\ell \notin \mathcal{I}},\ \{[C']_j\}_{j \in \mathcal{I}} \Big),$$

in which P and every $[C']_j$ are deterministic functions of the listed quantities.

Given the corrupted inputs and outputs, the simulator samples each honest provider's matrix according to the protocol and presents the samples as the honest broadcasts. It then applies the same deterministic transformation to the simulated shares. Because fewer than t covariance shares are available to the adversary, the simulated view is indistinguishable from the real view under Shamir secret sharing. □

Functionality 4 $\mathcal{F}_{\mathrm{ShareAssign}}$

Description: $\mathcal{F}_{\mathrm{ShareAssign}}$ interacts with the data providers, cloud servers, and the simulator $\mathcal{S}$.

1. Each DP_i holds its local obfuscated share $[C']_i$.
2. $\mathcal{F}_{\mathrm{ShareAssign}}$ labels each DP_j in $\mathcal{P}$ with a unique index $p \in \{0, 1, \ldots, |\mathcal{P}| - 1\}$.
3. Let the cloud servers be $\{\mathrm{CS}_0, \mathrm{CS}_1, \ldots, \mathrm{CS}_{s-1}\}$.
4. For each DP_j with index p, compute $r = p \bmod s$ to determine the assigned cloud server CS_r.
5. Each provider holding coordinate-i share $[C']_i$ transmits that share to the assigned cloud server CS_r; the Shamir coordinate remains i after routing.
6. Each CS_r forwards the received $[C']_i$ shares to the simulator $\mathcal{S}$.

Theorem 5. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for $\Pi_{ShareAssign}$ relative to $\mathcal{F}_{ShareAssign}$ under the stated assumptions.*

Proof. Consider a semi-honest adversary $\mathcal{A}$ corrupting a subset of data providers or cloud servers, with fewer than t corrupted participants. Its view consists of the public parameters $(s, \mathcal{P})$, the deterministic mapping $p \mapsto p \bmod s$, and the shares held or received by corrupted participants. The simulator reproduces the public mapping and the corresponding corrupted shares. Since the transmitted values remain Shamir shares and the corruption threshold is not reached, the simulated view is indistinguishable from the real view. □

A.4 Local Approximation, Eigenpair Computation, and Projection

Functionality 5 $\mathcal{F}_{\mathrm{SSTay}}$

Description: $\mathcal{F}_{\mathrm{SSTay}}$ interacts with the cloud servers and the simulator $\mathcal{S}$.

1. Each CS_i holds its local input share $[x_0]_i$ and receives the public parameters B and o.
2. Upon receiving B and o, CS_i locally computes

$$[x_0']_i = \mathrm{TaylorApprox}([x_0]_i, B, o),$$

   where the approximation is centered at $x = B/2 + 1$ and truncated at order o.
3. Upon computing $[x_0']_i$, CS_i sends this share to the simulator $\mathcal{S}$.

Theorem 6. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for Π_{SSTay} relative to $\mathcal{F}_{SSTay}$ under the stated assumptions.*

Proof. Consider a semi-honest adversary $\mathcal{A}$ corrupting a subset of cloud servers. A corrupted server's view consists of its input share $[x_0]_i$, the public parameters (B, o), and its output share $[x_0']_i$. The simulator receives the same corrupted input and public parameters from $\mathcal{F}_{\mathrm{SSTay}}$, applies the deterministic Taylor approximation specified by the protocol, and outputs the resulting share. The simulated view is therefore indistinguishable from the real view. □

Functionality 6 $\mathcal{F}_{\mathrm{SSIS}}$

Description: $\mathcal{F}_{\mathrm{SSIS}}$ interacts with the cloud servers and the simulator $\mathcal{S}$.

1. Each CS_i holds its local input share $[x_0]_i$ and the initial approximation $[x_0']_i$, and receives the public parameter τ.
2. Upon receiving τ, CS_i locally computes its refined approximation share:

$$[x_0'']_i = \mathrm{NewtonApprox}([x_0]_i, [x_0']_i, \tau).$$

3. Upon computing $[x_0'']_i$, CS_i sends this share to the simulator $\mathcal{S}$.

Theorem 7. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for Π_{SSIS} relative to $\mathcal{F}_{SSIS}$ under the stated assumptions.*

Proof. Let $\mathcal{A}$ be a semi-honest adversary corrupting a subset of cloud servers. In the real execution, each corrupted server holds $[x_0]_i$ and $[x_0']_i$, receives the public parameter τ, and locally computes

$$[x_0'']_i = \mathrm{NewtonApprox}([x_0]_i, [x_0']_i, \tau).$$

The corrupted server's view is $\big([x_0]_i, [x_0']_i, \tau, [x_0'']_i\big)$. Products of shared values in the Newton update are evaluated through the Beaver multiplication functionality, so the only opened values are masked Beaver differences. The simulator receives the corrupted input shares and public parameter τ from $\mathcal{F}_{\mathrm{SSIS}}$, applies the Newton approximation specified by the protocol, and outputs the resulting share. The simulated view is therefore indistinguishable from the real view. □

Functionality 7 $\mathcal{F}_{\text{SSPM}}$

Description: $\mathcal{F}_{\text{SSPM}}$ interacts with the cloud servers, the sub-functionalities $\mathcal{F}_{\text{SSTay}}$ and $\mathcal{F}_{\text{SSIS}}$, and the simulator $\mathcal{S}$.

1. Each CS_i holds its obfuscated covariance matrix share $[C']_i$.
2. Upon initialization, CS_i obtains the public initial vector $v^{(0)}$ and the public parameters l_p and τ.
3. CS_i uses $[C']_i$ and $v^{(0)}$ to perform l_p rounds of iterative updates.
4. CS_i invokes $\mathcal{F}_{\text{SSTay}}$ and then $\mathcal{F}_{\text{SSIS}}$ to obtain a normalization factor for the current vector share.
5. Upon completing l_p rounds, CS_i outputs its transformed eigenvector share $[v'_j]_i$ and computes the corresponding eigenvalue share $[\lambda_j]_i$.
6. Upon generating the output, CS_i sends $([v'_j]_i, [\lambda_j]_i)$ to the simulator $\mathcal{S}$.

Theorem 8. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for Π_{SSPM} relative to $\mathcal{F}_{SSPM}$ under the stated assumptions.*

Proof. Consider a semi-honest adversary $\mathcal{A}$ corrupting a subset of cloud servers. Each corrupted server observes its covariance share $[C']_i$, the public initialization parameters, its intermediate local shares, and the output shares $([v'_j]_i, [\lambda_j]_i)$. Matrix-vector products and inner products over shared operands are evaluated through Beaver multiplication whenever both operands are secret-shared. The simulator receives the corrupted input and output shares from $\mathcal{F}_{\text{SSPM}}$ and reproduces the deterministic protocol execution, including the calls to $\mathcal{F}_{\text{SSTay}}$ and $\mathcal{F}_{\text{SSIS}}$. Hence the simulated view is indistinguishable from the real view. □

Functionality 8 $\mathcal{F}_{\text{SSES}}$

Description: $\mathcal{F}_{\text{SSES}}$ interacts with the cloud servers and the simulator $\mathcal{S}$.

1. Each CS_i holds its local obfuscated covariance share $[C']_i$.
2. Upon initialization, CS_i receives the transformed eigenvector shares $[v'_j]_i$ and eigenvalue shares $[\lambda_j]_i$.
3. For each $j = 1$ to $k-1$, CS_i updates

$$[C'_{j+1}]_i = [C'_j]_i - [\lambda_j]_i \cdot [v'_j]_i \cdot [v'_j]_i^{\top}.$$

4. Upon completing all iterations, CS_i obtains the final shifted covariance share $[C'_k]_i$.
5. Upon computing $[C'_k]_i$, CS_i sends the result to the simulator $\mathcal{S}$.

Theorem 9. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for Π_{SSES} relative to $\mathcal{F}_{SSES}$ under the stated assumptions.*

Proof. Consider a semi-honest adversary $\mathcal{A}$ corrupting a subset of cloud servers. A corrupted server observes its covariance and eigenpair shares together with the resulting shifted covariance share $[C'_k]_i$. The products in the eigenshift update are evaluated through Beaver multiplication over the active cloud-server set. The simulator receives the same

corrupted input and output shares from $\mathcal{F}_{\mathrm{SSES}}$ and reproduces the deterministic eigenshift updates. The simulated view is therefore indistinguishable from the real view. □

Functionality 9 $\mathcal{F}_{\mathrm{DPN}}$

Description: $\mathcal{F}_{\mathrm{DPN}}$ interacts with the data providers and the simulator $\mathcal{S}$.

1. Each DP_j holds its local dataset D_j.
2. Upon receiving threshold reconstruction sets of eigenvalue shares $\{[\lambda_k]_i\}_{k=1}^{l_E}$, transformed eigenvector shares $\{[v'_k]_i\}_{k=1}^{l_E}$, and transformation matrix P, DP_j reconstructs the transformed eigenpairs, restores the original eigenvectors via $v_k \leftarrow Pv'_k$, and constructs the projection matrix $V \in \mathbb{R}^{d\times l_E}$.
3. DP_j computes the projected dataset:

$$X'_j = D_j V.$$

4. Upon computing X'_j, DP_j sends the result to the data consumer and to the simulator $\mathcal{S}$.
5. The data consumer forms the full reduced matrix by concatenation:

$$X' = \begin{pmatrix} X'_1 \\ \vdots \\ X'_N \end{pmatrix}.$$

Theorem 10. *For semi-honest adversaries corrupting at most $t-1$ parties, the simulator described below reproduces the corrupted parties' view for Π_{DPN} relative to $\mathcal{F}_{DPN}$ under the stated assumptions.*

Proof. Consider a semi-honest adversary $\mathcal{A}$ corrupting a subset of data providers. Each corrupted DP_j observes its own dataset D_j, the received eigenpair shares, the transformation matrix P, and its projected output X'_j. The simulator receives the same corrupted inputs and outputs from $\mathcal{F}_{\mathrm{DPN}}$, reconstructs the eigenpairs as specified by the protocol, and computes $X'_j = D_j V$. The simulated view is therefore indistinguishable from the real view. □

References

[1] Hussien AbdelRaouf, Mahmoud Abouyoussef, Mostafa M Fouda, and Mohamed I Ibrahem. Empowering ai-driven healthcare with secure, decentralized, and privacy-enhancing adaptive intelligence. *IEEE Internet of Things Journal*, 2025.

[2] Abbas Acar, Hidayet Aksu, A Selcuk Uluagac, and Mauro Conti. A survey on homomorphic encryption schemes: Theory and implementation. *ACM Computing Surveys (Csur)*, 51(4):1–35, 2018.

[3] Olakunle Abayomi Ajala, Chuka Anthony Arinze, Onyeka Chrisanctus Ofodile, Chinwe Chinazo Okoye, and Obinna Donald Daraojimba. Reviewing advancements in privacy-enhancing technologies for big data analytics in an era of increased surveillance. *World Journal of Advanced Engineering Technology and Sciences*, 11(1):294–300, 2024.

[4] Inayat Ali, Eraj Khan, and Sonia Sabir. Privacy-preserving data aggregation in resource-constrained sensor nodes in internet of things: A review. *Future Computing and Informatics Journal*, 3(1):41–50, 2018.

[5] Jinnat Ara, Moumita Roy, and Samia Hossain Swarnali. Differential privacy and federated learning for secure predictive modeling in healthcare finance. *International Journal of Research and Innovation in Applied Science*, 10(9):745–759, 2025.

[6] Donald Beaver. Efficient multiparty protocols using circuit randomization. In *Advances in Cryptology—CRYPTO'91: Proceedings 11*, pages 420–432. Springer, 1992.

[7] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, page 1175–1191, New York, NY, USA, 2017. Association for Computing Machinery.

[8] Paulo Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis. Wine Quality. UCI Machine Learning Repository, 2009. DOI: https://doi.org/10.24432/C56S3T.

[9] Louis W Ehrlich. A modified newton method for polynomials. *Communications of the ACM*, pages 107–108, 1967.

[10] Xiaoyu Fan, Guosai Wang, Kun Chen, Xu He, and Wei Xu. Ppca: Privacy-preserving principal component analysis using secure multiparty computation (mpc). *arXiv preprint arXiv:2105.07612*, 2021.

[11] Oded Goldreich. Secure multi-party computation. *Manuscript. Preliminary version*, 78(110):1–108, 1998.

[12] David Hasselquist, Jacob Wahlman, and Niklas Carlsson. Pet-exchange: A privacy enhanced trading exchange using homomorphic encryption. In *2023 20th Annual International Conference on Privacy, Security and Trust (PST)*, pages 1–12. IEEE, 2023.

[13] Johannes Heurix, Peter Zimmermann, Thomas Neubauer, and Stefan Fenz. A taxonomy for privacy enhancing technologies. *Computers & Security*, 53:1–17, 2015.

[14] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.

[15] Wayne Jansen and Timothy Grance. Sp 800-144. guidelines on security and privacy in public cloud computing. Technical report, National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA, 2011.

[16] Ian T Jolliffe. *Principal component analysis for special types of data*. Springer, 2002.

[17] Nileshkumar Kakade and Utpalkumar Patel. Secure secret sharing using homomorphic encryption. In *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–7, 2020.

[18] RP Kanwal and KC Liu. A taylor expansion approach for solving integral equations. *International Journal of Mathematical Education in Science and Technology*, 20(3):411–414, 1989.

[19] Saloni Kwatra and Vicenç Torra. Data reconstruction attack against principal component analysis. In *International Symposium on Security and Privacy in Social Networks and Big Data*, pages 79–92. Springer, 2023.

[20] Hoon Wei Lim, Geong Sen Poh, Jia Xu, and Varsha Chittawar. PrivateLink : Privacy-preserving integration and sharing of datasets. *IEEE Transactions on Information Forensics and Security*, 15:564–577, 2020.

[21] Xinbo Liu, Yaping Lin, Qin Liu, and Xin Yao. A privacy-preserving principal component analysis outsourcing framework. In *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pages 1354–1359. IEEE, 2018.

[22] Yingting Liu, Chaochao Chen, Longfei Zheng, Li Wang, Jun Zhou, Guiquan Liu, and Shuang Yang. Privacy preserving pca for multiparty modeling. *arXiv preprint arXiv:2002.02091*, 2020.

[23] Xirong Ma, Chuan Ma, Yali Jiang, and Chunpeng Ge. Improved privacy-preserving pca using optimized homomorphic matrix multiplication. *Computers & Security*, 138:103658, 2024.

[24] Arvind Narayanan and Vitaly Shmatikov. Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 111–125, 2008.

[25] Valeria Nikolaenko, Stratis Ioannidis, Udi Weinsberg, Marc Joye, Nina Taft, and Dan Boneh. Privacy-preserving matrix factorization. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, CCS '13, page 801–812, New York, NY, USA, 2013. Association for Computing Machinery.

[26] Daniel E. O'Leary. Artificial intelligence and big data. *IEEE Intelligent Systems*, 28(2):96–99, 2013.

[27] Samanvaya Panda. Principal component analysis using ckks homomorphic scheme. In *Cyber Security Cryptography and Machine Learning: 5th International Symposium, CSCML 2021, Be'er Sheva, Israel, July 8–9, 2021, Proceedings 5*, pages 52–70. Springer, 2021.

[28] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[29] Vinoth Punniyamoorthy, Ashok Gadi Parthi, Mayilsamy Palanigounder, Ravi Kiran Kodali, Bikesh Kumar, and Kabilan Kannan. A privacy-preserving cloud architecture for distributed machine learning at scale. *arXiv preprint arXiv:2512.10341*, 2025.

[30] Adi Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, November 1979.

[31] Peter van der Putten and Maarten van Someren. CoIL Challenge 2000: The Insurance Company Case. Technical Report 2000-09, Leiden Institute of Advanced Computer Science, Leiden, The Netherlands, June 2000. Published by Sentient Machine Research, Amsterdam; also Leiden Institute of Advanced Computer Science Technical Report 2000-09, June 22, 2000.

[32] Zhihua Xia, Qi Gu, Lizhi Xiong, and Wenhao Zhou. Privacy-preserving image retrieval based on additive secret sharing. *International Journal of Autonomous and Adaptive Communications Systems*, 17(2):99–126, 2024.

[33] Hsin-Chan Yen. Privacy preserving principal components analysis: A dual-cloud construction based on secret sharing for protecting covariance matrix. Master's thesis, National Taiwan University of Science and Technology, 2024. Available from the National Digital Library of Theses and Dissertations in Taiwan.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 76–88.

Light Image Encryption via Wavelet Transform

Nur Azman Abu[1,2], Rania Mohammed[3] and Mohd Rizuan Baharon[3]

[1] Faculty of AI and Cyber Security, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia nura@utem.edu.my

[2] Malaysia Cryptology Technology and Management Centre Universiti Putra Malaysia, 43400 Serdang Selangor, Malaysia norazman@upm.edu.my

[3] Faculty of ICT, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia mohd.rizuan@utem.edu.my

Abstract. Traditional full-image encryption imposes significant computational overhead, despite the majority of visual data within an image being non-sensitive public information. Typically, only specific features, such as diagnostic markers in medical imaging or identifiable biometric details in personal photography, require strict confidentiality. The present paper introduces a novel lightweight encryption technique designed to selectively secure only the sensitive components of an image. By applying the Advanced Encryption Standard(AES) to a targeted subset of image wavelet transform coefficients, the proposed method encrypts less than 2% of the total data. Consequently,the encrypted image remains visually reconstructable and accessible for general viewing, while the original high-fidelity image requires decryption for full recovery. Finally, a comprehensive set of quality metrics is presented to evaluate the impact of such lightweight encryption on overall image fidelity, demonstrating a highly efficientbalance between data security and computational cost.

Keywords: Image Encryption · DWT Encryption · JPEG2000 Encryption

1 Introduction

In the technology world of today, safe data transmission is a basic requirement from the growth reliance on social media and the communication platforms. An exponential expansion of multimedia data including images transmitted over the internet for diverse uses including healthcare, military, e-commerce, and social networking there is a need for private protection from illegal access, modification, or interception. Although secure standard encryption methods are good for text-based data, their intrinsic properties such as great size, high redundancy, and strong correlation among pixels often cause them to betoo heavy for casual multimedia data including images.

Current encryption process is computationally heavy and visually destroying the whole information which render an image object to useless level.

Given a raw Lena image in Figure 1, this paper will focus on encrypting a beauty element as part of the privacy quality that a person might want to protect from public view. These qualities include a skin smoothness upon her shoulder, a beautiful lip, nose and sparkling eyes. A general feature of Lena as a female person here is not to be protected. After a wavelet transform, a set of coefficients will be encrypted to protect these virtues. This paper will present a basic mechanism to lightly encrypt Lena beauty rather than her personal identification. An encrypted image can still be viewed without decryption and

displayed for public consumption. Only an intended receiver who has the shared secret key can decrypt and view the natural beauty of Lena.

Figure 1: A raw Lena image.

2 Literature Review on Image Encryption

An image encryption is a computationally intensive process. An image typically carries redundant information and plain features. In this paper, a focus should be on a sensitive private to be protected. Only a certain sensitive quality should be encrypted but the whole image.

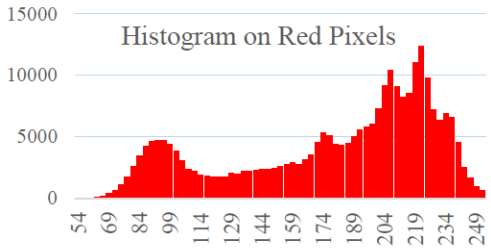


Figure 2a: A histogram of red pixel values from a raw Lena image.

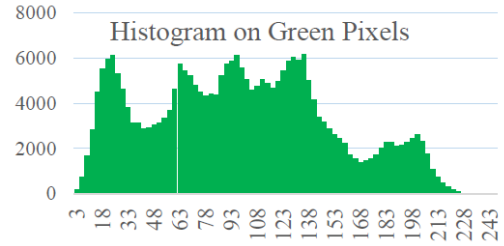


Figure 2b: A histogram of green pixel values from a raw Lena image.

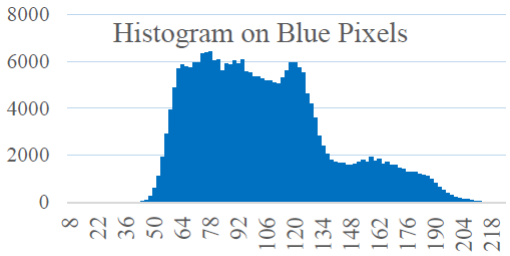


Figure 2c: A histogram of blue pixel values from a raw Lena image.

An image can be represented by RGB layers. The RGB is a popular colour representation with three components that scale a colour image. These RGB distributions from Lena image are sporadically depicted in Figures 2abc. An ideal distribution to operate on is a normal distribution.

There is one significant difference between the encryption of visual data and the encryption of data encryption is classically applied to (e.g. text data, web documents): visual data is usually much larger, especially in the case of image/video encryption. Given this fact together with possible timing constraints or real-time requirements it becomes clear that speed might be an important issue [14].

In a video conference application if mobile devices are involved, it is a challenging task to deliver due to processing power to provide full real-time encryption. A real-time encryption while using classical full encryption with a standard cryptographic cipher may be hard to fulfil due to simultaneous requests from many clients.

Based on recent literatures, 8 contemporary image encryption approaches have been collected and tabulated in Table 1. A visual light weight image encryption for a specific application is called for here.

Table 1: Selected recent image encryption project for a specific application.

Paper	Approach	Strengths	Limitations
Nonlinear Permutation Framework [5].	3 chaotic maps (Zaslavsky, Chebyshev, logistic)	One-iteration processing; excellent entropy near ideal 8	Lightweight focus may sacrifice some security depth
Optical Chaos Multi-Image [16]	Electro-optical feedback + optical injection chaos	Handles multiple images simultaneously; full-size support	Hardware complexity; requires optical systems
AES on Edge Devices [1]	AES-128 on Jetson edge devices	Real-time performance (0.0268s on Xavier NX); agricultural focus	Limited to edge/IoT scenarios
Quantum-Resistant Lightweight [8]	ASCON, TinyJAMBU, SPARKLE variants	Post-quantum resistant; NIST-standardized	Needs more testing across image types

Paper	Approach	Strengths	Limitations
Random Strip Peeling [6]	1D Tent map + Logistic map	Lightweight; efficient for IoT	Slightly lower entropy than ideal 8
Chaotic + AES-GCM [4]	Hyperbolic tan + Renyi maps + AES-256	Authenticated encryption; dynamic IV per image	Newer, less extensively tested
Deep Learning + AES [9]	Deep autoencoder + AES encryption	Combines compression with security; emerging paradigm	Low imperceptibility under constrained bandwidth
Selective MSB and LSB [7]	Only MSBs and sensitive bits of each pixel	Good visual obfuscation and selection can be adjusted for higher security or faster performance	Security relies on bit selection; may leak partial information; potentially vulnerable if patterns are exploited;

İnce and et. al proposes lightweight image encryption via a nonlinear permutation-based image encryption framework that integrates multiple chaotic maps to enhance randomness and security while keeping computation low [5].

Xia and et. al proposes a hardware-oriented image encryption scheme that leverages optical chaos generated by electro-optical feedback and optical injection [16]. Unlike traditional digital chaos methods, it enables parallel encryption of multiple full-size images, achieving high efficiency and strong security. However, its reliance on complex optical hardware makes it less practical for lightweight or purely software-based applications.

Alexan et. al proposes a new image encryption algorithm that combines a chaotic system to generate pseudo-random sequences and a nonlinear transformation to increase complexity and security [2]. The goal is to achieve high security, fast performance, and strong resistance to attacks. The algorithm is fast and suitable for real-time applications.

Kaya and İnce investigates the use of NIST-standardized lightweight cryptographic algorithms (ASCON, TinyJAMBU, SPARKLE) for image encryption, emphasizing quantum resistance and efficiency [8]. The results show that these algorithms are fast, secure, and suitable for constrained devices, but require further validation and adaptation for image-specific security challenges.

İnce and et. al introduces a lightweight image encryption algorithm using a novel Random Strip Peeling (RSP) technique combined with Tent and Logistic chaotic maps [6]. It performs color-plane-based permutation and diffusion, achieving strong efficiency and near-ideal randomness. The method is well-suited for IoT environments, though it sacrifices a small degree of security (entropy slightly below 8) due to its lightweight design.

Chothe and et. al proposes a hybrid image encryption scheme that integrates chaotic maps (hyperbolic tangent and Rényi) with AES-GCM (AES-256) [4]. The chaos component enhances randomness and key generation, while AES-GCM provides standardized, authenticated encryption with integrity protection. A dynamic initial value(IV) per image further strengthens security. Although highly promising, the approach is new and not yet widely validated.

Madani and Bourennane shows that CNN autoencoders can simultaneously compress and encrypt images, offering a practical approach for secure image transmission in constrained environments [9]. It's an emerging paradigm bridging deep learning and cryptography, but for high-security applications, traditional cryptography is still recommended since its computational cost is higher than standard lightweight ciphers, but lower than full homomorphic encryption or hybrid schemes.

Jacaman and Farajallah introduces a selective image encryption framework for colour imagery, emphasizing computational efficiency without compromising practical security [7]. This selective image encryption framework targets colour images by encrypting only the most informative and perceptually critical bits of each pixel — notably the most significant bits (MSBs) and sensitive bits — while leaving less-critical bits in plaintext to reduce computational overhead.

An AES cipher is preferred of chaos cipher. Lightweight has been the main focus of this project. This selective image encryption approach targets colour images by encrypting only the most informative and perceptually critical bits at DWT parameters.

3 Wavelet Transform and Its Role in Encryption

Especially for image data, Discrete Wavelet Transform (DWT) is a special ability of contemporary encryption methods to handle and expose signals across several sizes and resolutions. Originally derived from signal processing, unlike conventional transforms like Fourier, which rely simply on frequency and lose spatial context, a wavelet transform is a logical tool for separating a signal—such as an image—into many frequencies keeping spatial information. DWT permits selective alteration of visual characteristics at different resolutions, therefore keeping a perfect mix of security, efficiency, and adaptability, this dual trait makes DWT very effective for encryption.

DWT separates a picture into roughly low-frequency and high-frequency bits. The approximation values capture the general image form, but the detail values better textures and edges. This split offers a mechanism to selectively encrypt important portions of the image, therefore optimising computational speed by concentrating encrypt efforts where most effective—that is, by safeguarding low-frequency data while leaving high-frequency bits unaltered.

Another great advantage in DWT encryption is interoperability with compressed and lightweight encryption systems—which are vitally essential for real-time applications and resource-limited environments like IoT devices. Compressive sensing (CS), for instance, conveniently links with DWT by combining encryption and sampling into one phase [14].

The JPEG 2000 image coding standard [13] is based on a scheme originally proposed by Taubman and known as Embedded Block Coding with Optimised Truncation (EBCOT) in [12]. This light image encryption on block of DWT coefficient should seamlessly in JPEG2000 prior to lossless compression.

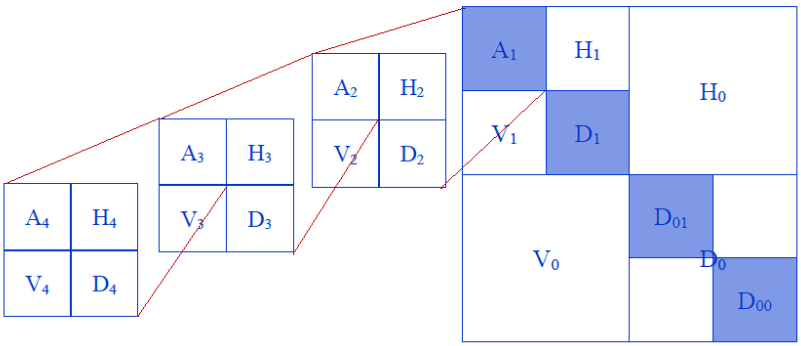


Figure 3: A selection of coefficients a long a diagonal path within a wavelet transform.

4 A Selection of DWT Coefficients

Initially, an image object A will go through discreet wavelet transform into 4 quadrants (A_0, H_0, V_0, D_0). Then first, A_0 will be transformed consecutively into (A_1, H_1, V_1, D_1), Second, A_1 will be transformed consecutively into (A_2, H_2, V_2, D_2) and so on as depicted in Figure 3.

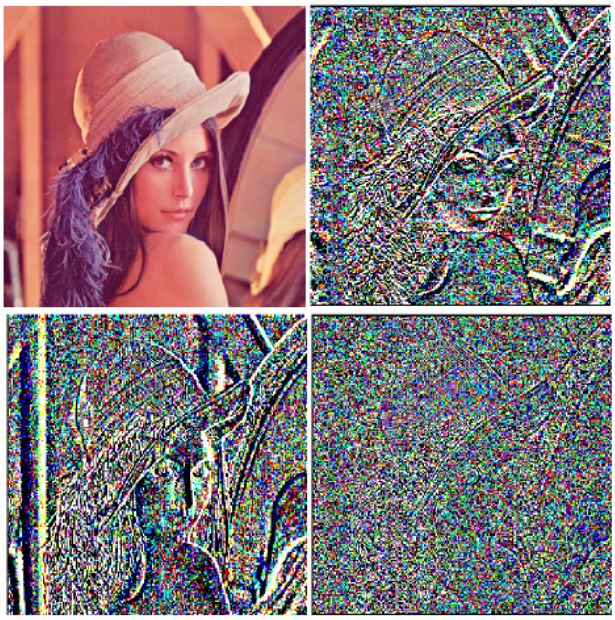


Figure 4: An initial tiling from a DWT Daubechies 1.

An initial 4 quadrants (A_0, H_0, V_0, D_0) is presented in Figure 4. Then first quadrant A_0 from Lena image is sporadically distributed similar to Figures 2abc. However, the rest of the quadrants $H_0, V_0 and D_0$ follow closer toward an ideal normal distribution. They can be easily compressed which is not a good arena for encryption.

Figure 5: A level 4 tiling from a DWT Daubechies 1 with 64 by 64 coefficients in a blue box.

An original image is 512x512x3. High frequency signal coefficients D_{00} and D_{01} carry the size of 128x128x3. Mid frequency signal coefficients D_1 and A_1 carry the size of 128x128x3. Taking into another DWT on A_1, low frequency signal coefficients D_2 and A_2 carry the size of 64x64x3. An image can be typically transformed from 512x512x3 level 0 to 32x32x3 level 4 coefficients. All coefficients can be presented as an image in Figure 5.

5 Balancing Security and Computational Efficiency

Most of the digital era data security and telecommunication architecture define them by their two goals: safety and computational economy. This difficult balance calls for security systems to be robust enough to resist contemporary attacks while also maintaining either efficient processor speed, ram use, or power consumption. Depending on that equilibrium, efficient adoption of technologies for encryption, secured means of communication, and privacy protection, a light image encryption is introduced in this paper.

In streaming applications such as video encryption, the rapid encryption and decryption of image data is crucial for maintaining seamless operation, making real-time processing and latency issues paramount. Reaching the best mix between security, speed, and overall performance calls for careful evaluation of the particular application needs. This light image encryption technique focuses on quality and beauty protection only.

6 Quality Measure on Image Encryption

The quality measurement is an important aspect in image encryption. The conventional quality measurements such as reconstruction error, Means Square Error (MSE), Peak Signal to Noise Ratio (PSNR) are measured to analyze the performance of image encryption. The reconstruction error is calculated by obtaining the differences between image reconstruction $g(i, j, k)$ and original image $f(i, j, k)$. An absolute image reconstruction error is defined as follows [10]:

$$ABE = \frac{1}{3MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sum_{k=0}^{R-1} |g(i,j,k) - f(i,j,k)|. \tag{1}$$

where the original image size is $M \times N$ and the third index refers to the three colours of RGB image. Another convenient measurement is the Means Squared Error (MSE) which calculates the average of the square error. The MSE is defined as follows [15]:

$$MSE = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sum_{k=0}^{2} \|g(i,j,k) - f(i,j,k)\|^2. \tag{2}$$

A Peak Signal to Noise Ratio(PSNR) is calculated to measure the quality of image reconstruction. A higher PSNR means that the image reconstruction is closer to the original image. The PSNR is defined as follows [11]:

$$PSNR = 20 \log_{10} \left(\frac{Max_i}{\sqrt{MSE}} \right) = 10 \log_{10} \left(\frac{255}{MSE} \right)^2 \tag{3}$$

where Max_i is the maximum possible pixel value of the image. The image samples are represented by using 8-bits per pixel per colour channel which means that the maximum intensity of sample image is 255. Another image quality measurement is Structural Similarity (SSIM) index. The SSIM is defined by three components as follows:

$$SSIM(x,y) = [l(x,y)]^{\alpha} \cdot [c(x,y)]^{\beta} \cdot [s(x,y)]^{\gamma} \tag{4}$$

where exponents $\alpha > 0, \beta > 0, \gamma > 0$, are parameters to adjust the relative importance of the three components luminance l, contrast c and structure s. A luminance comparison is:

$$l(x,y) = \frac{2\mu_x^2 \mu_y^2 + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{5}$$

A contrast comparison is:

$$c(x,y) = \frac{2\sigma_x \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{6}$$

A structure element is:

$$s(x,y) = \frac{2\sigma_{xy} + C_3}{\sigma_x + \sigma_y + C_3}, \tag{7}$$

where C_1 is a small quantity introduced for numerical stability, as are C_2 and C_3 in the equations(5 – 7). The three quantities are given as functions of the dynamic range of the pixel values L ($L = 255$ for 8 bits/pixel gray-scale images) and two scalar constants $K_1 \ll 1$ and $K_2 \ll 1$ (traditionally set to 0.01 and 0.03, respectively): $C_1 = (K_1 L)^2$, $C_2 = (K_2 L)^2$, $C_{13} = C_2/2$. Contrast is represented through the use of standard deviation σ. The detail description of SSIM is given in Bakurov and et. al [3]. Recently, SSIM has been more efficient than traditional image quality measurement. The SSIM index provides improvement in accuracy of the objective quality measurement.

In order to measure the encryption ratio, the total bits are encrypted out of the original image 24-bit with 512x512 pixels. In this project only 64x64x3 bits are really encrypted which is equal to one out of 8x8x8 bits. The encryption ratio is just 0.1953125%.

7 Image Quality Assessment

Traditionally, an image encryption is a method to securely protect storage image data. However, an image is being shared and transmitted at will. A fully encrypted image will only render it useless. Therefore, there is a need for this light encryption method to not significantly alter the images' quality. Using objective criteria for picture quality, such as the Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR), we will compare the decoded images to the original photos and determine their authenticity. In addition, we shall conduct subjective evaluations to verify if the aesthetic appeal remains high. The main objectives are to minimize visual data loss and ensure that the light encrypted images remain identical to the originals while preserving medium visual quality.

Table 2: A level of quality across a global quantizer g under DWT image.

g	ABE(g)	MSE(g)	PSNR(g)	SSIM(g)	Quality
1	0.0752	0.1359	56.7978	0.9999	Highest
2	0.2439	0.4486	51.612	0.9998	High
4	0.5210	1.4415	46.5427	0.9994	Fine
8	1.0132	5.0659	41.0843	0.9979	Medium
16	1.5056	14.7949	36.4297	0.9938	Acceptable
32	2.1871	33.5898	32.8687	0.9863	Poor
64	2.9378	68.3987	29.7803	0.9739	Weak

A high-fidelity quality compressed image under DWT, after going through global quantization $g = 2$, MSE should be less than 1 and PSNR above 48. Ideally, using a global quantizer $g = 16$ where MSE is near 16, PSNR = 36 as pointed in Table 2. This quality level is meant for regular user. A low printing quality target after encryption MSE should be above 32 and PSNR below 32. At this level a skin quality beautiful smoothness has been degraded.

We start from a global quantizer g = 2. We want to select a section of DWT coefficients to encrypt down to a below beauty quality level PSNR = 32. The main idea is to protect the visual beauty of a feminine skin but the facial identity. There is no need to go down to such a low degraded quality with a blocking effect.

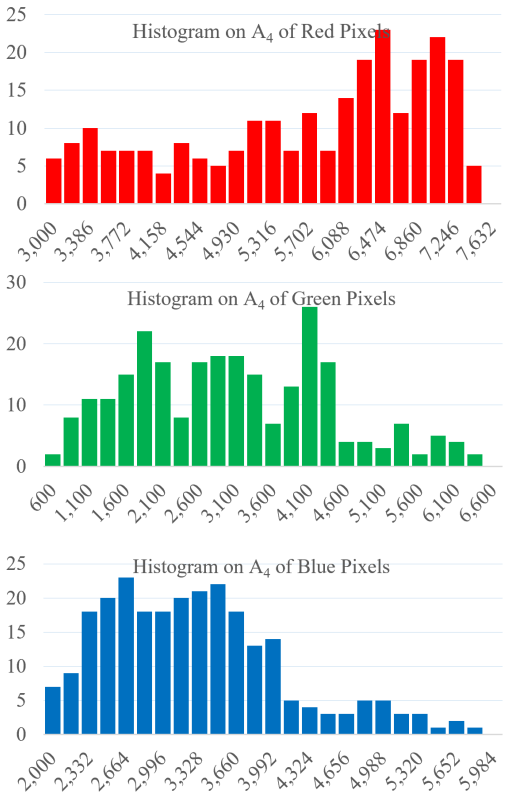


Figure 6: A level 4 DWT A_4 coefficients of RGB layers ranges from 9 to 13-bit scales.

A traditional image encryption may select a region of encryption. Since an encryption is heavy computational process, another approach is called for here. The most significant element in DWT concentrates at the highest level of coefficient transform, namely, level 4 here. This light encryption project will simply start from encrypting level 4 DWT coefficient set namely, (A_4, H_4, V_4, D_4). Each coefficient set consist of 16 by 16 elements. Since these elements range higher than 8-bit values as depicted in Figure 6, they are pruned to be encrypted.

Next, an ith bit from each top left 64 by 64 coefficients shall be encrypted. In the end, there will be only 64x64x3x1 = $3x2^{12}$ bits will be encrypted out of 512x512x3x8 = $3x2^{21}$ bits from the original image. This encryption ratio one out of 2^9 is efficiently small which is less than 0.2% of original high-fidelity image. Full quality measure across i^{th} bit encryption is tabulated in Table 3.

Table 3: A level of quality across i^{th} least significant bit encryption on 64 by 64 DWT coefficients.

i	ABE	MSE	PSNR	SSIM	Quality
1	0.0776	0.1441	56.5438	0.9999	Highest
2	0.0930	0.1728	55.7546	0.9999	High
3	0.1521	0.3054	53.2821	0.9999	Fine
4	0.3004	0.6977	49.6943	0.9998	Medium
5	0.5743	2.2250	44.6574	0.9993	Acceptable
6	1.1336	8.0311	39.0831	0.9974	Poor

i	ABE	MSE	PSNR	SSIM	Quality
7	2.2621	31.9676	33.0837	0.9903	Weak
8	4.4362	128.1921	27.0522	0.9630	Ugly
9	8.6672	484.6258	21.2767	0.8792	Very ugly

In this paper, the 6th LSB are ripped to be encrypted from each top left 64 by 64 coefficient as presented in Figure 5. They are collected via a zigzag scan prior to an encryption process by AES.

Figure 7: On the left is an encrypted Lena and on the right is its enhanced difference from original Lena by a factor of 32.

The primary objective is not to encrypt Lena as a private subject of interest. As shown in Figure 7, this efficient technique will only encrypt Lena beauty which is equivalent to the printing quality of Lena image. In this paper, the 6th LSB has been chosen from 64 by 64 wavelet coefficients as a room to encrypt which is aligned to a medium quality at global quantization parameter $g = 16$. At PSNR $= 39.0831$, Lena beauty has been sufficiently degraded. At higher level, the 7th bit encryption will produce PSNR $= 33.0837$ where a public viewer may not be able to appreciate Lena image in general as a model.

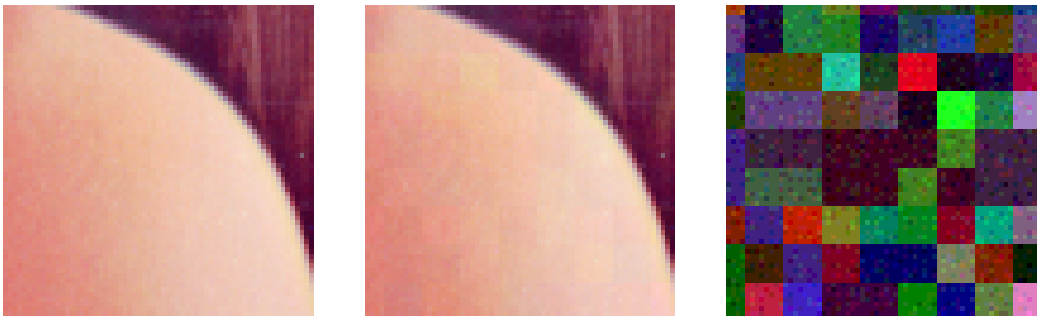


Figure 8: On the left is a zoomed in Lena shoulder, in the middle is after encryption and on the right is its enhanced difference by a factor of 32.

As pointed in Figure 8, the quality of an image here is the beauty of Lena shoulder which has been obscured. An observer or outsider, he or she will not be able to judge on the beauty of Lena shoulder. The objective is not to protect Lena shoulder as a private subject of interest but its beauty.

8 Discussion

In this paper, one bit from a portion of wavelet coefficients is sufficient to encrypt which is aligned to a medium quality image under compression. At higher i^{th} LSB, this technique will introduce higher intentional noise per bit encryption. A watermarking technique may want to choose lower i^{th} LSB to achieve psychovisual invisibility. In a case, there is a glimpse of hope to break this light image encryption, it will certainly open a new arena on attacking a cipher via digital signal processing (DSP). This arena has been longed for in the last several decades due to a vast processing power from DSP devices.

9 Conclusion

Traditional full-image encryption often incurs unnecessary computational costs by securing non-sensitive visual data. To resolve the inefficiencies, a novel selective encryption technique was introduced, applying the Advanced Encryption Standard (AES) to less than 2% of image wavelet transform coefficients. The results demonstrate the lightweight approach successfully secures sensitive visual features while leaving the majority of the image visually reconstructable for general access. Quality metric evaluations validate the proposed method maintains an excellent balance between computational efficiency and image fidelity. While the present study establishes a strong foundation for selective encryption in static images, future work will focus on optimizing coefficient selection algorithms for real-time video streams. By securing only the sensitive components of an image, the proposed light image encryption offers a highly practical, resource-efficient solution for modern print publications.

References

[1] Mohammad Ashik Alahe, Young Chang, James Kemeshi, Kwanghee Won, Xufei Yang, and Lin Wei. Real-Time agricultural image encryption algorithm using AES on edge computing devices. *Computers and Electronics in Agriculture*, 237(Part B):10594, 2025.

[2] Wassim Alexan, Noura H. El Shabasy, Noha Ehab, and Engy Aly Maher. A secure and efficient image encryption scheme based on chaotic systems and nonlinear transformations. *Scientific Reports*, 15:31246, 2025.

[3] Illya Bakurov, Marco Buzzelli, Raimondo Schettini, Mauro Castelli, and Leonardo Vanneschi. Structural similarity index (SSIM) revisited: A data-driven approach. *Expert Systems with Applications*, 189:116087, 2022.

[4] Rupaliben V. Chothe, Sunita P. Ugale, Dinesh M. Chandwadkar, and Shraddha V. Shelke. Authenticated image encryption using robust chaotic maps and enhanced advanced encryption standard. *Indonesian Journal of Electrical Engineering and Computer Science*, 37(3):1543–1554, 2025.

[5] Cemile İnce, Kenan İnce, and Davut Hanbay. A Multilayer Nonlinear Permutation Framework and Its Demonstration in Lightweight Image Encryption. *Entropy*, 26(10):885, 2024.

[6] Kenan İnce, Cemile İnce, and Davut Hanbay. Random strip peeling: A novel lightweight image encryption for iot devices based on colour planes permutation. *CAAI Transactions on Intelligence Technology*, 10(2):529–544, 2025.

[7] Issa Jacaman and Mousa Farajallah. Selective Color Image Encryption Based on MSB and Sensitive Bits. *Journal of Intelligent Communication (JIC)*, 4(2):115–140, 2025.

[8] Muhammed Saadetdin Kaya and Kenan İnce. Using Quantum Resistant Variants of NIST IR 8454 Lightweight Encryption Algorithms in Image Encryption. In *Proceedings of the 2024 Conference, Malatya, Turkiye*. IEEE Xplore, 2024.

[9] Mahdi Madani and El-Bay Bourennane. Visually Image Encryption and Compression Using a CNN-Based Auto Encoder. *International Journal of Computer Networks & Communications (IJCNC)*, 17(2), 2025.

[10] Yuhei Nomura and Hirotaka Hachiya. Frequency-dependent Image Reconstruction Error for Micro Defect Detection. In *Proceedings of the 15th Asian Conference on Machine Learning*, volume 222, pages 1007–1022, 2024.

[11] Umme Sara, Morium Akter, and Mohammad Shorif Uddin. Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study. *Journal of Computer and Communications*, 7(3):8–18, 2019.

[12] David S. Taubman. High performance scalable image compression with EBCOT. *IEEE Transactions on Image Processing*, 9(7):1158–1170, 2000.

[13] David S. Taubman and Michael W. Marcellin. *JPEG2000 — Image Compression Fundamentals, Standards and Practice*. Springer Science+Business Media, Kluwer Academic Publishers, 2002.

[14] Andreas Uhl and Andreas Pommer. *Image and Video Encryption from Digital Rights Management to Secured Personal Communication*. Advances in Information Security. Springer, Boston, 2005.

[15] Paul W. Vos. Rethinking Mean Square Error: Why Information is a Superior Assessment of Estimators. *Statistics Theory*, 2025. arXiv:2412.08475v2.

[16] Yiyang Xia, Xuefang Zhou, Zan Li, and Jianrong Bao. An encryption algorithm for full-size color and grayscale images utilizing heterogeneous optical chaos systems. *Expert Systems with Applications*, 301:130534, 2025.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 89–100.

A Perfect Code Based $(t+1, 2t+1)$-Threshold Non-Perfect Secret Sharing Scheme

Sanjit Bhomick[1], Deepak Kumar Dalai[2] and Haradhan Ghosh[3]

[1] Department of Electronics and Electrical Engineering, Indian Institute of Technology, Guwahati, India sanjitbhowmick@rnd.iitg.ac.in
[2] School of Mathematical Sciences, National Institute of Science Education and Research, An Off-Campus Center of Homi Bhabha National Institute, Bhubaneswar, India deepak@niser.ac.in
[3] Department of Mathematics, Indian Institute of Technology, Kharagpur, India haradhan.ghosh@niser.ac.in

Abstract. Non-perfect secret sharing schemes offer an efficient approach for distributing sensitive information over networks. In this work, we propose a threshold non-perfect scheme derived from perfect error-correcting codes, in which secret reconstruction is achieved through unique bounded-distance decoding. Exploiting the partitioning property of perfect codes, any $t+1$ shares from $2t+1$ shares suffice to recover the secret deterministically. We prove correctness and non-perfect secrecy, and analyze the scheme's information rate. The construction highlights a decoding-centric alternative to classical threshold secret sharing schemes. This secret sharing scheme is based on coding theory, which is of theoretical interest and holds potential for diverse applications due to its ease of analysis, easy implementation and low complexity.

Keywords: Secret sharing · Threshold scheme · Perfect code

1 Introduction

Secret sharing is a fundamental cryptographic primitive that enables a secret to be distributed among a set of participants such that only authorized subsets can reconstruct it, while unauthorized subsets are prevented from fully recovering the secret. Since the seminal works of Shamir and Blakley, threshold secret sharing schemes have played a central role in secure multiparty computation, distributed storage, access control, and fault-tolerant systems [3, 25]. Classical threshold schemes typically aim to achieve perfect secrecy, where any coalition of fewer than the threshold number of participants learns zero information about the secret, while any authorized set can reconstruct it efficiently and uniquely [1, 2, 4, 8, 16, 26, 29]. However, in several practical and combinatorial settings, relaxing perfect secrecy to bounded information leakage leads to more compact or structurally simpler constructions.

Classical threshold schemes, most notably Shamir's polynomial-based construction, rely on algebraic interpolation over finite fields. Although these schemes provide strong information-theoretic guarantees, they may incur nontrivial computational overhead and offer limited flexibility in settings where decoding-based recovery or combinatorial structure is desirable. This has motivated the exploration of alternative constructions based on error-correcting codes, where reconstruction can be naturally interpreted as a decoding process [5, 7, 9, 17, 21–23]. Massey [21] initiated the coding-theoretic approach to secret sharing by relating access structures to minimal codewords, and later extended this framework using dual codes [22]. Ding et al. [11] established sufficient conditions for

determining minimal codewords of linear codes, while Calkavur and Solé [6] proposed a multiple-secret sharing scheme based on bounded-distance decoding. Subsequent works have explored specialized code families, including Golay codes [10] and linear complementary dual (LCD) codes [2, 7, 17, 24], to enhance efficiency, verifiability, and structural flexibility. These schemes, however, do not fully exploit the decoding optimality inherent in perfect error-correcting codes. Non-perfect secret sharing schemes relax perfect secrecy to improve efficiency and flexibility. Kaced [19] introduced almost-perfect secret sharing, allowing negligible information leakage, while Farras et al. [15] studied the efficiency gains of non-perfect schemes via the information ratio. More recently, Yoshida [27] proposed hybrid non-perfect constructions combining algebraic and information-theoretic techniques. These works motivate decoding-based schemes that trade perfect secrecy for compact shares and deterministic reconstruction.

In this paper, we investigate a threshold secret sharing construction based on perfect error-correcting codes. Perfect codes, such as Hamming codes, are characterized by the property that Hamming balls of a fixed radius around codewords partition the entire ambient space, ensuring unique bounded-distance decoding. This feature provides a natural and deterministic mechanism for secret reconstruction: any sufficiently accurate share vector can be decoded unambiguously to a unique codeword. Using this property, we propose a $(t + 1)$-threshold secret sharing scheme derived from perfect t-error correcting codes [10, 14, 20]. Shares are generated via randomization, and reconstruction is achieved through standard perfect decoding once a sufficient number of shares are available, directly mapping the threshold condition to the decoding radius. The resulting construction realizes a ramp secret sharing scheme with compact shares and decoding-based reconstruction, offering a combinatorial alternative to polynomial-based approaches. Restricting to perfect codes is a deliberate design choice motivated by complexity theory. General bounded-distance decoding is computationally hard; it is $\mathcal{NP}$-hard, and in the parameterized setting, $\mathcal{W}[1]$-hard when parameterized by the number of errors t, and $\mathcal{W}[2]$-hard when parameterized by the code dimension k [12, 13]. Perfect codes circumvent these barriers entirely, admitting deterministic polynomial-time decoding (e.g., syndrome decoding for Hamming codes). Thus, perfect codes are not merely convenient but essential for practical, efficient secret reconstruction. The proposed scheme offers a clean combinatorial interpretation of threshold secret sharing through the lens of error-correcting codes, where the access structure is determined by the decoding radius. Its reconstruction mechanism reduces to efficient, deterministic bounded-distance decoding, which simplifies both analysis and implementation relative to algebraic alternatives.

The remainder of this paper is organized as follows. Section 2 introduces perfect codes and summarizes their key properties. Section 3 presents the proposed $(t + 1, 2t + 1)$ threshold scheme and demonstrates its correctness through an illustrative example. Section 4 analyzes the features of the proposed protocol and discusses its limitations. Finally, Section 5 concludes the paper.

2 Background

In this section, we briefly discuss and present notations for perfect error correcting codes and secret sharing.

2.1 Perfect t-Error Correcting Codes

A binary $[n, k, d]$ linear code $\mathcal{C}$ is called a *perfect t-error correcting code* if $d = 2t{+}1$ and the Hamming balls of radius t centered at the codewords of $\mathcal{C}$ partition $\mathbb{F}_2^n$ [14]. Equivalently, every vector in $\mathbb{F}_2^n$ lies within at most t Hamming distance from a *unique* codeword,

achieving equality in the sphere-packing bound

$$|\mathcal{C}| \sum_{i=0}^{t} \binom{n}{i} = 2^n \implies \sum_{i=0}^{t} \binom{n}{i} = 2^{n-k}. \tag{1}$$

Perfectness implies *unique bounded-distance decoding* i.e., for any $y \in \mathbb{F}_2^n$, there exists exactly one $c \in \mathcal{C}$ such that $d(y, c) \leq t$. This deterministic decoding property is well-suited for cryptographic and threshold-based constructions requiring unambiguous reconstruction. Binary perfect linear codes are extremely rare. Apart from trivial repetition codes, the only known families are Hamming codes $[2^m - 1,\ 2^m - 1 - m,\ 3]$ (i.e., $t = 1$) and the binary Golay code $[23,\ 12,\ 7]$ (i.e., $t = 3$). Although their parameter choices are limited, perfect codes provide optimal packing and sharp decoding thresholds, making them attractive for constructions with strict correctness guarantees.

2.2 Support of a Codeword in a Perfect Code

Let $v = (v_1, v_2, \ldots, v_n) \in \mathbb{F}_2^n$ be a vector. The *support* of v is defined as

$$\mathrm{supp}(v) = \{\, i \in \{1, 2, \ldots, n\} : v_i \neq 0 \,\}.$$

The Hamming weight of v is defined as $\mathrm{wt}(v) = |\mathrm{supp}(v)|$. Given a set of integers $U \in \{1, 2, \ldots, n\}$, the *indicator vector* $\mathbb{1}_U \in \mathbb{F}_2^n$ is the vector with $\mathrm{supp}(\mathbb{1}_U) = U$.

Given a vector $v \in \mathbb{F}_2^n$, $\overline{v} = (1 + v_1, 1 + v_2, \ldots, 1 + v_n)$, is the complement vector of v. Further, given a set $U \in \{1, 2, \ldots, n\}$, $\overline{U} = \{1, 2, \ldots, n\} \setminus U$, is the complement set of U.

Minimum-support constraint. Let $\mathcal{C} \subseteq \mathbb{F}_2^n$ be a perfect linear $[n, k, 2t{+}1]$ code. Since $\mathcal{C}$ has minimum distance $2t{+}1$, every nonzero codeword s satisfies

$$|\mathrm{supp}(s)| \geq 2t{+}1.$$

Consequently, no nonzero codeword s can be supported entirely inside a set $T \subseteq \{1, \ldots, n\}$ of size at most $2t$. That is,

$$s \in \mathcal{C} \setminus \{0\},\ \mathrm{supp}(s) \subseteq T \quad \implies \quad |T| \geq 2t{+}1.$$

Relevance to the secret-sharing construction. In our construction, the support of s has no special algebraic significance. What matters is that the code has minimum distance $2t{+}1$, which guarantees the correct decodability from at most t errors; and the code is perfect, which guarantees the incorrect decodability from at least $t{+}1$ errors. Consequently, the scheme allows the authorized set of participants to reconstruct the secret, whereas the unauthorized set cannot.

2.3 Secret Sharing Scheme

Secret sharing enables a secret to be distributed among a set of participants in such a way that only authorized subsets can reconstruct it, while unauthorized subsets learn nothing about the secret. Formally, a secret sharing scheme allows a dealer to encode a secret into multiple shares and distribute them among participants according to a predefined access structure. The design of such schemes often uses algebraic techniques, particularly polynomial interpolation over finite fields, and has deep connections with error-correcting codes. A special and widely used class of secret sharing schemes is the (k, n)-threshold scheme. In such a scheme, a dealer distributes shares of a secret among n participants such that any subset of at least k participants can reconstruct the secret, while any subset of fewer than k participants obtains no information about the secret. Formally, let

$\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ be the set of participants. The access structure of a (k, n)-threshold scheme is defined as

$$\Gamma = \{A \subseteq \mathcal{P} \mid |A| \geq k\}.$$

A (k, n)-threshold scheme is said to be perfect if for every subset $A \subseteq \mathcal{P}$ with $|A| < k$,

$$H(s \mid \{s_i : P_i \in A\}) = H(s),$$

and for every subset $B \subseteq \mathcal{P}$ with $|B| \geq k$, the secret s can be uniquely reconstructed from $\{s_i : P_i \in B\}$, where $H(s)$ is the entropy of s.

The essential components of a secret sharing scheme are described as follows [21, 25]:

- **Shares:** The individual pieces of data derived from the secret and distributed to participants. Only specific combinations of these shares, belonging to authorized subsets, can be used to reconstruct the secret.
- **Dealer:** A trusted entity responsible for selecting the secret and distributing the corresponding shares to all participants according to the scheme.
- **Access Structure:** A monotone collection of subsets of participants, each of which is authorized to reconstruct the secret. The minimal subsets in this collection are called minimal qualified sets.
- **Information Rate:** The ratio between the size (entropy) of the secret and the size of a participant's share. It reflects the efficiency of the scheme. Formally, for participant P_i with share S_i, the information rate is $\rho = \frac{H(S)}{H(S_i)}$.
- **Ideal Scheme:** A secret sharing scheme is called ideal if the size of each share equals the size of the secret, i.e., the information rate $\rho = 1$ for all participants.
- **Perfect Scheme:** A scheme is perfect if every unauthorized subset of participants has no information about the secret; otherwise, it is non-perfect.
- **Ramp secret sharing:** A (ℓ, t, n)-ramp secret sharing scheme distributes a secret among n participants such that any set of at most ℓ participants obtains no information about the secret, any set of at least t participants can reconstruct the secret, and coalitions of size between $\ell + 1$ and $t - 1$ may obtain partial information.
- **Information leakage function :** The information leakage function from a coalition of j participants is defined as $I_S(j)$ is the max number of bits of the secret s can be obtained by the coalition. Let $\mathcal{S}$ be the space of secrets. For a perfect (t, n)-threshold scheme,

$$I_{\mathcal{S}}(j) = \begin{cases} 0 & \text{for } 0 \leq j < t \\ \log_2 |\mathcal{S}| & \text{for } j \geq t \end{cases}$$

and for a (ℓ, t, n)-ramp secret sharing scheme,

$$I_{\mathcal{S}}(j) = \begin{cases} 0 & \text{for } 0 \leq j \leq \ell \\ \rho_j \in (0, \log_2 |\mathcal{S}|) & \text{for } \ell < j < t \\ \log_2 |\mathcal{S}| & \text{for } j \geq t. \end{cases}$$

3 Proposed scheme

We propose a $(t + 1, 2t + 1)$-threshold secret sharing scheme using a perfect code. Consider a perfect t-error correcting code $\mathcal{C} = [n, k, 2t + 1]$ of length n over $\mathbb{F}_2$.

Participants and Setup:

- **Participants ($\mathcal{P}$):** Denote $\mathcal{P} = \{P_1, P_2, \ldots, P_{2t+1}\}$ be the set of $2t+1$ participants.
- **Secret (s):** The secret $s \in \mathcal{C}$ is chosen as a codeword from $\mathcal{C}$.
- **Helper Vector (h):** A helper vector $h \in \mathbb{F}_2^n$ is chosen satisfying

$$d(s,h) = 2t+1 \quad \text{and} \tag{2}$$

$$|\text{supp}(s+h) \cap \text{supp}(s)| \leq t. \tag{3}$$

To get such a vector, choose two arbitrary subsets A, B such that

$$A \subseteq \text{supp}(s) \text{ with } |A| \leq t, \text{ and}$$

$$B \subseteq \text{supp}(\overline{s}) \text{ with } |B| = 2t+1-|A|.$$

Then define

$$h := s + \mathbb{1}_{A \cup B}. \tag{4}$$

By construction, $d(s,h) = \text{wt}(\mathbb{1}_{A\cup B}) = |A \cup B| = 2t+1$, and $|\text{supp}(s+h) \cap \text{supp}(s)| = |A| \leq t$.

If s is a nonzero codeword, then $\text{wt}(s) \geq 2t+1$ and hence A can be chosen any subset of $\text{supp}(s)$ with cardinality $0 \leq |A| \leq t$. There are $\sum_{i=0}^{t} \binom{\text{wt}(s)}{i} \geq \sum_{i=0}^{t} \binom{2t+1}{i}$ options to choose such a set A. Further, to choose a subset B of $\text{supp}(\overline{s})$ with cardinality $|B| = 2t+1-|A|$, it needs to be that $|\text{wt}(\overline{s})| = n - \text{wt}(s) \geq |B| = 2t+1-|A|$. That implies, $\text{wt}(s) \leq n + |A| - 2t - 1$. For instance, if $|A| = t$, then $\text{wt}(s) \leq n - t - 1$ and if $|A| = t$, then $\text{wt}(s) \leq n - 2t - 1$. There are many codewords with such a weight bound in the existing perfect codes. There are $\binom{n-\text{wt}(s)}{2t+1-|A|}$ options to choose such a set B. Hence, in total, there are $\sum_{i=0}^{t} \binom{\text{wt}(s)}{i}\binom{n-\text{wt}(s)}{2t+1-i}$ options to choose from a $s \in \mathcal{C}$ and the weight constraint restricts the valid secret space to $\mathcal{S}_{\text{valid}} = \{\, s \in \mathcal{C} \mid 2t+1 \leq \text{wt}(s) \leq n-t-1 \,\}$, with relative size $\eta = \frac{|\mathcal{S}_{\text{valid}}|}{|\mathcal{C}|} = \frac{\sum_{i=2t+1}^{n-t-1} A_i}{2^k}$, where A_i is the number of codewords of weight i. For the known perfect code families

  - Repetition code $[2t+1, 1, 2t+1]$: The condition is unsatisfiable since $2t+1 > t$, yielding $\mathcal{S}_{\text{valid}} = \varnothing$.
  - Hamming code $[2^m-1, 2^m-1-m, 3]$: Excludes only the zero, all-one codewords i.e., $\mathcal{S}_{\text{valid}} = \mathcal{C} \setminus \{00\ldots0, 11\ldots1\}$. Hence, $\eta = \frac{2^k-2}{2^k} \approx 1$ where $k = 2^m - 1 - m$.
  - Golay code $[23, 12, 7]$: The valid set size is $|\mathcal{S}_{\text{valid}}| = 3082$ out of $|\mathcal{C}| = 4096$ codewords, giving $\eta \approx 0.752$.

If s is the zero codeword, then A is empty set and B to be chosen as a subset of $\{1, 2, \ldots, n\}$ with $|B| = 2t+1$. Hence, there are $\binom{n}{2t+1}$ choices to choose. As a result, a helper vector satisfying the stated conditions always exists and is abundant.

Share Distribution: Denote $Z = \text{supp}(s+h) = \{i \in \{1, 2, \ldots, n\} \mid s_i \neq h_i\}$, the set of disagreement indices between s and h. Hence, $|Z| = d(s,h) = 2t+1$. The shares are distributed as follows:

- The dealer distributes the shares to participants using a random bijection $\mathsf{Sh} : \mathcal{P} \to Z$. Denote the share distributed to $P_j \in \mathcal{P}$ is $\mathsf{Sh}_j = \mathsf{Sh}(P_j)$ for $1 \leq j \leq n$.
- Note that the helper vector h is public, while each participant's share is private, and no participant gains information about another participant's share from its own index.

3.1 Secret Reconstruction

The reconstruction of the secret is achievable by any subset of $t+1$ or more participants. Let $T \subseteq \mathcal{P}$ be a subset of $t+1$ or more participants. Denote $Z_T = \{\mathsf{Sh}_i : i \in T\} \subseteq Z$ as the set of indices received from a set of $t+1$ participants T.

Define a vector $\tilde{h} \in \mathbb{F}_2^n$ as

$$\tilde{h} := h + \mathbb{1}_{Z_T}. \tag{5}$$

The secret is returned by decoding $\tilde{h}$ to the nearest codeword, that is the secret is $\texttt{Decode}(\tilde{h})$.
We prove the correctness of the reconstruction in Theorem 1.

Theorem 1 (Correctness)**.** *The proposed scheme is a* $(t+1, 2t+1)$ *threshold secret sharing scheme.*

Proof. Recall that $Z = \mathrm{supp}(s+h)$ with $|Z| = 2t+1$, and for a set of participants T, let $Z_T \subseteq Z$ denote the indices corresponding to their shares. For $\tilde{h} = h + \mathbb{1}_{Z_T} \in \mathbb{F}_2^n$, we have

$$s + \tilde{h} = s + h + \mathbb{1}_{Z_T} = \mathbb{1}_Z + \mathbb{1}_{Z_T} = \mathbb{1}_{Z \setminus Z_T}$$

as $Z_T \subseteq Z$. That implies,

$$d(s, \tilde{h}) = |\mathrm{supp}(s + \tilde{h})| = |Z \setminus Z_T| = |Z| - |Z_T|.$$

Case 1 (authorized set): In this case, $|T| \geq t+1$, i.e., $|Z_T| \geq t+1$. Then, we have

$$d(s, \tilde{h}) = |Z \setminus Z_T| \leq (2t+1) - (t+1) = t.$$

Since $\mathcal{C}$ is a code with minimum distance $2t+1$, its unique decoding radius is t. Therefore, the decoding algorithm $\texttt{Decode}(\tilde{h})$ will return the secret codeword s.
Case 2 (unauthorized set): In this case, $|T| \leq t$, i.e., $|Z_T| \leq t$. Then, we have

$$d(s, \tilde{h}) = |Z \setminus Z_T| \geq (2t+1) - t = t+1.$$

Since $\mathcal{C}$ is a perfect code, the vector $\tilde{h}$ belongs to a unique ball of radius t. Further, since $\mathcal{C}$ is a code with minimum distance $2t+1$, the vector $\tilde{h}$ belongs to a ball of radius t which does not contain the codeword s. Therefore, the decoding algorithm $\texttt{Decode}(\tilde{h})$ will return a codeword $\tilde{s} \in \mathcal{C}$ which is different than s. It does not return the secret s and hence fails to return the correct secret. □

Remark 1 (Usefulness of Perfect code). The correctness of the proposed reconstruction algorithm relies critically on the perfectness of the underlying code.

- If $\mathcal{C}$ is not a perfect t-error-correcting code, then the Hamming balls of radius t centered at codewords of $\mathcal{C}$ do not partition $\mathbb{F}_2^n$.
- If C has minimum distance $2t+1$ but is not perfect, then there exists a vector $y \in \mathbb{F}_2^n$ such that $d(y, c) > t \quad$ for all $c \in \mathcal{C}$. If a unauthorised set T has the reconstructed vector $\tilde{h} = y$, then $d(y, s) > t$. Although y does not belong to any of the Hamming balls of radius t centered at codewords, the nearest codeword to y still could be the secret s and hence decodes s correctly. Therefore, there is a high possibility that an unauthorised set can reconstruct the secret.

3.2 Achievable Thresholds from the Perfect Codes

Since binary perfect linear codes are completely classified, we can explicitly list the achievable $(t+1, 2t+1)$-threshold parameter. This provides a practical reference for implementers and highlights the parameter constraints inherent in the construction.

1. Repetition codes: For any integer $t \geq 1$, the binary $[2t+1, 1, 2t+1]$ repetition code is perfect. This yields a $(t+1, 2t+1)$-threshold scheme for any desired value of t.

2. Hamming codes: Since the binary Hamming code has parameters $[2^m-1, 2^m-1-m, 3]$ for integer $m \geq 2$, this yields only $(2, 3)$-threshold schemes.

3. Golay code: Since the binary Golay code has parameters $[23, 12, 7]$, this yields a $(4, 7)$-threshold scheme.

3.3 An example

Let $\mathcal{C}$ be the binary Hamming code with parameters $[7, 4, 3]$. Here $t = 1$, so the scheme involves $2t + 1 = 3$ participants and any $t + 1 = 2$ participants can reconstruct the secret. Let the secret be the codeword $s = (1\,0\,1\,0\,1\,0\,1) \in \mathcal{C}$.

The dealer chooses a helper vector $h \in \mathbb{F}_2^7$ such that $d(h, s) = 2t + 1 = 3$ and $|\text{supp}(h + s) \cap \text{supp}(s)| \leq 1$. Let the chosen vector be $h = (0\,1\,1\,1\,1\,0\,1)$. Then $h + s = (1\,1\,0\,1\,0\,0\,0)$, with $Z = \text{supp}(h + s) = \{1, 2, 4\}$.

In the share distribution, each participant receives one distinct index from Z. Suppose two participants cooperate and hold the shares $Z_T = \{1, 2\} \subset Z$.

The reconstructed vector $\tilde{h}$ is computed as

$$\tilde{h} = h + \mathbb{1}_{Z_T} = (0\,1\,1\,1\,1\,0\,1) + (1\,1\,0\,0\,0\,0\,0) = (1\,0\,1\,1\,1\,0\,1).$$

Here, $d(\tilde{h}, s) = 1 \leq t$. Since $\mathcal{C}$ is a code with distance 3, decoding $\tilde{h}$ yields the unique codeword s.

However if one participant, say $Z_T = \{2\}$, tries to recover the secret. In this case,

$$\tilde{h} = h + \mathbb{1}_{Z_T} = (0\,1\,1\,1\,1\,0\,1) + (0\,1\,0\,0\,0\,0\,0) = (0\,0\,1\,1\,1\,0\,1).$$

Then, $d(\tilde{h}, s) = 2 > t$. Since $\mathcal{C}$ is a perfect code with distance 3, decoding $\tilde{h}$ yields a codeword $\tilde{s}$ which is different than s.

4 Discussion

In this section, we discuss secrecy and the information rate of the proposed scheme.

4.1 Privacy

Privacy in the proposed scheme follows from randomized coset encoding and the distance properties of the underlying perfect code. Since shares are coordinate indices revealing disagreement positions between the helper vector and the secret codeword, unauthorized coalitions may obtain limited information. However, this leakage is strictly bounded and does not enable reconstruction below the decoding threshold, resulting in a ramp secret sharing scheme whose privacy and reconstruction properties are governed by the decoding radius of the underlying perfect code.

Theorem 2 (Ramp Secrecy)**.** *Let $\mathcal{C} \subseteq \mathbb{F}_2^n$ be a perfect linear $[n, k, 2t+1]$ code. The proposed secret sharing scheme realizes a $(0, t+1, 2t+1)$-ramp scheme with the following leakage properties.*

(i) Any set of at least $t+1$ participants can uniquely and deterministically reconstruct the secret, and fewer than $t+1$ participants can not reconstruct the secret.

(ii) Any set $W \subseteq \mathcal{P}$ of size $|W| = w \leq t$ learns exactly w bits of information about the secret. That is, $I_{\mathcal{S}}(w) = w > 0$.

Proof. Item 1 is proved as the scheme is a $(t+1, 2t+1)$ threshold secret sharing scheme (Theorem 1).

For proving Item (ii), consider W to be a set of at most t participants, holding shares $Z_W \subset Z$. Then $|Z_W| \leq t$. As $Z_W \subseteq Z = \mathrm{supp}(s+h) = \{i \in \{1, 2, \ldots, n\} \mid s_i \neq h_i\}$, these participants learn that

$$s_i = 1 + h_i \quad \text{for all } i \in Z_W.$$

Thus, for each $i \in Z_W$, the value of s_i is uniquely determined from h_i. Hence, the coalition learns $|W|$ bits of information about s. Further, since s is uniformly distributed over $\mathcal{C}$, conditioning on these w known coordinates restricts the secret to an affine subspace of $\mathcal{C}$ of dimension $k - w$, with all codewords equally likely. Therefore,

$$I_{\mathcal{S}}(w) = I(S|Z_W) = w > 0,$$

where S denotes the random variable corresponding to the secret and $I(X|Y)$ denotes the conditional information. Consequently, the scheme does not satisfy perfect secrecy and is a ramp scheme.

□

4.2 Information Rate and Idealness

The information rate of a secret sharing scheme is defined as

$$\rho = \min \frac{H(s)}{H(\mathsf{Sh}_i)}$$

i.e., the minimum of the ratio between the entropy of the secret (i.e., the number of bits required for its representation) and the entropy of an individual share.

A scheme is said to be ideal if each share has the same size as the secret, i.e., $\rho = 1$ [18,28].

Lemma 1. *The information rate of the proposed scheme is $\rho = \dfrac{k}{\log_2 n}$ where k and n are the dimension and length of the used perfect code.*

Proof. In the proposed scheme, the secret s is a codeword from a binary $[n, k, 2t+1]$ perfect code $\mathcal{C}$. Since $\dim(\mathcal{C}) = k$, the entropy of the secret s is $H(s) = k$ bits. Each share consists of a single index (i.e., integer) $i \in \{1, 2, \ldots, n\}$. Hence, $H(\mathsf{Sh}_i) = \log_2 n$ bits. Therefore, the information rate of the scheme is $\rho = \dfrac{k}{\log_2 n}$. □

Since $\log_2 n < k$ for all nontrivial perfect codes (e.g., Hamming and Golay codes), we have $\rho > 1$. Table 1 summarizes the information rate ρ for each family of binary perfect codes. Hence, the scheme is not ideal in the strict sense, since shares are indices rather than symbols from the same alphabet as the secret. Nevertheless, this design yields compact shares while preserving non-perfect secrecy. In contrast to classical ideal schemes such as Shamir's construction, reconstruction here relies on bounded-distance decoding in a perfect code rather than polynomial interpolation. Thus, the proposed scheme achieves a high information rate with minimal share size, making it particularly attractive for decoding-based and combinatorial secret sharing applications.

Table 1: Information Rate of Binary Perfect Codes

Code	**Code parameter**	**Threshold value**	ρ
Repetition	$[2t+1, 1, 2t+1]$	$(t+1, 2t+1)$	$1/\log_2(2t+1))$
Hamming	$[2^m-1, 2^m-1-m, 3]$	$(2, 3)$	$(2^m-1-m)/\log_2(2^m-1)$ $= O(2^m/m)$
Golay	$[23, 12, 7]$	$(4, 7)$	$\frac{12}{\log_2 23} \approx 8.81$ (constant)

4.3 Comparison

Table 2 compares the proposed scheme with related code-based secret sharing schemes. The proposed construction differs fundamentally in that the secret is represented as a codeword and reconstruction is performed via perfect decoding, avoiding polynomial evaluation, matrix inversion, or linear system solving required in existing schemes [2,6,7,17]. Threshold enforcement in the proposed scheme is directly governed by the decoding radius $t = \lfloor (d-1)/2 \rfloor$, linking access control to the minimum distance of the underlying perfect linear code rather than to code dimension or ramp parameters. This yields a decoding-theoretic interpretation of security with deterministic reconstruction guarantees. The information rate is not ideal, and parameter flexibility is limited; the reduced share size and low-complexity decoding make the scheme well-suited for storage and computation-constrained environments.

Table 2: Comparison of Secret Sharing Schemes

Property	**Proposed Scheme**	[7]	[17]	[2]	[6]
Underlying structure	Perfect linear codes	LCD codes and weighing orthogonal matrices	LCD codes and projection matrices	LCD codes	Linear codes
Secret representation	Codeword $s \in \mathbb{F}_2^n$	Element in $\mathbb{F}_q^k$	Element in $\mathbb{F}_q^k$	Codeword inner products	Element in $\mathbb{F}_q^k$
Dealer computation	Randomized codeword	Vector inner product	Scalar product	Vector inner product	Codeword having weight t
Reconstruction mechanism	Unique coset decoding	LCD decoding	Matrix multiplication	Solving linear system	Linear decoding
Threshold enforcement	Decoding radius $t = \lfloor (d-1)/2 \rfloor$	Ramp type	Code dimension k	Code dimension k	Code dimension k
Participants	$2t+1$	q^n	n	n	n
Share size (in bits)	$\log_2 n$	$\log_2 n + 1$	$k \log_2 q + l$	$\log_2 q$	$\log_2 q$
Number of coalitions	$\binom{2t+1}{i}$	$\sum_{i=1}^{n-1} \frac{(q^n - q^i)}{n!}$	$\binom{n}{k}$	$\sum_{i=1}^{k-1} \frac{(q^k - q^i)}{k!}$	$\geq \binom{n}{d-t}$
Perfect secrecy	No	No	Yes	Yes	Yes
Information rate	$\frac{k}{\log_2 n}$	$\frac{n}{n+1}$	$\frac{n}{n+l}$	$\frac{k}{n}$	1
Ideal	No	No	No	No	Yes
Parameter flexibility	Limited	Moderate	Moderate	Moderate	High

n= Code length, k= Code dimension, d=Minimum distance, t= Error-correcting capacity of code, l= Hash length, q= Field size.

4.4 Discussion

The primary contribution of the proposed construction is the introduction of a decoding-based reconstruction paradigm for threshold secret sharing using perfect codes. In contrast to classical schemes, such as Shamir's, which rely on polynomial interpolation over finite fields, the proposed approach achieves secret recovery through bounded-distance decoding. This perspective yields the following advantages.

- **Low complexity:** Reconstruction is achieved via efficient bounded-distance de-

coding (e.g., simple syndrome decoding for Hamming codes), avoiding polynomial interpolation or linear system solving.

- **Deterministic recovery:** Perfect decoding ensures a unique reconstruction within radius t, eliminating ambiguity.
- **Conceptual clarity:** The scheme offers a combinatorial interpretation of threshold access through error-correction capability.
- **Asthetic :** It is a secret sharing scheme based on coding theory, which is of theoretical interest and combinatorial interest.

However, the proposed scheme relies fundamentally on the existence of perfect error-correcting codes, which are known to be extremely rare. As a consequence, the choice of threshold parameters is limited compared to polynomial-based schemes such as Shamir's [25]. Moreover, the scheme is not strictly ideal, since each share consists of a coordinate index rather than a secret-sized symbol.

5 Conclusion

We proposed a threshold secret sharing scheme based on perfect error-correcting codes, where reconstruction is achieved via unique bounded-distance decoding. The scheme realizes a ramp access structure. Unauthorized coalitions below the threshold may learn limited information, while any $t + 1$ participants can deterministically recover the secret. Compact shares and decoding-based reconstruction distinguish the scheme from polynomial-based constructions. Despite limited parameter flexibility due to the use of perfect codes, the framework offers a clean decoding-theoretic view of threshold secret sharing.

Future work includes extending the construction to non-binary perfect codes, near-perfect or quasi-perfect codes, exploring adaptive security models based on the privacy of the helper vector h, and incorporating verifiability properties for adversarial settings.

References

[1] D. Q. Adams et al., "Lower Bounds for Leakage-Resilient Secret-Sharing Schemes against Probing Attacks," 2021 IEEE International Symposium on Information Theory (ISIT), Melbourne, Australia, 2021, pp. 976-981.

[2] A. Alahmadi, A. Altassan, A. AlKenani, S. Çalkavur, H. Shoaib, and P. Solé, "A multisecret-sharing scheme based on LCD codes," *Mathematics*, vol. 8, no. 2, Art. 272, 2020.

[3] G. R. Blakley, "Safeguarding cryptographic keys," in *Proc. Nat. Comput. Conf. (AFIPS)*, vol. 48, 1979, pp. 313–317.

[4] C. Blundo, A. De Santis, D. R. Stinson, and U. Vaccaro, "Graph decompositions and secret sharing schemes," *J. Cryptol.*, vol. 8, no. 1, pp. 39–64, 1995.

[5] S. Çalkavur, A. Bonnecaze, R. de la Cruz, and P. Solé, "Code-based secret sharing schemes: Applied combinatorial coding theory," 2022.

[6] S. Çalkavur and P. Solé, "Multisecret-sharing schemes and bounded-distance decoding of linear codes," *Int. J. Comput. Math.*, vol. 94, pp. 107–114, 2017.

[7] S. Çalkavur and P. Solé, "Enhanced multisecret-sharing schemes using LCD codes and weighing orthogonal matrices," *J. Appl. Math. Comput.*, pp. 1–25, 2025.

[8] K. Chatouh, D. Mokhtari, and K. Guenda, "Application of multi-secret sharing schemes using LCD codes over $R = \mathbb{R}[\omega]$," *Comput. Appl. Math.*, vol. 44, no. 3, Art. 123, 2025.

[9] C. Ding, T. Laihonen, and A. Renvall, "Linear multisecret-sharing schemes and error-correcting codes," *J. Comput. Sci.*, vol. 3, pp. 1023–1036, 1997.

[10] J. Ding, C. Lin, and S. Mesnager, "Secret sharing schemes based on the dual of Golay codes," *Cryptogr. Commun.*, vol. 13, pp. 1025–1041, 2021.

[11] C. Ding, J. Yuan, C. S. Calude, M. J. Dinneen, and V. Vajnovszki, "Covering and secret sharing with linear codes," in *Lect. Notes Comput. Sci.*, vol. 2731. Berlin, Germany: Springer, 2003.

[12] R. G. Downey and M. R. Fellows, *Fundamentals of Parameterized Complexity.* Springer, 2013.

[13] I. Dumer, D. Micciancio, and M. Sudan, "Hardness of approximating the minimum distance of a linear code," *IEEE Trans. Inf. Theory*, vol. 49, no. 1, pp. 22–37, 2003.

[14] T. Etzion and A. Vardy, "Perfect binary codes: Constructions, properties, and enumeration," *IEEE Trans. Inf. Theory*, vol. 40, no. 3, pp. 754–763, 2002.

[15] O. Farras, T. B. Hansen, T. Kaced, and C. Padró, "On the information ratio of non-perfect secret sharing schemes," *Algorithmica*, vol. 79, no. 4, pp. 987–1013, 2017.

[16] U. Gupta and H. Mahdavifar, "Bounds on the Statistical Leakage-Resilience of Shamir's Secret Sharing," 2024 IEEE International Symposium on Information Theory (ISIT), Athens, Greece, 2024, pp. 184-189.

[17] H. Ghosh, P. K. Maurya, and S. Bagchi, "Secret sharing scheme based on LCD codes," in *Proc. Int. Interdisciplinary Conf. Mathematics, Engineering and Science (MESIICON)*, IEEE, 2022, pp. 1–4.

[18] W. A. Jackson, K. M. Martin, and C. M. O'Keefe, "Ideal secret sharing schemes with multiple secrets," J. Cryptology, vol. 9, pp. 233–250, 1996.

[19] T. Kaced, "Almost-perfect secret sharing," In 2011 IEEE International Symposium on Information Theory Proceedings, pp. 1603–1607. IEEE, 2011.

[20] S. Ling and C. Xing, *Coding Theory.* Cambridge, U.K.: Cambridge Univ. Press, 2004.

[21] J. L. Massey, "Minimal codewords and secret sharing," in *Proc. 6th Joint Swedish–Russian Workshop on Information Theory*, 1993, pp. 276–279.

[22] J. L. Massey, "Some applications of coding theory in cryptography," in *Cryptography and Coding IV*, Essex, U.K.: Formara Ltd., 1995, pp. 33–47.

[23] R. J. McEliece and D. V. Sarwate, "On sharing secrets and Reed–Solomon codes," *Commun. ACM*, vol. 24, no. 9, pp. 583–584, 1981.

[24] M. Mkadmi, T. Abualrub, and I. Siap, "A verifiable multi-secret sharing scheme based on LCD quadratic residue codes," *Int. J. Comput. Math.*, pp. 1–15, 2025.

[25] A. Shamir, "How to share a secret," *Commun. ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[26] R.Vassantlal, E. Alchieri, B. Ferreira, and A. Bessani, "COBRA: Dynamic proactive secret sharing for confidential BFT services," in *Proc. IEEE Symp. Security and Privacy (SP)*, 2022, pp. 1335–1353.

[27] M. Yoshida, "Hybrid multiplicative non-perfect secret sharing," in *Proc. 2022 IEEE Int. Symp. Inf. Theory (ISIT)*, 2022, pp. 649–653.

[28] J. Yang, S. -T. Xia, X. Wang, J. Yuan and F. W. Fu, "A Perfect Ideal Hierarchical Secret Sharing Scheme Based on the CRT for Polynomial Rings," 2024 IEEE International Symposium on Information Theory (ISIT), Athens, Greece, 2024, pp. 321–326.

[29] J. Yuan, and L. Li, "A fully dynamic secret sharing scheme," Inf. Sci., 496, 42–52, 2019.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 101–115.

Guess-and-Determine Cryptanalysis of the 3-Round L-Feistel

Nickson Cheok-Jin Chong[1], Denis Chee-Keong Wong[1], Kuan-Wai Wong[2], Wun-She Yap[1], Ming-Jie Lee[1] and Bok-Min Goi[1]

[1] Lee Kong Chian Faculty of Engineering and Science, Universiti Tunku Abdul Rahman, Kajang, Malaysia
chongnickson6@1utar.my {deniswong,yapws,leemj,goibm}@utar.edu.my
[2] School of Computing, Asia Pacific University of Technology & Innovation, Kuala Lumpur, Malaysia kuanwai.wong@apu.edu.my

Abstract. The L-Feistel construction is a recent iterative block cipher that unifies the Lai-Massey and Feistel structures via linear matrices satisfying an involutive constraint, where both the branch and round-key sizes are n bits. While this generalization offers design flexibility, its security against classical cryptanalytic techniques requires careful evaluation. In this paper, we present a Guess-and-Determine attack on the three-round variant of L-Feistel. By exploiting the algebraic relations between the internal state variables and the ciphertext, we show that the full $3n$-bit secret key can be recovered with a time complexity of at most 2^{n+d} operations and a data requirement of ideally two known plaintext-ciphertext pairs, where $d = \dim\ker(\mathcal{B})$ with $\mathcal{B} = \left[\begin{smallmatrix} B_1 & B_2 \\ B_2 & B_1 \end{smallmatrix}\right]$. Our results demonstrate that three rounds of L-Feistel offer n-bit security, falling short of the intended $3n$-bit strength.

Keywords: L-Feistel · Guess-and-Determine attack · Block cipher cryptanalysis

1 Introduction

Block ciphers designed in the Feistel tradition, such as DES and its successors, rely on iterating a round function or key to achieve confusion and diffusion. In recent years, the provable security paradigm has shifted toward Key-Alternating Ciphers (KACs) [5], seen as a generalization of a construction proposed by Even and Mansour in 1997 [6], wherein the security reduction relies on the independence of the round keys rather than a complex key schedule.

Liu *et al.* [9] designed a structure, L-Feistel, which is an iterative block cipher that unifies the Lai-Massey and Feistel ciphers and consists of two branches; each branch takes n bits of plaintext. This structure consists of four linear transformation matrices $A_1, A_2, B_1, B_2 \in \mathbb{F}_2^{n\times n}$ that satisfy the structural constraint

$$A_1B_1 \oplus A_2B_2 = 0, \tag{1}$$

to achieve the involutive property. The L-Feistel encryption procedure is given in Algorithm 1, where k_i denotes the round key and s_i denotes the bijective round function. The decryption algorithm is identical to the encryption algorithm, but applies the rounds in reverse order that need to swap the ciphertext as the input to the decryption.

Algorithm 1 Encryption of L-Feistel

Require: $(x_1, x_2) \in P$
Public matrices $A_1, A_2, B_1, B_2 \in \mathbb{F}_2^{n \times n}$
Nonlinear round functions $s_i : \mathbb{F}_2^n \to \mathbb{F}_2^n$
Secret keys $k_i \in \mathcal{K}, i = 1, 2, \ldots, r$
Ensure: $(c_1, c_2) \in C$
$L_0 \leftarrow x_1$, $R_0 \leftarrow x_2$
1: **for** $i = 1$ to r **do**
2: $u_i \leftarrow A_1 L_{i-1} \oplus A_2 R_{i-1} \oplus k_i$
3: $v_i \leftarrow s_i(u_i)$
4: $L'_i \leftarrow L_{i-1} \oplus B_1 v_i$
5: $R'_i \leftarrow R_{i-1} \oplus B_2 v_i$
6: Branch swapping: $L_i \leftarrow R'_i$, $R_i \leftarrow L'_i$
7: **end for**
8: **return** $c_1 \leftarrow R_r$, $c_2 \leftarrow L_r$

The cryptanalysis results for L-Feistel are summarized in Table 1. As shown in Table 1, Zhang *et al.* (2025) [14] applied Yoyo cryptanalysis to the 3-round L-Feistel, achieving a 100% success rate in recovering secret round functions when the matrices are drawn from $\mathrm{GL}(n, \mathbb{F}_2)$. Replacing these matrices with zero or identity transformations extends Yoyo cryptanalysis to additional rounds. Shortly, in the work of Fesenko (2025) [7] in the quantum computing model, and showed that L-Feistel is a special case of the quasi-Feistel scheme with the combiner function. Additionally, Liu *et al.* [9] proposed impossible differential attacks that serve as distinguishers for up to 7 rounds of the L-Feistel structure.

Table 1: Cryptanalysis result on L-Feistel.

Round	Attack	Power	Reference
2	Quantum	Recover secret keys	[7]
3	Yoyo	Recover round functions	[14]
7	Impossible differential[1]	Distinguisher	[9]

Despite the elegance of provable constructions, the practical security of any block cipher is ultimately bounded by the best-known cryptanalytic attacks. While statistical attacks (differential [1] and linear [10] attacks) and algebraic attacks [4] are well-studied, the class of Guess-and-Determine (GD) attacks has received renewed attention due to the advent of automated solver tools. Understanding the resilience of the Key-Alternating L-Feistel cipher against GD is critical for establishing tight security margins.

Guess-and-Determine Attack. The Guess-and-Determine (GD) attack is a classic cryptanalytic technique that combines partial exhaustive search with algebraic deduction. GD attacks partition the cipher's unknown variables into a guess basis $\mathcal{G}$ and a deterministically computable set $\mathcal{D}$ [12]. The attack's complexity is $2^{|\mathcal{G}|}$, making the identification of a minimal guess basis paramount.

The basic GD strategy involves guessing part of the state (e.g., round function inputs or partial subkeys) and propagating known values through the cipher equations. Early applications, such as Pasalic's work on nonlinear filter generators [11], demonstrated the paradigm's strength against stream ciphers.

Historically applied to stream ciphers, GD methodology has been successfully adapted to block ciphers. Biryukov *et al.* [2] at SAC 2015 utilized GD to attack 5- to 7-round

[1] Also for zero-correlation linear and integral attack via dual structure.

Feistel networks employing modular addition, circumventing the limitations of Yoyo attacks. The most significant recent advancement lies in the automation of guess-based search. Tools such as Autoguess [8] harness SAT/SMT and MILP solvers to systematically find minimal guess bases and perform key-bridging—the deduction of all round sub-keys from a minimal guessed subset. Concurrently, the generic framework by Broll *et al.* [3] introduced affine decision trees for S-boxes, optimizing guessing strategies for ciphers like GIFT and RECTANGLE.

Main Contribution. In this paper, we study the classical GD attack against a 3-round key-alternating L-Feistel without a key schedule, where the only secret components are the round keys $k_1, k_2, k_3 \in \mathcal{K}$. The linear transformation matrices A_1, A_2, B_1, B_2 are assumed to be public and elements of $\mathrm{GL}(n, \mathbb{F}_2)$ following Zhang *et al.*'s work [14], which represents a specific subcase of $\mathbb{F}_2^{n\times n}$. By exploiting the algebraic relations between the internal state variables and the ciphertext, we show that the full $3n$-bit secret key can be recovered with a time complexity of only 2^{n+d} operations and a data requirement of merely two known plaintext-ciphertext pairs, where $d = \dim\ker(\mathcal{B})$.

Note 1. The round function is typically implemented using a non-linear component, such as the Substitution-Permutation Network (SPN) style. If the round function were purely linear, attackers could model the cipher as a system of linear equations (linear cryptanalysis) and allow predictable propagation of differences (differential cryptanalysis). Although the round function is non-linear, algebraic cryptanalysis can be applied if the round function has a low algebraic degree [13]. In this paper, we used the public bijective non-linear round function for ease of calculation, as the ultimate goal is to recover the secret keys.

Notation. In this paper, we denote by $\mathbb{F}_2$ the finite field with order 2. The symbol $\mathbb{F}_2^{n\times n}$ refers to the set of all $n \times n$ matrices with entries in $\mathbb{F}_2$. The general linear group of degree n over the field $\mathbb{F}_2$ is written as $\mathrm{GL}(n, \mathbb{F}_2)$, consisting of all invertible $n \times n$ matrices with entries in $\mathbb{F}_2$. We defined the plantexts set, $P = \{(x_1 \| x_2) \mid x_1, x_2 \in \mathbb{F}_2^n\}$, ciphertexts set, $C = \{(c_1 \| c_2) \mid c_1, c_2 \in \mathbb{F}_2^n\}$, and the secret keys set $\mathcal{K} = \{k \mid k \in \mathbb{F}_2^n\}$. Finally, the operator $'\|'$ denotes concatenation, and the operator $\oplus$ denotes the bitwise XOR (exclusive or). Table 2 concludes the linear algebra notations used in the study with the associated meaning.

Table 2: Notations and the Associated Meaning

Symbol	Name	Meaning / Definition in Context
AB	Matrix multiplication	The product of matrices A and B.
$\ker(M)$	Kernel / Null space	The subspace of all vectors x such that $Mx = 0$.
$\mathrm{Im}(M)$	Image / Column space	The subspace spanned by the columns of M (i.e., $\{Mx \mid x \in \mathbb{F}_2^n\}$).
$\mathrm{rank}(M)$	Rank	The dimension of the column space (or image) of matrix M.
$\dim(V)$	Dimension	The number of vectors in a basis for the vector space or subspace V.
$\det(M)$	Determinant	A scalar characterizing the square matrix M; $\det(M) \neq 0$ if and only if M is invertible.
M^{-1}	Matrix inverse	The unique matrix satisfying $MM^{-1} = M^{-1}M = I_n$.

Symbol	Name	Meaning / Definition in Context
$M(V)$	Transformation of a subspace	The image of a subspace V under the linear map defined by M (e.g., $B_2(\ker A) = \{B_2 x \mid x \in \ker A\}$).
$U \cap V$	Intersection	The subspace containing all vectors present in both U and V.
$U + V$	Subspace sum	The subspace defined by $\{u + v \mid u \in U, v \in V\}$.
$\left[\begin{smallmatrix} A & B \\ C & D \end{smallmatrix}\right]$	Block matrix	A matrix partitioned into rectangular sub-matrices (blocks).
$\left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$	Block column vector	A column vector partitioned into sub-vectors.

Organization of Paper. The organization of the paper is briefly outlined as follows: Section 2 details the secret key recovery attack on the 3-round L-Feistel construction, including the guess-and-determine framework, the analysis of both invertible and singular cases for the block matrix $\mathcal{B}$. Section 3 provides the complexity analysis, experimental results, brief discussion, and the preliminary result on extending to more attack rounds. Section 4 concludes the paper with the limitations and future research directions for extended rounds.

2 Secret Key Recovery

At the beginning, we showed the $A_1, A_2, B_1, B_2 \in \mathrm{GL}(n, \mathbb{F}_2)$ are exist and satisfy the structural condition with Equation (1).

Proposition 1. *There exist linear transformation matrices $A_1, A_2, B_1, B_2 \in \mathrm{GL}(n, \mathbb{F}_2)$ that satisfy the structural condition $A_1 B_1 \oplus A_2 B_2 = 0$.*

Proof. Let $A_1, A_2 \in \mathrm{GL}(n, \mathbb{F}_2)$, then the two matrices A_1, A_2 have the full rank. Multiply the inverse of A_1 to the structural condition $A_1 B_1 \oplus A_2 B_2 = 0$, forms

$$B_1 = A_1^{-1} A_2 B_2.$$

By the fact that $\mathrm{rank}(AB) \leq \min(\mathrm{rank}(A), \mathrm{rank}(B))$,

$$\begin{aligned} \mathrm{rank}(B_1) &\leq \min\left(\mathrm{rank}(A_1^{-1} A_2), \mathrm{rank}(B_2)\right) \\ &= \min\left(n, \mathrm{rank}(B_2)\right). \end{aligned}$$

This shows that the rank of B_1 depends on B_2. Therefore, if $\mathrm{rank}(B_1) = n$, then $\mathrm{rank}(B_2) = n$. □

By Proposition 1, the setting $A_1, A_2, B_1, B_2 \in \mathrm{GL}(n, \mathbb{F}_2)$ is valid. For the remainder of this paper, we focus on the 3-round variant ($r = 3$). We assume the round functions $s_i : \mathbb{F}_2^n \to \mathbb{F}_2^n$ are publicly known non-linear bijective mappings, while the round keys $k_i \in \mathcal{K}$ are the sole secret material. We omit the key schedule and treat k_1, k_2, k_3 as independent secret values to evaluate the reduced-round cipher's core structural security.

Tracing Algorithm 1 for $r = 3$ yields the following step-by-step evolution of the internal state.

Round 1.

$$\begin{aligned} &u_1 = A_1 L_0 \oplus A_2 R_0 \oplus k_1, \quad v_1 = s_1(u_1), \\ &L_1 = R_0 \oplus B_2 v_1, \quad R_1 = L_0 \oplus B_1 v_1. \end{aligned} \tag{2}$$

Round 2.

$$\begin{aligned} u_2 &= A_1L_1 \oplus A_2R_1 \oplus k_2, \quad v_2 = s_2(u_2), \\ L_2 &= R_1 \oplus B_2v_2, \quad R_2 = L_1 \oplus B_1v_2. \end{aligned} \tag{3}$$

Round 3.

$$\begin{aligned} u_3 &= A_1L_2 \oplus A_2R_2 \oplus k_3, \quad v_3 = s_3(u_3), \\ c_1 &= R_2 \oplus B_2v_3, \quad c_2 = L_2 \oplus B_1v_3. \end{aligned} \tag{4}$$

By backward substitution, we can express the ciphertext directly in terms of the plaintext and the round function outputs v_i:

$$\begin{aligned} c_1 &= R_0 \oplus B_2v_1 \oplus B_1v_2 \oplus B_2v_3, \\ c_2 &= L_0 \oplus B_1v_1 \oplus B_2v_2 \oplus B_1v_3. \end{aligned} \tag{5}$$

Conversely, if the values v_i are known, the round keys can be recovered by inverting the round functions:

$$\begin{aligned} k_1 &= A_1L_0 \oplus A_2R_0 \oplus s_1^{-1}(v_1), \\ k_2 &= A_1L_1 \oplus A_2R_1 \oplus s_2^{-1}(v_2), \\ k_3 &= A_1L_2 \oplus A_2R_2 \oplus s_3^{-1}(v_3). \end{aligned} \tag{6}$$

2.1 Guess-and-Determine (GD) Attack Framework

Given a single known plaintext-ciphertext pair $(x_1\|x_2,\, c_1\|c_2)$, the goal is to recover the secret round keys (k_1, k_2, k_3). The unknowns are the intermediate values $v_1, v_2, v_3 \in \mathbb{F}_2^n$. A naive brute-force over all three requires 2^{3n} trials. The GD attack reduces this complexity by guessing only v_1 and then deterministically solving for v_2 and v_3 using Equation (5).

Substituting the expressions for L_1 and R_1 into Equation (5) and rearranging, we isolate the terms involving (v_2, v_3) on the left-hand side:

$$B_1v_2 \oplus B_2v_3 = c_1 \oplus R_0 \oplus B_2v_1 = c_1 \oplus L_1, \tag{7}$$

$$B_2v_2 \oplus B_1v_3 = c_2 \oplus L_0 \oplus B_1v_1 = c_2 \oplus R_1. \tag{8}$$

Define the block matrix $\mathcal{B} = \begin{bmatrix} B_1 & B_2 \\ B_2 & B_1 \end{bmatrix} \in \mathbb{F}_2^{2n\times 2n}$. Then Equations (7)–(8) form the $2n \times 2n$ block matrix system

$$\mathcal{B} \begin{bmatrix} v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} c_1 \oplus L_1 \\ c_2 \oplus R_1 \end{bmatrix}. \tag{9}$$

Denote $\tilde{v}_1 \in \mathbb{F}_2^n$ be the guess of v_1. For each guess $\tilde{v}_1$, the right-hand side of Equation (9) becomes fully known. The attack's efficiency hinges on the properties of $\mathcal{B}$. We analysed two cases separately. We analyse the properties of $\mathcal{B}$ using Lemma 1.

Lemma 1. *Let* $\mathcal{B} = \begin{bmatrix} B_1 & B_2 \\ B_2 & B_1 \end{bmatrix}$ *where* $B_1, B_2 \in \mathrm{GL}(n, \mathbb{F}_2)$, *and set* $A = B_1 \oplus B_2$.

(i) $\mathcal{B}$ *is similar to the block-lower-triangular matrix* $T = \begin{bmatrix} A & 0_n \\ B_2 & A \end{bmatrix}$.

(ii) The rank of $\mathcal{B}$ *is given by*

$$\operatorname{rank}(\mathcal{B}) = 2\operatorname{rank}(A) + \dim B_2(\ker A) - \dim\big(B_2(\ker A) \cap \operatorname{Im}(A)\big).$$

(iii) The nullity of $\mathcal{B}$ *is*

$$\dim\ker(\mathcal{B}) = 2\dim\ker(A) - \dim\left(\frac{B_2(\ker A) + \operatorname{Im}(A)}{\operatorname{Im}(A)}\right).$$

(iv) The determinant satisfies $\det(\mathcal{B}) = (\det A)^2$*; hence* $\mathcal{B}$ *is invertible if and only if* A *is invertible.*

(v) For vectors $Y_1, Y_2 \in \mathbb{F}_2^n$*, the linear system*

$$\mathcal{B}\begin{bmatrix} v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$$

is consistent if and only if

$$Y_1 \oplus Y_2 \in \mathrm{Im}(A) \quad \textit{and} \quad Y_2 \in B_2\big(\{w \mid Aw = Y_1 \oplus Y_2\}\big) + \mathrm{Im}(A).$$

When consistent, the solution space is an affine subspace of dimension $\dim\ker(\mathcal{B})$.

Proof. Define the invertible block matrix $U = \begin{bmatrix} I_n & I_n \\ 0_n & I_n \end{bmatrix}$. Since we work over $\mathbb{F}_2$, we have $U^{-1} = U$ because $U^2 = I_{2n}$. A direct computation gives

$$U^{-1}\mathcal{B}U = \begin{bmatrix} I_n & I_n \\ 0_n & I_n \end{bmatrix}\begin{bmatrix} B_1 & B_2 \\ B_2 & B_1 \end{bmatrix}\begin{bmatrix} I_n & I_n \\ 0_n & I_n \end{bmatrix} = \begin{bmatrix} A & 0_n \\ B_2 & A \end{bmatrix} = T.$$

Thus (i) follows.

To prove (ii), we study the column space of T. Write $C(T) = C_1 + C_2$, where

$$C_1 = \left\{ \begin{bmatrix} Ax \\ B_2x \end{bmatrix} : x \in \mathbb{F}_2^n \right\}, \qquad C_2 = \left\{ \begin{bmatrix} 0_n \\ Ay \end{bmatrix} : y \in \mathbb{F}_2^n \right\}.$$

Clearly $\dim C_2 = \mathrm{rank}(A)$. For C_1, decompose $\mathbb{F}_2^n = V \oplus \ker A$, where V is a complement to $\ker A$. On V, the map $x \mapsto \begin{bmatrix} Ax \\ B_2x \end{bmatrix}$ is injective, contributing $\dim V = \mathrm{rank}(A)$ independent vectors. On $\ker A$, the image is $\begin{bmatrix} 0 \\ B_2x \end{bmatrix}$, which gives a subspace of dimension $\dim B_2(\ker A)$. These two contributions are independent because any vector from the first part has a non-zero top block unless $x = 0$, while vectors from the second part have a zero top block. Hence

$$\dim C_1 = \mathrm{rank}(A) + \dim B_2(\ker A).$$

Now $C_1 \cap C_2$ consists of vectors $\begin{bmatrix} Ax \\ B_2x \end{bmatrix} = \begin{bmatrix} 0 \\ Ay \end{bmatrix}$; the top block forces $x \in \ker A$, and the bottom requires $B_2x \in \mathrm{Im}(A)$. Thus

$$\dim(C_1 \cap C_2) = \dim\big(B_2(\ker A) \cap \mathrm{Im}(A)\big).$$

Applying the dimension formula for sums of subspaces gives

$$\mathrm{rank}(\mathcal{B}) = \mathrm{rank}(T) = \dim(C_1 + C_2) = \dim C_1 + \dim C_2 - \dim(C_1 \cap C_2),$$

which yields the formula in (ii).

Part (iii) follows immediately from the rank–nullity theorem. Using $2n = 2\,\mathrm{rank}(A) + 2\dim\ker(A)$ and the result of (ii), we obtain

$$\begin{aligned} \dim\ker(\mathcal{B}) &= 2n - \mathrm{rank}(\mathcal{B}) \\ &= 2\dim\ker(A) - \big[\dim B_2(\ker A) - \dim(B_2(\ker A) \cap \mathrm{Im}(A))\big] \\ &= 2\dim\ker(A) - \dim\left(\frac{B_2(\ker A) + \mathrm{Im}(A)}{\mathrm{Im}(A)}\right), \end{aligned}$$

where the last equality uses the standard identity $\dim(U+V)-\dim V = \dim U-\dim(U\cap V)$ with $U = B_2(\ker A)$ and $V = \mathrm{Im}(A)$.

For (iv), since T is block-triangular, its determinant is $(\det A)^2$. Similarity preserves determinants, so $\det(\mathcal{B}) = (\det A)^2$. Therefore $\mathcal{B}$ is invertible if and only if $\det A \neq 0$, i.e. if and only if A is invertible.

Finally, for (v), apply the change of variables $\begin{bmatrix} w \\ v_3 \end{bmatrix} = U^{-1}\begin{bmatrix} v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} v_2 \oplus v_3 \\ v_3 \end{bmatrix}$. Multiplying the original system on the left by U^{-1} yields the equivalent system $\begin{bmatrix} A & 0 \\ B_2 & A \end{bmatrix}\begin{bmatrix} w \\ v_3 \end{bmatrix} = \begin{bmatrix} Y_1 \oplus Y_2 \\ Y_2 \end{bmatrix}$.

The first block equation is $Aw = Y_1 \oplus Y_2$, which has a solution if and only if $Y_1 \oplus Y_2 \in \mathrm{Im}(A)$. For any such w, the second block equation becomes $Av_3 = Y_2 \oplus B_2 w$, which is solvable for v_3 if and only if $Y_2 \oplus B_2 w \in \mathrm{Im}(A)$, equivalently $Y_2 \in B_2 w + \mathrm{Im}(A)$. Therefore, a solution exists if and only if there is some w with $Aw = Y_1 \oplus Y_2$ satisfying this inclusion, which is precisely the condition stated in (v). When the system is consistent, its solution set is a coset of $\ker(\mathcal{B})$, so it is an affine subspace of dimension $\dim\ker(\mathcal{B})$. □

2.1.1 Case 1: $\mathcal{B}$ is Invertible

If $B_1 \oplus B_2$ is a full rank matrix then $\mathcal{B}$ is invertible, therefore Equation (9) has a unique solution (v_2, v_3) for every guess $\tilde{v}_1$. Each guess yields exactly one candidate triple (v_1, v_2, v_3). The time complexity is 2^n linear system solves, each costing $\mathcal{O}(n^3)$ bit operations. Verification with a second known plaintext-ciphertext pair eliminates false positives, making the ideal data complexity two pairs. The verification subroutine VERIFY with Algorithm 2 encrypts the second plaintext with the candidate keys and checks whether the resulting ciphertext matches the given one.

Algorithm 2 VERIFY: Key Verification Procedure for 3-Round L-Feistel

Require: Candidate keys $k_1, k_2, k_3 \in \mathcal{K}$
Known plaintext-ciphertext pair $p = (x_1, x_2)$, $c = (c_1, c_2)$
Public matrices $A_1, A_2, B_1, B_2 \in \mathrm{GL}(n, \mathbb{F}_2)$
Bijective round functions $s_1, s_2, s_3 : \mathbb{F}_2^n \to \mathbb{F}_2^n$
Ensure: True if (k_1, k_2, k_3) are correct, False otherwise
$L_0 \leftarrow x_1$, $R_0 \leftarrow x_2$
for $i = 1$ to 3 **do**
  $u_i \leftarrow A_1 L_{i-1} \oplus A_2 R_{i-1} \oplus k_i$
  $v_i \leftarrow s_i(u_i)$
  $L'_i \leftarrow L_{i-1} \oplus B_1 v_i$
  $R'_i \leftarrow R_{i-1} \oplus B_2 v_i$
  Branch swapping: $L_i \leftarrow R'_i$, $R_i \leftarrow L'_i$
end for
if $L_3 = c_1$ **and** $R_3 = c_2$ **then**
  return True
else
  return False
end if

Algorithm 3 formalizes this procedure. It precomputes $\mathcal{B}^{-1}$, exhaustively iterates over all 2^n guesses of $\tilde{v}_1$, uniquely recovers (v_2, v_3) via a single matrix-vector multiplication, derives the candidate keys, and filters them using a second known plaintext-ciphertext pair, effectively eliminating all false positives.

Algorithm 3 GD Attack on 3-Round L-Feistel ($\mathcal{B}$ Invertible)

Require: Two known plaintext-ciphertext pairs $(p^{(1)}, c^{(1)})$, $(p^{(2)}, c^{(2)})$
Public matrices $A_1, A_2, B_1, B_2 \in \mathrm{GL}(n, \mathbb{F}_2)$
Bijective round functions $s_1, s_2, s_3 : \mathbb{F}_2^n \to \mathbb{F}_2^n$
Ensure: Secret round keys $k_1, k_2, k_3 \in \mathcal{K}$

Precompute $\mathcal{B}^{-1} \leftarrow \begin{bmatrix} B_1 & B_2 \\ B_2 & B_1 \end{bmatrix}^{-1}$ {One-time $\mathcal{O}(n^3)$ cost}
$L_0^{(1)} \leftarrow x_1^{(1)}$, $R_0^{(1)} \leftarrow x_2^{(1)}$
for all $\tilde{v}_1 \in \mathbb{F}_2^n$ **do**
  $L_1 \leftarrow R_0^{(1)} \oplus B_2\tilde{v}_1$
  $R_1 \leftarrow L_0^{(1)} \oplus B_1\tilde{v}_1$
  $Y_1 \leftarrow c_1^{(1)} \oplus L_1$
  $Y_2 \leftarrow c_2^{(1)} \oplus R_1$
  $\begin{bmatrix} v_2 \\ v_3 \end{bmatrix} \leftarrow \mathcal{B}^{-1} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$ {Unique solution}
  $u_1 \leftarrow s_1^{-1}(\tilde{v}_1)$, $u_2 \leftarrow s_2^{-1}(v_2)$, $u_3 \leftarrow s_3^{-1}(v_3)$
  $k_1 \leftarrow A_1L_0^{(1)} \oplus A_2R_0^{(1)} \oplus u_1$
  $k_2 \leftarrow A_1L_1 \oplus A_2R_1 \oplus u_2$
  $L_2 \leftarrow R_1 \oplus B_2v_2$, $R_2 \leftarrow L_1 \oplus B_1v_2$
  $k_3 \leftarrow A_1L_2 \oplus A_2R_2 \oplus u_3$
  if VERIFY$(k_1, k_2, k_3, p^{(2)}, c^{(2)})$ = `True with Algorithm 2` **then**
    return k_1, k_2, k_3
  end if
end for
return Failure

2.1.2 Case 2: $\mathcal{B}$ is Non-Invertible

We now treat the situation when $B_1 \oplus B_2$ is not a full-rank matrix, hence $\mathcal{B}$ is not invertible. To understand the solvability of system (9), we use the result from Lemma 1, the property of $\mathcal{B}$.

Explanation of Algorithm 4 using Lemma 1. The algorithm exploits the structural properties given in Lemma 1 to handle the possibly singular block matrix $\mathcal{B}$ efficiently.

1. **Consistency check via Gaussian elimination.** For each guess $\tilde{v}_1$, the right-hand side (Y_1, Y_2) is computed. Lemma 1(v) states two necessary and sufficient conditions for the system $\mathcal{B}[v_2; v_3] = [Y_1; Y_2]$ to be solvable. Instead of testing these algebraic conditions directly, the algorithm forms the augmented matrix $[\mathcal{B} \mid Y]$ and performs Gaussian elimination over $\mathbb{F}_2$. The appearance of a contradictory row $[0 \cdots 0 \mid 1]$ indicates that the conditions of Lemma 1(v) are violated; the guess $\tilde{v}_1$ is therefore discarded. This on-the-fly detection avoids the need for an explicit (and error-prone) implementation of the two consistency criteria.

2. **Enumeration of the solution space.** If the system is consistent, Lemma 1(v) guarantees that the set of valid pairs (v_2, v_3) is an affine subspace of dimension $d = \dim\ker(\mathcal{B})$. The exact value of d is given by Lemma 1(iii); it reduces to $2\dim\ker(A)$ only in special cases (e.g. when $B_2(\ker A) \subseteq \mathrm{Im}(A)$, $\mathcal{B}$ is invertible). The algorithm precomputes a basis $\{\delta_1, \ldots, \delta_d\}$ of $\ker(\mathcal{B})$. After obtaining one particular solution $(\tilde{v}_2, \tilde{v}_3)$ (by setting free variables to zero and back-substituting), it enumerates all 2^d solutions as $(\tilde{v}_2, \tilde{v}_3) \oplus \sum_{i=1}^{d} \epsilon_i\delta_i, \quad \epsilon_i \in \{0, 1\}$. For each such shift,

a full candidate triple (v_1, v_2, v_3) is formed and the corresponding round keys are derived using Equation (6). The second known plaintext-ciphertext pair eliminates false positives, ensuring that only the correct key material survives.

Note that enumerating the entire kernel is essential: when $\mathcal{B}$ is not invertible, several different (v_2, v_3) pairs satisfy the ciphertext equations for the correct guess of v_1, but only one of them matches the actual internal state of the encryption. The attack must therefore test every solution; Picking an arbitrary one would risk discarding the correct key candidate. By Lemma 1, since $\mathcal{B}$ is singular with a large value of d, Algorithm 4 collapses to the trivial baseline and offers zero speed-up.

Algorithm 4 GD Attack on 3-Round L-Feistel (Not Invertible $\mathcal{B}$)

Require: Two known plaintext-ciphertext pairs $(p^{(1)}, c^{(1)})$, $(p^{(2)}, c^{(2)})$
Public matrices $A_1, A_2, B_1, B_2 \in \mathrm{GL}(n, \mathbb{F}_2)$
Bijective round functions $s_1, s_2, s_3 : \mathbb{F}_2^n \to \mathbb{F}_2^n$
Ensure: Secret round keys $k_1, k_2, k_3 \in \mathcal{K}$
$L_0^{(1)} \leftarrow x_1^{(1)}$, $R_0^{(1)} \leftarrow x_2^{(1)}$
for all $\tilde{v}_1 \in \mathbb{F}_2^n$ **do**
  $L_1 \leftarrow R_0^{(1)} \oplus B_2\tilde{v}_1$
  $R_1 \leftarrow L_0^{(1)} \oplus B_1\tilde{v}_1$
  $Y_1 \leftarrow c_1^{(1)} \oplus L_1$
  $Y_2 \leftarrow c_2^{(1)} \oplus R_1$
  Form the augmented matrix $\left[\, \mathcal{B} \mid Y \,\right]$ where $\mathcal{B} = \begin{bmatrix} B_1 & B_2 \\ B_2 & B_1 \end{bmatrix}$ and $Y = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$
  Perform Gaussian elimination over $\mathbb{F}_2$ to obtain row-echelon form
  if the system is inconsistent **then**
    continue {this guess of $\tilde{v}_1$ is wrong}
  end if
  Choose a particular solution (v_2, v_3) by setting free variables to 0 and back-substituting
  $u_1 \leftarrow s_1^{-1}(\tilde{v}_1)$
  $u_2 \leftarrow s_2^{-1}(v_2)$
  $u_3 \leftarrow s_3^{-1}(v_3)$
  $k_1 \leftarrow A_1L_0^{(1)} \oplus A_2R_0^{(1)} \oplus u_1$
  $k_2 \leftarrow A_1L_1 \oplus A_2R_1 \oplus u_2$
  $L_2 \leftarrow R_1 \oplus B_2v_2, \quad R_2 \leftarrow L_1 \oplus B_1v_2$
  $k_3 \leftarrow A_1L_2 \oplus A_2R_2 \oplus u_3$
  if VERIFY$(k_1, k_2, k_3, p^{(2)}, c^{(2)}) =$ True with Algorithm 2 **then**
    return k_1, k_2, k_3
  end if
end for
return Failure

3 Discussion

3.1 Complexity Analysis

Table 3 summarizes the resources required by the GD attack. When $\mathcal{B}$ is invertible ($d = 0$), the system has a unique solution, and the attack examines exactly 2^n candidate triples, giving complexity $O(2^n \cdot n^3)$. When $\mathcal{B}$ is not invertible, the solution space is an affine subspace of dimension $d = \dim\ker(\mathcal{B})$ (Lemma 1(v)). For each guess $\tilde{v}_1$ that yields a

consistent system, the algorithm enumerates all 2^d solutions. In the worst case, every guess leads to a consistent system, producing at most $2^n \cdot 2^d = 2^{n+d}$ candidate triples, hence the $O(2^{n+d} \cdot n^3)$ bound. In practice, d is often small, so the attack remains highly efficient even when $\mathcal{B}$ is not invertible. The data complexity ideally remains fixed at two known plaintext-ciphertext pairs for both cases.

Table 3: Complexity of the Guess-and-Determine Attack on 3-Round L-Feistel

Case	Condition	Time Complexity	Data (Ideally)
1	$\mathcal{B}$ invertible ($d = \dim\ker(\mathcal{B}) = 0$)	$\mathcal{O}(2^n \cdot n^3)$	2 known pairs
2	$\mathcal{B}$ not invertible ($d = \dim\ker(\mathcal{B}) > 0$)	$\mathcal{O}(2^{n+d} \cdot n^3)$	2 known pairs

Alternative: Guessing the Round Key k_1. An equivalent attack can be mounted by guessing the first round key $k_1 \in \mathcal{K}$ rather than the intermediate value v_1. Given a candidate k_1, the input to the first round function is obtained directly from the known plaintext: $u_1 = A_1 L_0 \oplus A_2 R_0 \oplus k_1$. Because s_1 is publicly known, the attacker simply evaluates $v_1 = s_1(u_1)$ in the forward direction, bypassing any need to invert the S-box. The state values L_1 and R_1 are then computed exactly as in the encryption algorithm. At this point, the remainder of the attack proceeds unchanged: the linear system derived from Equation (5) is solved for v_2 and v_3, and the remaining round keys are subsequently recovered using Equation (6).

This variant offers two practical advantages. First, it relaxes the requirement that the first round function s_1 be bijective; only forward computability is needed for s_1 (the inversion of later round functions is still required to recover k_2 and k_3). Second, it replaces the evaluation of s_1^{-1} with the typically cheaper forward evaluation s_1 during the online guessing phase. The asymptotic time complexity remains $\mathcal{O}(2^{n+d} \cdot n^3)$, and the data complexity ideally stays at two known plaintext–ciphertext pairs.

3.2 Experimental Result

The experimental setup was conducted using Python version 3.13 on a system equipped with a 12th Gen Intel(R) Core(TM) i5-12400F CPU, an NVIDIA GeForce RTX 4060 GPU, and 32.0 GB of RAM.

Table 4: Experimental results of the GD attack on the 3-round L-Feistel.

n	Condition	Rank of $\mathcal{B}$	Time 1 (s)	Time 2 (s)	Time 3 (s)
8	Invertible	16	0.3149	0.0025	0.0026
16		32	0.0194	0.0045	0.0036
32		64	22.4091	10.5202	5.8815
8	Non-invertible	15/14/14	0.0243	0.0094	0.0089
16		31/31/31	0.0100	0.0085	0.0113
32		63/62/63	62.7903	13.4226	46.7843

Table 4 summarizes the experimental results. The whole experiment was conducted using vectorized search with the GPU for acceleration purposes. By setting three different seeds (42, 1234, and 5678), each n was run three times; the three time columns correspond to these independent runs. The experimental results presented in Table 4 empirically

validate the theoretical time complexity of $\mathcal{O}(2^{n+d} \cdot n^3)$ for the proposed attack. In the invertible case ($d = 0$), the complexity reduces to $\mathcal{O}(2^n \cdot n^3)$. The effect of this exponential factor is clearly demonstrated by the rise in execution time as n grows: while the attacks for $n = 8$ and $n = 16$ complete in well under one second (excluding the initial GPU warm-up run), the $n = 32$ invertible case takes around 22.4091 seconds for the first run and less than 11 seconds for subsequent runs, reflecting the 2^n growth combined with the n^3 cost of processing $2n \times 2n$ matrices. When $\mathcal{B}$ is non-invertible, the nullity $d > 0$ introduces an additional multiplicative factor 2^d from the coset expansion of intermediate states. For $n = 8$ and $n = 16$, the small nullities ($d \in \{1, 2\}$) keep the 2^{n+d} search space small, and the runtimes stay below a tenth of a second. For $n = 32$, the base exponential cost is already noticeable, and the extra 2^d factor further amplifies the workload. Nevertheless, the attack remains practical: the three runs complete in 62.7903 s, 13.4226 s, and 46.7843 s, with the variation reflecting the sensitivity to the specific singular structure. Ultimately, these results confirm that the attack's feasibility is strictly governed by the exponential search space 2^{n+d} and the cubic linear algebra overhead n^3, in full agreement with the theoretical bounds.

The performance gap between the two cases arises from the algebraic structure of the block-matrix system (9). When $\mathcal{B}$ is invertible, the intermediate state is uniquely determined via a single matrix-vector multiplication, allowing the key recovery to proceed as a deterministic linear sweep over all possible first-round keys using only two known plaintext-ciphertext pairs. In contrast, when $\mathcal{B}$ is singular, the system admits either zero or 2^d solutions per consistent key guess, forcing the algorithm to enumerate the full coset of candidate intermediate states and validate each against a second known pair by re-evaluating the complete three-round non-linear structure. This shifts the computational bottleneck from efficient GF(2) linear algebra to the repeated evaluation of expensive non-linear components, which accounts for the significant overhead and the increased runtime variance observed in the non-invertible cases.

3.3 Discussion

Comparison with Yoyo and Classical Cryptanalysis. It is crucial to distinguish the GD attack from existing cryptanalytic techniques on the L-Feistel structure. Zhang *et al.* [14] applied Yoyo cryptanalysis to the 3-round L-Feistel, but under a fundamentally different threat model where the round functions (S-boxes) are entirely secret. Their approach requires a data complexity of adaptively chosen pairs of 2^{2n} to recover the 2^n-size secret round functions. In contrast, our GD attack operates under the standard KAC model, assuming public S-boxes and secret round keys. Our attack requires only two known plaintext-ciphertext pairs and achieves a time complexity of $\mathcal{O}(2^n)$ (when $\mathcal{B}$ is invertible) to recover the $3n$-bit key. Furthermore, classical attacks such as impossible differentials and zero-correlation linear hulls [9] cover up to 6 or 7 rounds but serve primarily as distinguishers and require specific conditions on the linear matrices. Our GD approach provides unconditional full key recovery for 3 rounds regardless of the public matrices. However, a limitation of the GD approach is its scalability and time complexity; while Yoyo can be extended to 4 or 5 rounds for specific L-Feistel instances by exploiting structural properties, the algebraic entanglement in GD makes it inefficient beyond 3 rounds. The comparison between our GD attack and the Yoyo cryptanalysis is summarized in Table 5, which details the differences in attack targets, data and time complexities, and requirements on the round function. As highlighted in the table, while Yoyo attacks recover secret round functions with adaptive chosen plaintexts, our GD attack recovers secret round keys using only two known plaintext-ciphertext pairs.

Table 5: Comparison of classical cryptanalytic approaches on L-Feistel with secret components.

Attack Type	Target	Rounds	Data Complexity	Time Complexity	Requirement on Round Function
Yoyo [14]	Recover functions	3 (L-Feistel)	$\mathcal{O}(2^{2n})$	$\mathcal{O}(2^{2n})$	Secret
GD (This works)	Recover keys	3 (L-Feistel)	2 known plaintexts	$\mathcal{O}(2^{n+d} \cdot n^3)$	Public bijection

Applicability under Realistic Assumptions. In practice, block ciphers employ specific S-box constructions and key schedules, which warrant further discussion regarding the GD framework:

1. Practical S-boxes: While we modelled the round functions as ideal public bijections for theoretical clarity, practical S-boxes possess specific algebraic properties. If an S-box is not strictly bijective, the inversion step s_i^{-1} yields multiple preimages, which would increase the branching factor of the GD attack. Conversely, if the S-box has a low algebraic degree, the GD framework can be seamlessly hybridized with algebraic cryptanalysis. Moreover, for complex or hardware-oriented S-boxes where analytical inversion is cumbersome, modern automated constraint solvers (e.g., MILP or SMT solvers like AutoGuess) can directly ingest the S-box truth tables or algebraic equations to automate the guess-and-determine process without requiring explicit analytical inversion.

2. Key Schedules: Our analysis assumes independent round keys to evaluate the core structural security of the L-Feistel permutation. In a realistic cipher, a key schedule derives k_1, k_2, k_3 from a master key K. If the key schedule is linear or has a low algebraic degree—a common design choice in lightweight ciphers to minimize gate count—it actually benefits the GD attack. The attacker can simply guess the master key K (or a smaller subset of state variables) and deterministically derive the round keys, potentially reducing the effective guessing complexity. Even if the key schedule introduces non-linear operations, it merely adds a set of algebraic constraints between k_1, k_2, k_3 that can be easily accommodated by SAT/SMT solvers within the GD framework. Thus, the presence of a typical lightweight key schedule does not hinder the fundamental viability of the GD attack.

Towards Extended-Round Attacks. For a 4-round L-Feistel, the ciphertext equations become

$$\begin{aligned} c_1 &= L_0 \oplus B_1 v_1 \oplus B_2 v_2 \oplus B_1 v_3 \oplus B_2 v_4, \\ c_2 &= R_0 \oplus B_2 v_1 \oplus B_1 v_2 \oplus B_2 v_3 \oplus B_1 v_4. \end{aligned} \tag{10}$$

After guessing v_1, we are left with two linear equations over $\mathbb{F}_2^n$ for three unknowns v_2, v_3, v_4. The solution space is an affine subspace of dimension $n + \dim\ker(\mathcal{B}')$, where $\mathcal{B}'$ is a suitable $2n \times 3n$ block matrix. Enumerating all solutions yields a worst-case complexity $\mathcal{O}(2^{n+d'} \cdot n^3)$, which exceeds the 2^n security margin and effectively renders the naive GD approach inefficient for typical n.

To overcome this barrier while preserving the attack's algebraic core, we propose a hybrid Guess-and-Determine with Meet-in-the-Middle strategy. Let the two halves of the known plaintext be (L_0, R_0) and the corresponding ciphertext be (L_4, R_4). The core observation is that the intermediate states after the first round (L_1, R_1) and after the third round (L_3, R_3) are linearly coupled through the outputs of the second and third round

functions, v_2 and v_3. Specifically, from the encryption equations, we derive:

$$\begin{aligned} L_3 &= L_1 \oplus B_1 v_2 \oplus B_2 v_3, \\ R_3 &= R_1 \oplus B_2 v_2 \oplus B_1 v_3. \end{aligned} \tag{11}$$

which yields the block-linear system $\begin{bmatrix} B_1 & B_2 \\ B_2 & B_1 \end{bmatrix} \begin{bmatrix} v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} L_3 \oplus L_1 \\ R_3 \oplus R_1 \end{bmatrix}$. We assume that the block matrix $\mathcal{B}$ is invertible (similar process to a non-invertible $\mathcal{B}$), which is the same sufficient condition required for the 3-round GD attack; in this case, the pair (v_2, v_3) is uniquely determined for any given (L_1, R_1) and (L_3, R_3).

The hybrid attack proceeds as follows. In the forward direction, the adversary guesses the first-round intermediate value $v_1 \in \mathbb{F}_2^n$. Since the matrices A_1, A_2, B_1, B_2 are public, this guess deterministically yields the full state (L_1, R_1) from the known plaintext via $L_1 = R_0 \oplus B_2 v_1$ and $R_1 = L_0 \oplus B_1 v_1$. In the backward direction, the adversary guesses the fourth-round intermediate value $v_4 \in \mathbb{F}_2^n$ and derives the state (L_3, R_3) uniquely from the known ciphertext via $L_3 = R_4 \oplus B_1 v_4$ and $R_3 = L_4 \oplus B_2 v_4$.

For every pair of guesses (v_1, v_4), the above linear system is solved algebraically to recover v_2 and v_3. With these values known, the inputs to the corresponding S-boxes are obtained by applying the inverse public round function $u_i = s_i^{-1}(v_i)$, and all four round keys are immediately recovered as:

$$\begin{aligned} k_1 &= A_1 L_0 \oplus A_2 R_0 \oplus s_1^{-1}(v_1), \\ k_2 &= A_1 L_1 \oplus A_2 R_1 \oplus s_2^{-1}(v_2), \\ k_3 &= A_1 L_2 \oplus A_2 R_2 \oplus s_3^{-1}(v_3), \\ k_4 &= A_1 L_3 \oplus A_2 R_3 \oplus s_4^{-1}(v_4), \end{aligned} \tag{12}$$

where (L_2, R_2) are derived from (L_1, R_1) and v_2 via the standard round transition. Consequently, a full candidate key (k_1, k_2, k_3, k_4) is obtained in constant time per guess-pair.

The attack enumerates 2^n possibilities for v_1 and 2^n for v_4, resulting in a total time complexity of $\mathcal{O}(2^{2n} \cdot n^3)$ for the linear-system solutions. By employing a standard meet-in-the-middle memory trade-off—e.g., storing the forward-derived states in a hash table indexed by (L_1, R_1)—the online phase can be reduced to $\mathcal{O}(2^{2n})$ table lookups. Crucially, a single known plaintext-ciphertext pair is insufficient for filtering, as every (v_1, v_4) pair yields a consistent solution. However, a second known pair acts as a strong filter: the probability that a wrong guess-pair survives is approximately 2^{-2n}, effectively isolating the unique correct 4-round key. This demonstrates that four rounds of the L-Feistel construction offer at most $2n$-bit security against this structural hybrid attack, significantly below the intended $4n$-bit strength. This preliminary result needs a deeper analysis as future work.

4 Conclusion

This paper presents a preliminary study of a naive GD attack on the 3-round L-Feistel in light of the structural properties of the public linear transformation matrices. By exploiting the linear relationship between the internal variables and the ciphertext equations, we showed that the secret round keys can be recovered with a time complexity of $\mathcal{O}(2^{n+d} \cdot n^3)$ and a data complexity of only two known plaintext-ciphertext pairs.

Limitation and Future Work. Our results show that 3-round L-Feistel with $d = 0$ offers at most n-bit security instead of the $3n$-bit key length, however Algorithms 3 and 4 is inefficient for large n. Also, when $\mathcal{B}$ is singular with a large null space, the attack provides zero speed-up. Future work should improve the GD attack by reducing the

guessing material, and extend it to more rounds via hybrid attacks, e.g., combining with a meet-in-the-middle attack on 4-round L-Feistel by decomposing the cipher where the inner 2-round structure is easy to analyse.

Acknowledgments

This research was supported by the Ministry of Higher Education under the Fundamental Research Grant Scheme (FRGS), Project No. FRGS/1/2025/ICT07/UTAR/03/1.

References

[1] Eli Biham and Adi Shamir. Differential cryptanalysis of des-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, 1991.

[2] Alex Biryukov, Gaëtan Leurent, and Léo Perrin. Cryptanalysis of feistel networks with secret round functions. In *International Conference on Selected Areas in Cryptography*, pages 102–121. Springer, 2015.

[3] Marek Broll, Federico Canale, Antonio Flórez-Gutiérrez, Gregor Leander, and María Naya-Plasencia. Generic framework for key-guessing improvements. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 453–483. Springer, 2021.

[4] Nicolas T Courtois and Josef Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In *International conference on the theory and application of cryptology and information security*, pages 267–287. Springer, 2002.

[5] Joan Daemen and Vincent Rijmen. The wide trail design strategy. In *IMA International Conference on Cryptography and Coding*, pages 222–238. Springer, 2001.

[6] Shimon Even and Yishay Mansour. A construction of a cipher from a single pseudorandom permutation. *Journal of Cryptology*, 10(3):151–161, 1997.

[7] Andrii Fesenko. Quantum cryptanalysis of ciphers based on generalized feistel and lai-massey schemes. *Theoretical and Applied Cybersecurity*, 7(3), 2025.

[8] Hosein Hadipour and Maria Eichlseder. Autoguess: a tool for finding guess-and-determine attacks and key bridges. In *International Conference on Applied Cryptography and Network Security*, pages 230–250. Springer, 2022.

[9] Jiajie Liu, Bing Sun, and Chao Li. Design and cryptanalysis of a new iterative structure. *IET Information Security*, 17(3):335–346, 2023.

[10] Mitsuru Matsui. Linear cryptanalysis method for des cipher. In *Workshop on the Theory and Application of Cryptographic Techniques*, pages 386–397. Springer, 1993.

[11] Enes Pasalic. On guess and determine cryptanalysis of lfsr-based stream ciphers. *IEEE Transactions on Information Theory*, 55(7):3398–3406, 2009.

[12] Aleksandr Anatol'evich Semenov. Guess-and-determine attacks and automatic methods for their construction. *Prikladnaya Diskretnaya Matematika. Supplement*, (11):81–86, 2018.

[13] Hoang Duc Tho et al. An algorithm for improving algebraic degree of s-box based on affine equivalence transformation. *International Journal of Knowledge and Systems Science (IJKSS)*, 8(1):53–64, 2017.

[14] Jiyan Zhang, Yuxin Niu, Zhen Shi, and Ting Cui. Yoyo cryptanalysis against reduced-round l-feistel structure for recovering the secret components. *IET Information Security*, 2025(1):3355095, 2025.

10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 116–133.

Low-Trace Profiling Side-Channel Attack on Kyber basemul Using Random Forest and ANOVA-Based Feature Selection

Gavireddy Diimpull[1], Tanala Ratna Bharath[1], Arijit Kumar Tripathy[1], Rajat Sadhukhan[1] and Biswajeet Sethi[2]

[11]Department of Computer Science and Engineering, Indian Institute of Technology Roorkee, Roorkee 247667, India
{gavireddy_d,tanala_rb,arijit.cs,rajat.sadhukhan}@cs.iitr.ac.in
[22]School of Computer Engineering, KIIT University, Bhubaneswar 751024, India
biswajeet.sethifcs@kiit.ac.in

Abstract. The NIST standardization of CRYSTALS-Kyber as a post-quantum Key Encapsulation Mechanism (FIPS-203) has accelerated its adoption in practical systems. However, ensuring the security of real-world implementations against side-channel attacks remains a critical challenge. In this work, we present a machine learning-based profiling side-channel attack targeting the pair-pointwise multiplication (`basemul`) operation of Kyber, using a publicly available power trace dataset introduced by Rezaeezade *et al.*. Our approach models leakage as a 9-class classification problem based on the Hamming Weight of intermediate values and employs a Random Forest classifier to learn leakage patterns directly from traces. We replace traditional Sum-of-Squared-Differences-based feature selection with an ANOVA-like statistic that effectively captures inter-class variance and identifies informative Points of Interest. Key hypotheses are evaluated using a log-likelihood scoring framework derived from the classifier's probabilistic outputs. Unlike classical template-based approaches, the proposed method does not rely on explicit distributional assumptions and can capture non-linear leakage characteristics. Experimental results demonstrate successful Rank-1 key recovery using approximately 150 attack traces (750 total traces), highlighting the efficiency of the proposed method. Additionally, we observe the presence of key equivalence classes, revealing structural properties of the underlying finite field arithmetic and their impact on leakage behavior.

Keywords: Side-Channel Analysis (SCA) · CRYSTALS-Kyber (ML-KEM) · Machine Learning-based Attacks · Embedded Security

1 Introduction

The NIST standardization [9] of post-quantum cryptography (PQC) has led to the selection of Kyber [3] as a future standard for Key Encapsulation Mechanism (KEM). Kyber KEM, which is built upon the well-established Module Learning With Errors (MLWE) problem, emerged as the sole algorithm selected for standardization as a KEM due to its strong balance between security and performance [2]. With its standardization, Kyber is expected to see widespread deployment across diverse computing environments, including resource-constrained platforms such as embedded microcontrollers [1]. While mathematically secure against quantum adversaries, practical implementations of lattice-based post quantum schemes remain vulnerable to various side-channel analysis (SCA) as detailed in the survey [11]. These side-channel analysis encompasses attacks on both naive [12, 7]

and protected implementations [4] of Kyber. Recent research has enabled systematic evaluation of such attacks through publicly available datasets, notably the one introduced by Rezaeezade et al. [13], which captures power traces during the execution of the `basemul` operation on embedded platforms. Classical approaches, including correlation power analysis (CPA) and template attacks, have been shown to be effective on this dataset [10]. However, these methods typically rely on explicit statistical assumptions and may not fully capture complex leakage patterns present in high-dimensional traces. To address these limitations, machine learning (ML)-based profiling attacks have emerged as a powerful alternative, enabling data-driven modeling of leakage without requiring prior assumptions about its distribution. Unlike prior work that relies on coarse-grained leakage modeling and strong statistical assumptions, we consider a fine-grained multi-class leakage formulation and a fully data-driven approach, enabling more flexible and interpretable modeling of side-channel leakage. While template-based approaches such as [10] achieve strong performance under Gaussian assumptions, our approach avoids explicit distributional modeling and instead leverages a data-driven framework that generalizes to more complex and non-linear leakage scenarios. In this work, we investigate the application of ML techniques to exploit side-channel leakage in Kyber implementations. Recent works have demonstrated the feasibility of side-channel attacks on Kyber implementations, particularly targeting the pair-pointwise multiplication (`basemul`) operation. Classical approaches, including correlation power analysis (CPA) and template attacks, have been shown to successfully recover secret-dependent information from publicly available datasets [10]. These methods rely on explicit statistical models and assumptions about leakage behavior, which may limit their effectiveness in capturing complex, high-dimensional leakage patterns. In parallel, machine learning (ML)-based techniques have emerged as a promising direction for profiling side-channel attacks, enabling data-driven modeling of leakage without requiring prior assumptions. Recent studies have further explored advanced attack settings, such as blind side-channel attacks on Kyber, highlighting the growing role of ML in analyzing post-quantum implementations [12].

In this work, we revisit the profiling attack setting on Kyber's `basemul` operation and investigate the effectiveness of lightweight machine learning models for leakage exploitation. Unlike prior approaches that rely on handcrafted statistical models, we formulate the attack as a multi-class classification problem based on the Hamming Weight of intermediate values and employ a Random Forest classifier to learn leakage patterns directly from power traces. Our approach integrates an ANOVA-like feature selection strategy for identifying informative points of interest and a log-likelihood-based scoring framework for key hypothesis evaluation. Through experimental evaluation on a public dataset, we demonstrate efficient key recovery with a relatively small number of traces. Furthermore, we observe the presence of key equivalence classes in the recovered candidates, indicating that the learned model captures structural properties of the underlying finite field arithmetic. This observation provides additional insight into the nature of leakage in Kyber implementations.

Our main contributions are as follows:

- We present a machine learning-based profiling side-channel attack on Kyber's `basemul` operation, demonstrating that lightweight and interpretable models such as Random Forests can effectively exploit leakage in post-quantum implementations.
- We propose an ANOVA-like feature selection method for identifying Points of Interest in high-dimensional traces, improving leakage separability in a multi-class setting compared to traditional metrics such as SOSD.
- We develop a probabilistic key recovery framework based on log-likelihood scoring over classifier outputs, enabling efficient aggregation of information across traces.
- We demonstrate practical key recovery with Rank-1 performance using approximately 150 attack traces, highlighting the efficiency of the proposed approach.

- We observe the presence of key equivalence classes, providing insights into structural leakage properties of finite field arithmetic in Kyber implementations.

The rest of the paper is organized as follows. Section 3 provides the necessary background on side-channel analysis and the underlying cryptographic operations. Section 2 reviews the related literature in this domain. The proposed attack methodology is detailed in Section 4. Experimental results and evaluation are presented in Section 5. Finally, Section 6 concludes the paper and outlines directions for future work.

2 Related Work

Table 1: Comparison of side-channel attack methodologies on Kyber implementations

Method	Model	Training	Complexity	Remarks
CPA	Linear (HW/HD)	No	Low	Simple and fast but limited to linear leakage models
Template Attack	Gaussian	Yes	High	Accurate but depends on strong statistical assumptions
Deep Learning SCA	Data-driven	Yes	High	Captures complex leakage but less interpretable
Blind ML SCA [12]	Data-driven	No	High	Does not require labels but computationally expensive
This Work	RF + BSS	Yes	Moderate	Efficient, interpretable, and requires fewer traces

Side-channel analysis (SCA) has long been established as a practical threat to cryptographic implementations, exploiting physical leakages such as power consumption and electromagnetic emissions. Classical techniques such as Correlation Power Analysis (CPA) [5, 8] and template attacks [6] remain widely used baselines. CPA relies on statistical correlation between measured traces and hypothetical leakage models, while template attacks construct detailed probabilistic models under Gaussian assumptions. Although highly effective, these approaches depend on explicit leakage models and may struggle to capture complex, high-dimensional leakage behavior.

With the emergence of post-quantum cryptography (PQC), recent research has focused on evaluating the implementation security of lattice-based schemes such as CRYSTALS-Kyber. Several works have demonstrated that core operations, particularly the pair-pointwise multiplication (`basemul`) and its underlying `fqmul` routine, exhibit measurable side-channel leakage on embedded platforms [7, 4]. The publicly available dataset in-

troduced by Rezaeezade et al. [13] has enabled systematic evaluation of such leakage by providing high-resolution power traces collected from a Cortex-M4 microcontroller. Prior work has shown that classical attacks, including CPA and template-based approaches, can successfully recover secret-dependent information from this dataset [10], highlighting the practical vulnerability of Kyber implementations.

In parallel, machine learning (ML)-based approaches have emerged as a powerful alternative for profiling side-channel attacks. Unlike classical methods, ML-based techniques learn the relationship between leakage traces and intermediate values directly from data, without requiring explicit distributional assumptions. Early works have demonstrated that supervised learning models, including neural networks and ensemble methods, can effectively exploit complex and noisy leakage patterns [14]. More recently, advanced attack scenarios such as blind side-channel attacks have been explored, where the attacker does not require labeled profiling data [12]. While these approaches improve generality, they often involve higher computational complexity and reduced interpretability.

This work revisits the profiling attack setting on Kyber's `basemul` operation and focuses on the effectiveness of lightweight and interpretable ML models. In contrast to prior approaches based on deep learning or blind attacks, we demonstrate that a Random Forest classifier, when combined with an ANOVA-like feature selection strategy and log-likelihood-based key scoring, can achieve efficient and accurate key recovery with a relatively small number of traces. Unlike template attacks, our approach does not rely on Gaussian leakage assumptions, and unlike CPA, it leverages probabilistic outputs from the classifier to aggregate evidence across traces. A comparison of the main characteristics of classical and ML-based approaches, including our method, is summarized in Table 1.

To the best of our knowledge, limited work has explored the combination of ANOVA-based feature selection and probabilistic log-likelihood scoring in the context of Kyber `basemul` leakage using lightweight ML models. Our results demonstrate that such a combination provides an effective trade-off between attack performance, interpretability, and computational efficiency.

3 Background

This section provides the necessary background for understanding the proposed attack, including the Kyber `basemul` operation, machine learning-based side-channel analysis, feature selection using ANOVA-like metrics, and log-likelihood-based key scoring.

3.1 Kyber and the `basemul` Operation

CRYSTALS-Kyber (ML-KEM) is a lattice-based key encapsulation mechanism defined over the polynomial ring $R_q = \mathbb{Z}_q[x]/(x^n + 1)$, where $q = 3329$ and $n = 256$. Polynomial multiplication in Kyber is efficiently implemented using the Number Theoretic Transform (NTT), which converts convolution in the polynomial domain into component-wise multiplication in the transformed domain.

Given two polynomials $f, g \in R_q$, their product is computed as:

$$f \cdot g = \mathrm{INTT}(\mathrm{NTT}(f) \circ \mathrm{NTT}(g)), \tag{1}$$

where $\circ$ denotes coefficient-wise multiplication in the NTT domain.

The core operation enabling this multiplication is the pair-pointwise multiplication, referred to as `basemul`. In the NTT representation, each polynomial is decomposed into pairs of coefficients. For two such pairs (x_0, x_1) and (y_0, y_1), the `basemul` operation computes:

$$r_0 = x_0 \cdot y_0 + \zeta \cdot x_1 \cdot y_1 \mod q,\ r_1 = x_0 \cdot y_1 + x_1 \cdot y_0 \mod q, \tag{2}$$

where ζ is a precomputed constant determined by the NTT parameters. Each of the above coefficient multiplications is implemented using the `fqmul` operation, which performs modular multiplication followed by Montgomery reduction. As a result, a single `basemul` invocation consists of multiple `fqmul` operations on individual coefficients in $\mathbb{Z}_q$.

From a side-channel perspective, the leakage exploited in this work originates from these `fqmul` operations. Each `fqmul` processes a pair of coefficients, typically involving one secret-dependent coefficient and one input-dependent coefficient. Let $s_i \in \mathbb{Z}_q$ denote a coefficient of the secret key polynomial (in the NTT domain), and let $b_i \in \mathbb{Z}_q$ denote a corresponding input-dependent coefficient that varies across executions. The `fqmul` operation computes:

$$c_i = s_i \cdot b_i \cdot R^{-1} \mod q, \tag{3}$$

where $R = 2^{16}$ and the reduction is performed using Montgomery arithmetic.

In the publicly available dataset used in this work, specific secret coefficients are denoted using implementation-level notation such as $a[0]$ and $a[1]$. These correspond to instances of the general secret coefficients s_i described above. Since the secret coefficients s_i remain fixed across traces while the inputs b_i vary, repeated executions of `fqmul` expose multiple leakage instances corresponding to the same secret-dependent computation. Moreover, due to the coefficient-wise structure of `basemul`, these operations are performed independently for each index i, enabling fine-grained analysis of individual secret components. The sequence of arithmetic operations in `fqmul`, including multiplication and modular reduction, induces data-dependent variations in power consumption. Consequently, `fqmul` serves as the primary source of exploitable leakage, making it an effective target for profiling attacks aimed at recovering secret key coefficients.

3.2 Side-Channel Analysis using Machine Learning

In profiling-based side-channel analysis, the objective is to learn the relationship between observed leakage traces and secret-dependent intermediate values. Traditional approaches, such as template attacks, construct explicit statistical models of leakage distributions for each possible intermediate value. In contrast, machine learning (ML)-based approaches formulate this problem as a supervised classification task.

Let $\mathbf{x}_j \in \mathbb{R}^d$ denote j^{th} power trace consisting of d time samples, and let $y_j \in \{0, \ldots, C-1\}$ denote the corresponding leakage label derived from a function of the intermediate value (e.g., its Hamming Weight). Given a set of profiling traces $\{(\mathbf{x}_j, y_j)\}_{j=1}^{N_p}$, the goal is to train a classifier f_θ parameterized by θ that approximates the conditional probability distribution:

$$f_\theta(\mathbf{x}) \approx P(y \mid \mathbf{x}). \tag{4}$$

In the profiling phase, the model parameters θ are learned using labeled traces. In the attack phase, the trained model is applied to unseen traces $\mathbf{x}_j$ to compute class probabilities $P(c \mid \mathbf{x}_j)$ for all possible classes $c \in \{0, \ldots, C-1\}$. These probabilities are then used to evaluate key hypotheses by linking each key guess to a predicted leakage class. Compared to classical methods, this data-driven formulation enables the model to capture complex and non-linear relationships between the leakage and the underlying intermediate values, without requiring explicit assumptions about the leakage distribution.

3.3 ANOVA-like BSS for Point of Interest Selection

In side-channel analysis, each trace $\mathbf{x}_j \in \mathbb{R}^d$ consists of d time samples, many of which do not carry useful leakage information. Feature selection aims to identify a subset of informative time indices, referred to as Points of Interest (PoIs), that maximize the separability between leakage classes.

Let $\{(\mathbf{x}_j, y_j)\}_{j=1}^{N_p}$ denote the profiling dataset, where $y_j \in \{0, \ldots, C-1\}$ is the leakage class associated with trace $\mathbf{x}_j$. For a fixed time index $t \in 1, \ldots, d$, we evaluate its discriminative power using an ANOVA-like Between-Class Sum of Squares (BSS) metric:

$$\mathrm{BSS}(t) = \sum_{c=0}^{C-1} n_c \left(\mu_c(t) - \mu_{\mathrm{all}}(t)\right)^2, \tag{5}$$

where n_c is the number of traces belonging to class c, $\mu_c(t)$ is the mean value of class c at time index t, and $\mu_{\mathrm{all}}(t)$ is the overall mean across all traces at t. The BSS score measures the variance between class means at each time index. High values of $\mathrm{BSS}(t)$ indicate that the corresponding time sample provides strong discrimination between classes, making it a suitable candidate for a PoI. This metric corresponds to the numerator of the classical ANOVA F-statistic and is particularly effective for multi-class leakage models such as Hamming Weight. In practice, the top-k time indices with the highest BSS scores are selected as PoIs and used as input features for the classifier, thereby reducing dimensionality while preserving the most informative leakage components.

3.4 Log-Likelihood Scoring

Given a trained classifier f_θ that approximates the posterior distribution $P(c \mid \mathbf{x})$, key hypotheses can be evaluated by aggregating likelihoods across multiple traces. Let $\{\mathbf{x}_j\}_{j=1}^{N_a}$ denote the set of attack traces, and let k_g be a candidate key hypothesis. For each trace $\mathbf{x}_j$, the corresponding intermediate computation under k_g determines a predicted leakage class $c_j(k_g) \in \{0, \ldots, C-1\}$. The classifier provides the probability $P(c_j(k_g) \mid \mathbf{x}_j)$ for this class. The total log-likelihood score for the key hypothesis k_g is then computed as:

$$\mathrm{Score}(k_g) = \sum_{j=1}^{N_a} \log P\left(c_j(k_g) \mid \mathbf{x}_j\right). \tag{6}$$

This formulation aggregates evidence across traces under an independence assumption and corresponds to maximizing the joint likelihood of observing the predicted leakage classes. Equivalently, it can be interpreted as minimizing the cross-entropy between the predicted and hypothetical class distributions. The key hypothesis with the highest log-likelihood score is selected as the most likely candidate. This probabilistic scoring framework naturally leverages the soft outputs of the classifier and provides a robust mechanism for key recovery in the presence of noise. In this work, we use the terms *profiling* and *attack* phases interchangeably with training and testing data, respectively. Building upon the concepts discussed in the previous section, we now describe the proposed attack methodology in detail.

4 Attack Methodology

This section describes the proposed machine learning-based profiling attack, including dataset preparation, leakage modeling, feature selection, and key recovery.

4.1 Dataset Description

The dataset used in this work is obtained from the publicly available side-channel benchmark introduced by Rezaeezade et al. [13]. It consists of power consumption traces collected during the execution of the Kyber `basemul` operation on an STM32F3 microcontroller, based on the ARM Cortex-M4 architecture. This platform is representative of embedded and IoT devices, making it a realistic target for practical side-channel attacks. The dataset

contains approximately 100,000 traces, where each trace $\mathbf{x}_j \in \mathbb{R}^d$ corresponds to a single execution of the `basemul` function and consists of $d \approx 50{,}000$ time samples. While this high resolution enables detailed leakage analysis, it also introduces challenges in terms of computational complexity and noise. To address this, we apply a trace compression strategy by reducing the number of samples by a factor of 10, following the approach in [10]. This preprocessing step reduces dimensionality while preserving the overall leakage structure, thereby improving computational efficiency and stability during training.

The dataset is partitioned into profiling and attack sets, denoted by $\{(\mathbf{x}_j, y_j)\}_{j=1}^{N_p}$ and $\{\mathbf{x}_j\}_{j=1}^{N_a}$, respectively. Unless stated otherwise, we use $N_p = 800$ traces for profiling and $N_a = 200$ traces for attack, corresponding to a standard profiling attack setting. A key feature of this dataset is the availability of intermediate values associated with the cryptographic computation, which enables precise construction of leakage labels y_j. This facilitates supervised learning by allowing each trace to be associated with a class label derived from the underlying intermediate value (e.g., its Hamming Weight). The combination of labeled traces and realistic hardware leakage makes this dataset well-suited for evaluating machine learning-based profiling attacks. The choice of a relatively small profiling set is intentional, as it reflects a realistic attack scenario where the adversary has limited access to labeled traces. It also highlights the efficiency of the proposed approach in low-data regimes.

4.2 Montgomery Reduction in Kyber

The pair-pointwise multiplication (`basemul`) in Kyber relies on the `fqmul` operation, which performs modular multiplication in the finite field $\mathbb{Z}_q$, where $q = 3329$. A direct implementation of modular reduction using division is computationally expensive on embedded platforms. To address this, Kyber employs Montgomery reduction, which replaces division with efficient arithmetic operations such as fixed-width multiplication and bit-shifting. Montgomery reduction operates in a transformed domain using a radix $R = 2^{16}$ and a precomputed constant $q_{\text{inv}} \equiv -q^{-1} \pmod{R}$. This enables modular reduction to be performed using only 16-bit operations, which are highly efficient on microcontroller architectures.

Algorithm 1 describes the Montgomery-based finite field multiplication used in Kyber. Given two 16-bit inputs a and b, the intermediate product $P = a \cdot b$ is first computed. The reduction step then computes a correction term t using the lower 16 bits of P and the constant q_{inv}. This term is chosen such that $P - t \cdot q$ becomes divisible by R. A right shift by $\log_2 R = 16$ bits completes the reduction, yielding the result $c = a \cdot b \cdot R^{-1} \bmod q$. This formulation avoids explicit division and maps efficiently to hardware. At the same time, the sequence of arithmetic operations—particularly multiplications and modular reductions—introduces data-dependent power consumption. As a result, the `fqmul` operation constitutes a primary source of side-channel leakage in Kyber implementations.

Algorithm 1: Montgomery-based finite field multiplication (`fqmul`) in Kyber

Input: 16-bit integers a, b
Output: $c = a \cdot b \cdot R^{-1} \bmod q$, where $R = 2^{16}$
$q \leftarrow 3329$;
$q_{\text{inv}} \leftarrow -3327$;
Function `MONTGOMERY_REDUCE(`P`)`
  $t \leftarrow P \bmod 2^{16}$;
  $t \leftarrow (t \cdot q_{\text{inv}}) \bmod 2^{16}$;
  $c \leftarrow \dfrac{P - (t \cdot q)}{2^{16}}$;
  return c;
Function `FQMUL(`a, b`)`
  $P \leftarrow a \cdot b$;
  return `MONTGOMERY_REDUCE(`P`)`;

4.3 Leakage Model

A critical step in side-channel analysis is defining a leakage model that relates observed power traces to secret-dependent intermediate computations. In this work, we adopt a Hamming Weight (HW) leakage model, which assumes that the power consumption of a device is correlated with the number of bits set to one in the processed data.

Let $\mathbf{x}_j \in \mathbb{R}^d$ denote a power trace corresponding to an execution of the `fqmul` operation, and let c_j denote the associated intermediate value computed as:

$$c_j = s_i \cdot b_j \cdot R^{-1} \mod q, \tag{7}$$

where s_i is a fixed secret coefficient and b_j is an input-dependent value varying across traces.

Instead of modeling the full intermediate value c_j, we focus on its most significant byte (MSB), denoted by $\text{MSB}(c_j)$. This choice is motivated by the observation that higher-order bits often exhibit stronger and more stable leakage characteristics, making them easier to learn using machine learning models. This also reduces the complexity of the learning problem while preserving relevant leakage information.

To formulate the problem as supervised classification, each trace $\mathbf{x}_j$ is assigned a label:

$$y_j = \text{HW}(\text{MSB}(c_j)), \tag{8}$$

where $\text{HW}(\cdot)$ denotes the Hamming Weight function. Since the MSB is an 8-bit value, $y_j \in 0, \ldots, 8$, resulting in a 9-class classification problem. In practice, the intermediate values c_j are obtained from the dataset and stored in a matrix V, which is used for label generation.

Algorithm 2 describes the procedure used to generate leakage labels from the intermediate values. For each trace, the corresponding intermediate value is first extracted, followed by computation of its MSB. The Hamming Weight of this byte is then obtained using a precomputed lookup table, producing the label y_j used for training. The generated label array $L[i]$ corresponds to the class labels y_j used for training.

This multi-class leakage model provides a richer representation compared to binary models and integrates naturally with probabilistic classifiers, enabling effective key recovery through likelihood-based scoring.

Algorithm 2: Label Generation (Hamming Weight of MSB)

Input: Matrix of intermediate values V, target index idx, number of traces N
Output: Array of leakage labels L
Initialize $HW\_TABLE[0 \dots 255]$;
for $j \leftarrow 0$ **to** 255 **do**
  $HW\_TABLE[j] \leftarrow$ COUNT_SET_BITS(j);
for $i \leftarrow 0$ **to** $N-1$ **do**
  $val \leftarrow V[i, idx]$;
  $msb \leftarrow (val \gg 8)\&0xFF$;
  $L[i] \leftarrow HW\_TABLE[msb]$;
return L;

4.4 Feature Selection

Given the high dimensionality of power traces, feature selection is essential for both computational efficiency and classification accuracy. Each trace $\mathbf{x}_j \in \mathbb{R}^d$ contains thousands of time samples, but only a small subset carries significant information about the secret-dependent computation. These informative samples, referred to as Points of Interest (PoIs), correspond to time indices where leakage is maximally discriminative across the $C = 9$ Hamming Weight classes.

To identify such points, we employ an ANOVA-like Between-Class Sum of Squares (BSS) metric. The underlying intuition is that a time index t is informative if the mean leakage values of different classes are well separated at that point. Formally, given the profiling dataset $\{(\mathbf{x}_j, y_j)\}_{j=1}^{N_p}$, the BSS score at time index t is computed as:

$$\mathrm{BSS}(t) = \sum_{c=0}^{C-1} n_c \left(\mu_c(t) - \mu_{\mathrm{all}}(t)\right)^2, \tag{9}$$

where n_c is the number of traces in class c, $\mu_c(t)$ is the mean of traces belonging to class c at time t, and $\mu_{\mathrm{all}}(t)$ is the global mean across all traces.

This metric captures the between-class variance and highlights regions where the classifier can most effectively distinguish between different leakage classes. Compared to commonly used metrics such as Sum of Squared Differences (SOSD) or Signal-to-Noise Ratio (SNR), BSS naturally extends to multi-class settings and provides a stable measure of class separability. In practice, we observe that even omitting the weighting term n_c yields comparable results, indicating robustness of the approach. From the computed BSS scores, we select the top-K time indices as PoIs. These selected indices form a reduced feature representation of each trace, significantly lowering dimensionality while preserving the most informative leakage components. This reduction improves both training efficiency and generalization performance. Notably, when $K = 1$, the selection reduces to identifying a single most informative time sample, analogous to classical correlation-based attacks. The resulting BSS score distributions are illustrated in Figures 1 and 2, where the x-axis represents the time sample index and the y-axis denotes the corresponding BSS score. Prominent peaks in these plots indicate time indices at which the class-wise means are maximally separated, and therefore correspond to strong Points of Interest (PoIs). These high-scoring regions are selected as input features for the classifier, forming the basis of the reduced trace representation used in the profiling phase. The concentration of peaks within specific time intervals suggests that leakage is localized to particular stages of the `fqmul` execution. Furthermore, the observed peak locations closely align with those reported in prior work [10], indicating consistency in the leakage behavior across different analysis methodologies. This agreement validates the effectiveness of the BSS-based approach in identifying meaningful leakage regions.

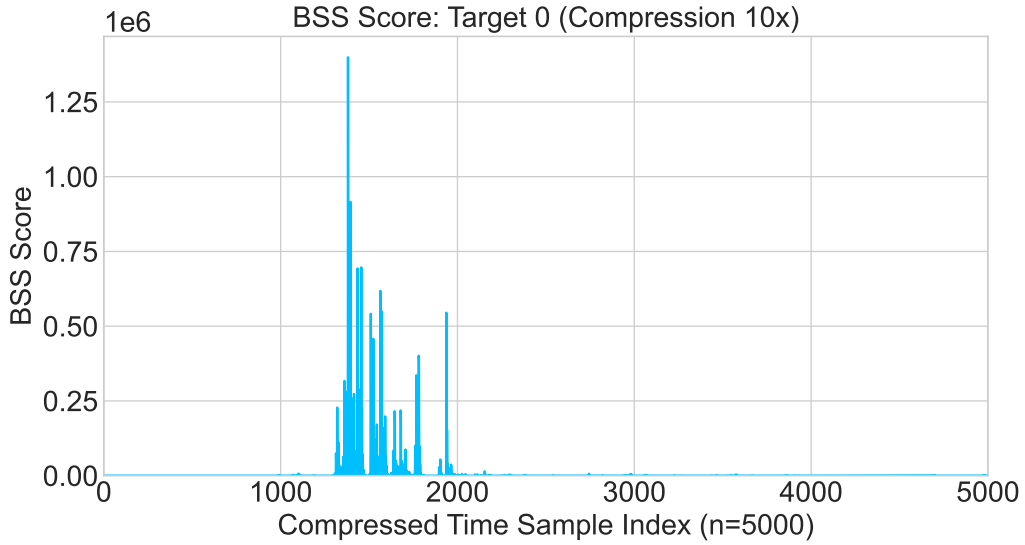


Figure 1: BSS score distribution for target coefficient 0.

Algorithm 3: ANOVA-like BSS for Point of Interest (PoI) Selection

Input: Profiling traces $T \in \mathbb{R}^{N \times d}$, labels $y \in \{0, \ldots, C-1\}^N$, number of PoIs K
Output: Array of selected PoI indices P
$\mu_{\text{all}} \leftarrow \text{MEAN}(T, \text{axis} = 0)$;
$S \leftarrow \mathbf{0} \in \mathbb{R}^d$;
for $c \leftarrow 0$ **to** $C-1$ **do**
  $mask \leftarrow (y = c)$;
  $n_c \leftarrow \text{SUM}(mask)$;
  $\mu_c \leftarrow \text{MEAN}(T[mask], \text{axis} = 0)$;
  $S \leftarrow S + n_c \cdot (\mu_c - \mu_{\text{all}})^2$;
$P \leftarrow \text{INDICES\_OF\_TOP\_K}(S, K)$;
return P;

Algorithm 3 computes the BSS score for each time sample and selects the top-K Points of Interest (PoIs). Given the profiling traces $T \in \mathbb{R}^{N \times d}$ and labels y, the global mean trace μ_{all} is first computed across all samples. An accumulator vector $S \in \mathbb{R}^d$ is initialized to store the BSS scores. The algorithm then iterates over each class $c \in \{0, \ldots, C-1\}$. For each class, a mask is used to extract the corresponding traces, from which the class size n_c and class-wise mean μ_c are computed. The contribution of class c to the BSS score is calculated as $n_c \cdot (\mu_c - \mu_{\text{all}})^2$, and accumulated into S element-wise across all time indices. After processing all classes, S contains the between-class variance at each time sample. The indices of the top-K largest values in S are then selected as the PoIs using $\text{INDICES\_OF\_TOP\_K}(S, K)$. The resulting set P corresponds to the most discriminative time samples used for feature reduction.

4.5 Machine Learning Model

After selecting the Points of Interest (PoIs), each trace $\mathbf{x}_j \in \mathbb{R}^d$ is reduced to a lower-dimensional feature vector $\tilde{\mathbf{x}}_j \in \mathbb{R}^K$ by retaining only the samples at the selected indices.

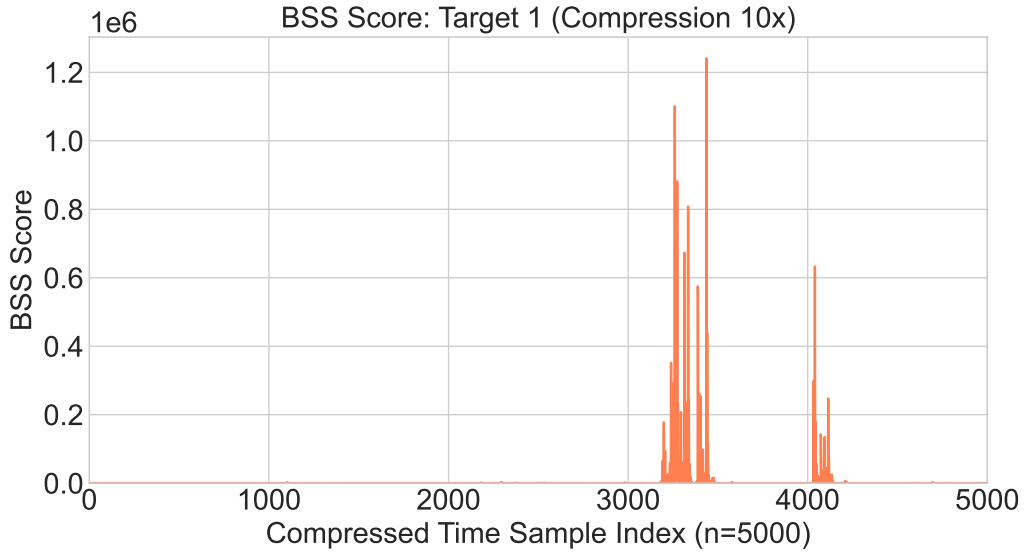


Figure 2: BSS score distribution for target coefficient 1.

The objective is to learn a mapping from these reduced traces to the corresponding leakage classes.

To this end, we employ a Random Forest classifier, a robust ensemble learning method that combines multiple decision trees to model complex, non-linear relationships. Given the profiling dataset $\{(\tilde{\mathbf{x}}_j, y_j)\}_{j=1}^{N_p}$, the classifier learns a function f_θ that approximates the posterior distribution:

$$f_\theta(\tilde{\mathbf{x}}) \approx P(y \mid \tilde{\mathbf{x}}), \tag{10}$$

where $y \in \{0, \ldots, 8\}$ corresponds to the Hamming Weight class.

Prior to training, the PoI features are standardized using the mean and standard deviation computed from the profiling set:

$$\tilde{\mathbf{x}}_j^{(s)} = \frac{\tilde{\mathbf{x}}_j - \boldsymbol{\mu}}{\boldsymbol{\sigma}}, \tag{11}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ denote the feature-wise mean and standard deviation over the profiling data. The same normalization parameters are applied to the attack traces to ensure consistency between training and inference. The classifier is trained on the standardized profiling data to predict the 9-class Hamming Weight labels. For each input trace, the trained model outputs a probability distribution over all classes, rather than a single deterministic prediction. These probabilistic outputs are essential for the subsequent attack phase, where they are used to compute likelihood-based scores for different key hypotheses. Algorithm 4 summarizes the training procedure. The profiling traces are first projected onto the selected PoIs, followed by feature standardization. The resulting normalized features are then used to train the classifier, which is subsequently applied to the attack traces to obtain class probability estimates. For each attack trace $\mathbf{x}_j$, the trained model outputs class probabilities $P(c \mid \tilde{\mathbf{x}}_j)$ for all $c \in \{0, \ldots, 8\}$. These probabilities are subsequently used in the log-likelihood computation for key hypothesis evaluation.

4.6 Key Recovery Strategy

In the attack phase, the goal is to recover the secret key by evaluating different key hypotheses using the trained classifier. For each attack trace $\mathbf{x}_j$, the classifier outputs

Algorithm 4: Classifier Training on Profiling Traces

Input: Trace matrix $T \in \mathbb{R}^{N \times d}$, PoI indices P, labels y, number of profiling traces N_p

Output: Trained classifier Clf, scaled attack features $X_{\text{att}}^{(s)}$

$X_{\text{train}} \leftarrow T[0 : N_p - 1,\ P]$;

$X_{\text{att}} \leftarrow T[N_p : N,\ P]$;

$\boldsymbol{\mu} \leftarrow \text{MEAN}(X_{\text{train}}, \text{axis} = 0)$;

$\boldsymbol{\sigma} \leftarrow \text{STD}(X_{\text{train}}, \text{axis} = 0)$;

$X_{\text{train}}^{(s)} \leftarrow \dfrac{X_{\text{train}} - \boldsymbol{\mu}}{\boldsymbol{\sigma}}$;

$X_{\text{att}}^{(s)} \leftarrow \dfrac{X_{\text{att}} - \boldsymbol{\mu}}{\boldsymbol{\sigma}}$;

Clf $\leftarrow$ RandomForestClassifier;

FIT_MODEL(Clf, $X_{\text{train}}^{(s)}, y$);

return *Clf*, $X_{att}^{(s)}$;

probabilities $P(c \mid \mathbf{x}_j)$ for all 9 Hamming Weight classes. Given a key guess k_g, we compute the corresponding intermediate value using the known input b_j, where $B = \{b_j\}_{j=1}^{N_a}$ denotes the set of input coefficients corresponding to each attack trace:

$$val = \texttt{fqmul}(k_g, b_j) \tag{12}$$

From this value, we extract the most significant byte (MSB) and compute its Hamming Weight to obtain the hypothetical class $c_j(k_g)$.

To evaluate a key guess, we sum the log-probabilities of these predicted classes across all attack traces:

$$Score(k_g) = \sum_{j=1}^{N_a} \log P\left(c_j(k_g) \mid \mathbf{x}_j\right) \tag{13}$$

This score reflects how well the key guess explains the observed leakage. The key with the highest score is selected as the recovered key.

Since the secret is 16 bits, we perform an exhaustive search over all 2^{16} possible key values. In addition, we rank all key guesses based on their scores to evaluate the attack performance. A rank of 1 indicates successful key recovery. Algorithm 5 summarizes the key recovery procedure based on log-likelihood aggregation. The algorithm begins by computing the matrix of log-probabilities $P_{\log}$ from the classifier outputs, where each entry $P_{\log}[j, c] = \log P(c \mid \mathbf{x}_j)$ corresponds to the log-probability of class c for the j-th attack trace. It then performs an exhaustive search over all possible 16-bit key candidates $k \in \{0, \ldots, 65535\}$. For each key hypothesis, a cumulative score is initialized and updated by iterating over all attack traces. For a given trace index j, the intermediate value is computed as $val = \texttt{FQMUL}(k, b_j)$, from which the most significant byte is extracted and mapped to a leakage class $c = \text{HW}(\text{MSB}(val))$. The score for the key hypothesis is incremented by adding the corresponding log-probability $P_{\log}[j, c]$, effectively measuring how well the predicted leakage aligns with the observed trace. After processing all traces, the total score represents the log-likelihood of the key candidate. The algorithm keeps track of the key k_{best} with the highest accumulated score and returns it as the recovered key.

Algorithm 5: Key Recovery using Log-Likelihood

Input: Classifier Clf, attack traces $X_{\text{att}} = \{\mathbf{x}_j\}_{j=1}^{N_a}$, inputs $B = \{b_j\}_{j=1}^{N_a}$
Output: Recovered key k_{best}
$P_{\log} \leftarrow \log(\text{Clf.PREDICT\_PROBA}(X_{\text{att}}))$;
$best\_score \leftarrow -\infty$;
for $k \leftarrow 0$ **to** 65535 **do**
  $score \leftarrow 0$;
  for $j \leftarrow 0$ **to** $N_a - 1$ **do**
    $val \leftarrow \text{FQMUL}(k, B[j])$;
    $msb \leftarrow (val \gg 8)\&0xFF$;
    $c \leftarrow \text{HW}(msb)$;
    $score \leftarrow score + P_{\log}[j, c]$;
  if $score > best\_score$ **then**
    $best\_score \leftarrow score$;
    $k_{\text{best}} \leftarrow k$;
return k_{best};

5 Experimental Evaluation

This section presents the experimental setup and evaluates the effectiveness of the proposed attack on the Kyber side-channel dataset.

5.1 Experimental Setup

All experiments in this work were conducted using the publicly available Kyber side-channel dataset described earlier. The traces were collected from an STM32F3 microcontroller, which is based on the ARM Cortex-M4 architecture, a common embedded platform in real-world deployments. This makes the evaluation relevant for practical attack scenarios rather than purely theoretical analysis.

For the implementation, we used a standard Python-based environment with commonly used machine learning libraries. followed by feature selection using the ANOVA-like BSS metric. This significantly reduced the dimensionality of the data and allowed efficient training of the classifier. We then split the data into profiling and attack sets using different ratios(test-train split), depending on the experiment. Feature normalization was performed using statistics computed only from the profiling set and then applied consistently to both profiling and attack traces. Overall, the setup ensures that the evaluation reflects a realistic profiling attack scenario, where the attacker has access to labelled traces for training and separate traces for key recovery.

5.2 Key Recovery Results

The effectiveness of the proposed attack is demonstrated through successful recovery of the secret subkeys $a[0]$ and $a[1]$. Using a relatively small number of traces, we are able to achieve Rank-1 recovery, meaning that the correct key consistently obtains the highest log-likelihood score among all candidates. In our experiments, We consider a subset of 1000 traces (800 profiling and 200 attack traces) for evaluation. Under this configuration, both target keys were successfully recovered at rank 1. As shown in Table 2, the log-likelihood scores clearly distinguish the correct key from all other candidates, with a noticeable gap between the top-ranked guess and the rest. This gap is important, as it indicates strong confidence in the recovered key.

Table 2: Top 10 key guesses and their log-likelihood scores (800 profiling traces, 200 attack traces) for poi=200.

Target: a[0] (True = 558)		**Target: a[1] (True = 17)**	
Guess	LogLikelihood	Guess	LogLikelihood
558	**-153.1047**	**17**	**-181.0522**
62765	-221.0199	3346	-216.6137
3887	-221.5682	62224	-248.8193
7216	-287.5244	58895	-316.1385
59436	-322.7492	55566	-384.6001
56107	-390.5059	6675	-386.4654
10545	-423.2735	10004	-423.4791
52778	-458.4242	52237	-452.7894
13874	-490.9208	13333	-491.1058
17203	-524.5801	16662	-524.8121

The dataset was randomly shuffled prior to splitting to ensure independence between profiling and attack sets. A more detailed view of the attack performance can be observed from the rank versus number of traces plots for both $a[0]$ and $a[1]$. Initially, when only a small number of traces are used, the rank of the correct key fluctuates between 2 and 3. This is expected, as the model does not yet have sufficient information to confidently separate the correct key from close competitors. However, as the number of traces increases, the rank quickly improves. Around 750 total traces, the rank stabilizes at 1 and remains consistent thereafter. This behavior is clearly visible in both plots, where the curves converge and flatten at rank 1. Once this point is reached, additional traces do not significantly change the ranking, indicating that the attack has already extracted enough information to reliably identify the correct key. This result highlights the efficiency of the proposed method. Achieving stable Rank-1 recovery with approximately 150 attack traces demonstrates that the combination of BSS-based feature selection and Random Forest classification is highly effective. The comparison in Table 3 highlights the trade-offs between classical and machine learning-based side-channel attacks. Correlation Power Analysis (CPA) relies on a linear leakage model and therefore requires a larger number of traces to achieve reliable key recovery, especially when the leakage is weak or non-linear. Template attacks, on the other hand, achieve significantly lower trace complexity (approximately 30–40 traces) by constructing Gaussian models of leakage combined with dimensionality reduction techniques such as LDA. However, these approaches rely on strong assumptions about the leakage distribution and require careful parameter estimation.

In contrast, the proposed method adopts a fully data-driven approach that does not assume any specific leakage distribution. By combining ANOVA-like BSS feature selection with a Random Forest classifier, the model is able to capture more complex and non-linear leakage characteristics. While the trace complexity is higher than template attacks, the proposed method provides greater flexibility and robustness, particularly in scenarios where classical assumptions about leakage may not hold. Furthermore, the consistent Rank-1 recovery achieved with approximately 150 attack traces demonstrates that the approach remains highly effective in practice.

5.3 Key Equivalence Observation

An interesting observation from our results is the presence of key equivalence classes, often referred to as a "ghost key" phenomenon. Instead of a single dominant key candidate, we observe that several high-ranking guesses share a specific modular relationship with

Table 3: Comparison of key recovery performance(While template attacks achieve lower trace complexity under strong Gaussian assumptions, the proposed method provides a flexible and assumption-free alternative that can capture non-linear leakage characteristics.)

Method	Model Type	Traces for Rank-1	Remarks
CPA [10]	Linear	∼30–250	Varies across targets
Template Attack [10]	Gaussian	∼30–40	Assumes Gaussian leakage
Proposed (RF + BSS)	Non-linear ML	∼150	Data-driven, no assumptions

the true key with respect to modulo `KYBER_Q` = 3329. For the first target $a[0]$, the top-ranked guesses are all congruent to, either 558 mod 3329 or 2843 mod 3329. Similarly, for the second target $a[1]$, the leading candidates are congruent to 17 mod 3329 or 2302 mod 3329. The difference between these equivalent representatives is consistent when we replicated both correlation power analysis and template attack, equal to 2285 in both cases. This suggests that the leakage produced by the `fqmul` operation is not uniquely determined by a single key value, but instead exhibits symmetry with respect to certain modular transformations. In particular, the leakage of `fqmul`(k, b) appears to be strongly correlated with that of `fqmul`$(k + 2285, b)$. From a machine learning perspective, this indicates that the model has learned structural properties of the underlying finite field arithmetic rather than memorizing individual key values. While this behavior does not prevent successful key recovery (since the correct key still ranks first), it reveals an important limitation in the distinguishability of certain key candidates. Understanding these equivalence classes is important for both attack analysis and countermeasure design, as it highlights how algorithmic structure can influence leakage behavior.

5.4 Attack Efficiency (Key Rank)

To ensure our profiling and attack sets were independent and not biased by contiguous trace collection, we *randomly shuffled the dataset* before splitting. Using a standard *80/20 split* (80% profiling, 20% attack) on the shuffled data, a stable Rank-1 recovery was achieved with approximately *750 total traces* for a_1(*150 attack traces*).we can reduce no.of total traces required a bit by changing split to *70/30* or *60/40* . We note that this rank can fluctuate, temporarily increasing before settling at 1.

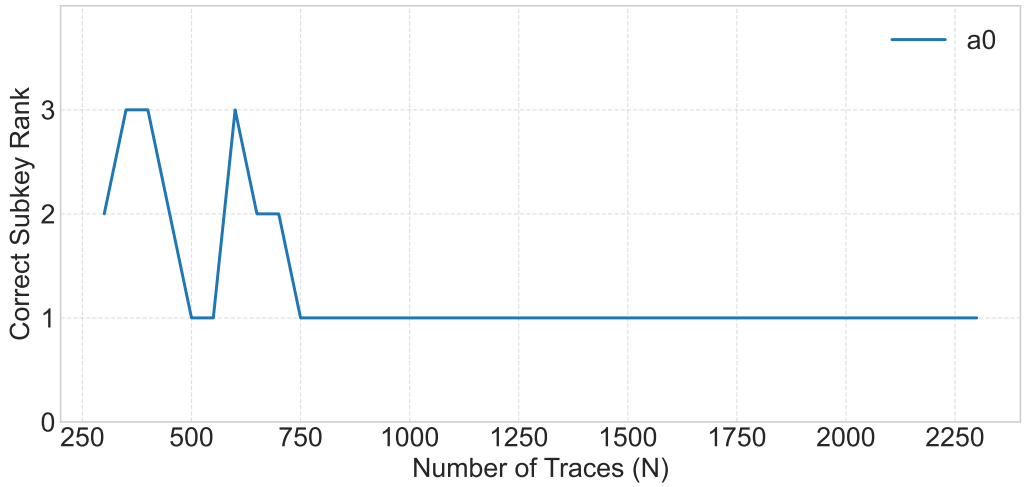


Figure 3: Key rank vs number of total traces for a[0]

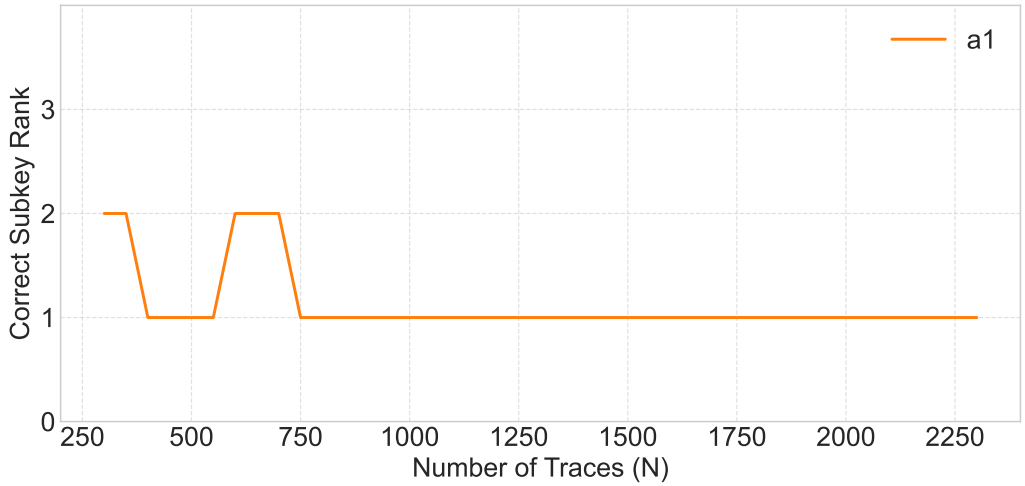


Figure 4: Key rank vs number of total traces for a[1]

Tie-inclusive rank: We adopt Tie inclusive ranking commonly used in side channel analysis, where the rank of key is no.of keys whose log-likelihood score is greater than or equal to that of true key's log-likelihood score

- If the true guess ties with the unique top-scoring guess, $R = 1$.
- If t guesses tie for the top score and the true guess is among them, $R = t$.
- If several guesses have scores strictly greater than the true score, R counts them plus any guesses tied with the true score.

Figures 3 and 4 illustrates the rank with respect to number of total traces. We observe that the key rank fluctuates initially before converging to a stable rank of 1, demonstrating that an insufficient number of traces cannot uniquely identify the secret key. Ultimately, the results indicate that a minimum of 750 total traces (yielding 150 attack traces) is required to reliably recover the correct keys. This demonstrates that the proposed approach can efficiently exploit leakage with significantly fewer traces compared to traditional methods.

Limitations: The proposed approach assumes access to labeled profiling traces, which may not always be available in practical attack scenarios. Additionally, the evaluation is performed on an unprotected implementation; the effectiveness of the method against countermeasures such as masking or hiding remains to be explored.

6 Conclusion and Future Work

In this work, we presented a machine learning-based profiling side-channel attack targeting the `basemul/fqmul` operations in Kyber. By modeling the leakage using a 9-class Hamming Weight representation of the most significant byte, and combining ANOVA-like feature selection with a Random Forest classifier, we demonstrated an effective and practical attack framework. Our results show that the proposed approach achieves Rank-1 key recovery with approximately 150 attack traces (750 total traces), highlighting the effectiveness of simple machine learning models when combined with appropriate feature selection and probabilistic scoring. The observed consistency of Points of Interest with prior work further validates the reliability of the leakage exploited in this study. Overall, this work

reinforces that even standard-compliant implementations of post-quantum cryptographic schemes remain vulnerable to side-channel attacks, emphasizing the need for robust implementation-level protections.

Several directions can be explored to extend this work. First, the use of more advanced models such as deep learning architectures could be investigated to capture more complex leakage patterns. Second, evaluating the proposed attack on protected implementations (e.g., masking or hiding countermeasures) would provide deeper insights into its practical applicability. Third, extending the analysis to other operations within Kyber or to other post-quantum schemes would help assess the generality of the approach. Finally, developing lightweight countermeasures tailored for embedded implementations remains an important direction for securing real-world deployments.

Acknowledgement

This work was supported by the Anusandhan National Research Foundation (ANRF), Government of India, under the Prime Minister Early Career Research Grant (PM-ECRG), Grant No. ANRF/ECRG/2024/003984.

References

[1] Amin Abdulrahman, Vincent Hwang, Matthias J Kannwischer, and Amber Sprenkels. Faster kyber and dilithium on the cortex-m4. In *International Conference on Applied Cryptography and Network Security*, pages 853–871. Springer, 2022.

[2] Gorjan Alagic, Gorjan Alagic, Daniel Apon, David Cooper, Quynh Dang, Thinh Dang, John Kelsey, Jacob Lichtinger, Yi-Kai Liu, Carl Miller, et al. Status report on the third round of the nist post-quantum cryptography standardization process. 2022.

[3] Joppe Bos, Leo Ducas, Eike Kiltz, T Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, and Damien Stehle. Crystals - kyber: A cca-secure module-lattice-based kem. In *2018 IEEE European Symposium on Security and Privacy (EuroSP)*, pages 353–367, 2018.

[4] Joppe W. Bos, Marc Gourjon, Joost Renes, Tobias Schneider, and Christine van Vredendaal. Masking kyber: First- and higher-order implementations. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2021(4):173–214, Aug. 2021.

[5] Eric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. In *International workshop on cryptographic hardware and embedded systems*, pages 16–29. Springer, 2004.

[6] Suresh Chari, Josyula R Rao, and Pankaj Rohatgi. Template attacks. In *International workshop on cryptographic hardware and embedded systems*, pages 13–28. Springer, 2002.

[7] Yanning Ji, Ruize Wang, Kalle Ngo, Elena Dubrova, and Linus Backlund. A side-channel attack on a hardware implementation of crystals-kyber. In *2023 IEEE European Test Symposium (ETS)*, pages 1–5. IEEE, 2023.

[8] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power analysis attacks: Revealing the secrets of smart cards*. Springer, 2007.

[9] Dustin Moody. Nist pqc standardization update. *National Institute of Standards and Technology*, pages 2021–10, 2021.

[10] S. Nkotto. A template sca attack on the kyber/ml-kem pair-pointwise multiplication. Cryptology ePrint Archive, Paper 2025/1577, 2025.

[11] Prasanna Ravi, Anupam Chattopadhyay, Jan Pieter D'Anvers, and Anubhab Baksi. Side-channel and fault-injection attacks over lattice-based post-quantum schemes (kyber, dilithium): Survey and new results. *ACM Transactions on Embedded Computing Systems*, 23(2):1–54, 2024.

[12] Prasanna Ravi, Dirmanto Jap, Shivam Bhasin, and Anupam Chattopadhyay. Invited paper: Machine learning based blind side-channel attacks on pqc-based kems - a case study of kyber kem. In *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, pages 01–07, 2023.

[13] A. Rezaeezade et al. Side-channel power trace dataset for kyber pair-pointwise multiplication on cortex-m4. Cryptology ePrint Archive, Paper 2025/811, 2025.

[14] L. Weissbart, S. Pk, and L. Batina. One trace is all it takes: Machine learning-based side-channel attack on eddsa. Cryptology ePrint Archive, Paper 2019/358, 2019.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 134–148.

Certificateless Identification Schemes: A Comprehensive Survey

Amani Balqis Johal[1], Sook-Chin Yip[2], Ji-Jian Chin[3] and Uzma Jafar[4]

[1] Faculty of Computing & Informatics, Multimedia University, Cyberjaya, 63100, Selangor, Malaysia amani.balqis.johal@student.mmu.edu.my
[2] School of Artificial Intelligence and Robotics, Xiamen University Malaysia, Sepang, 43900, Selangor, Malaysia sookchin.yip@xmu.edu.my
[3] Faculty of Science and Engineering, University of Plymouth, Plymouth, United Kingdom ji-jian.chin@plymouth.ac.uk
[4] Faculty of Computing & Informatics, Multimedia University, Cyberjaya, 63100, Selangor, Malaysia uzma.jafar@mmu.edu.my

Abstract.
Certificateless public key cryptography (CL-PKC) eliminates the need for certificates associated with Public Key Infrastructure (PKI) while addressing the key escrow problem inherent in identity-based cryptography. This survey presents a comprehensive review of existing certificateless identification (CLI) schemes. CLI remains significantly underexplored compared to other certificateless primitives. Existing CLI schemes are categorized into three primary classes: pairing-based, pairing-free, and post-quantum. The survey systematically compares existing CLI schemes with respect to security models, adversarial types, cryptographic assumptions, the tightness of security reductions, and computational cost. The findings reveal that CLI remains notably underexplored, particularly in pairing-free ECC-based constructions, practical implementation, and standard-model security. Furthermore, tight security reductions and post-quantum security remain insufficiently explored in existing CLI schemes. The survey concludes that further research is needed to improve the efficiency, security, and practical deployment of CLI schemes. In particular, there is a need for pairing-free ECC-based CLI constructions with stronger security guarantees, empirical evaluation, and real-world implementation. Finally, several open problems and future research directions are identified to facilitate further advancements in CLI cryptography.

Keywords: Certificateless Identification (CLI) · Certificateless Cryptography · Public Key Cryptography · Pairing-free Cryptography

1 Introduction

Traditional Public Key Infrastructure (PKI) requires digital certificates to bind public keys to identities, which introduces significant certificate management overhead. To address these limitations, Shamir [19] introduced identity-based cryptography (IBC), where a user's public key is derived directly from their identity, thereby eliminating the need for certificates, but it suffers from the key escrow problem, as the key generation centre (KGC) is responsible for generating all users' private keys.

To overcome this issue, Al-Riyami and Paterson [1] proposed certificateless public key cryptography (CL-PKC). In this framework, KGC generates a partial private key for each user, and then the user completes the private key by combining their secret value with the KGC's partial private key. This removes key escrow while avoiding the need for certificate management. Based on this framework, certificateless identification (CLI) schemes have

been extensively developed. In contrast, CLI remains substantially underexplored. This survey focuses exclusively on CLI schemes.

These schemes are designed to resist two types of adversaries: Type-1 adversaries, representing malicious users who do not possess the master secret key but are capable of replacing public keys, and Type-2 adversaries, representing a malicious key generation centre (KGC) that has access to the master secret key. Achieving simultaneous security against both adversarial models remains a nontrivial challenge. Many existing constructions rely on bilinear pairings, which are among the most computationally intensive cryptographic operations, resulting in high computational overhead.

CLI schemes are particularly well-suited for authentication in resource-constrained, certificate-free environments, where efficient identity verification is essential. Potential application domains include Internet of Things (IoT) devices, wireless sensor networks, smart cards, mobile authentication, and vehicle-to-everything (V2X) communications. In such environments, eliminating certificate management while maintaining resistance against key escrow and public key replacement attacks makes CLI an attractive authentication solution.

This paper presents a comprehensive review of CLI schemes, which have received comparatively less attention in the literature. This survey makes the following contributions. First, we review and categorise existing CLI schemes based on their cryptographic techniques. Second, we provide a systematic comparison of these schemes with respect to pairing dependence, mathematical assumptions, security models, tightness of security reductions, and adversary models (Type-1 and Type-2), as well as their computational costs. Finally, this survey identifies existing research gaps and proposes future directions. Overall, this survey aims to provide a clear understanding of the evolution and current state of certificateless cryptography.

2 Preliminaries

2.1 Bilinear Pairings

A bilinear pairing is a map $e : G_1 \times G_2 \to G_T$ defined over cyclic groups of prime order, satisfying bilinearity, non-degeneracy, and computability. Despite their flexibility, bilinear pairing operations are computationally expensive, motivating the development of pairing-free certificateless schemes.

2.2 Security Assumptions

The security of CLI schemes relies on the following standard hardness assumptions:

- **Computational Diffie–Hellman Problem (CDHP):** Given (P, aP, bP) for $a, b \in \mathbb{Z}_q$, compute abP. The One-More Computational Diffie–Hellman Problem (OMCDHP) extends the CDH assumption by allowing an adversary limited access to a computational oracle while requiring the computation of additional CDH instances. It is commonly used to demonstrate security against active, concurrent adversaries in pairing-based CLI schemes.

- **Discrete Logarithm Problem (DLP):** Given (g, g^x) in a cyclic group, compute x. The One-More Discrete Logarithm Problem (OMDLP) extends the DLP assumption in a one-more-oracle setting. It is commonly used to prove active and concurrent security for pairing-free CLI schemes.

- **Supersingular Isogeny Problem:** Given two supersingular elliptic curves, compute an isogeny between them.

3 Certificateless Identification

3.1 CLI Framework

The standard certificateless framework introduced by Al-Riyami and Paterson [1] consists of a set of polynomial-time algorithms, including Setup, Set-User-Key, Partial-Private-Key-Extract, Set-Private-Key, and Set-Public-Key.

1. **Setup**: The KGC takes a security parameter 1^k as input and generates the master public key MPK and master secret key MSK. The MPK is published, while the MSK is kept secret by the KGC.

2. **Set-User-Key**: The user selects a secret value SV based on the system parameters and their identity ID.

3. **Partial-Private-Key-Extract**: The KGC takes MPK, MSK, and the user's identity ID as input, and outputs the partial private key PPK.

4. **Set-Private-Key**: The user combines the partial private key PPK with the secret value SV to obtain the full private key USK.

5. **Set-Public-Key**: The user generates the public key UPK using the system parameters and the secret value SV.

6. **Identification–Protocol**: An interactive protocol between a Prover and a Verifier. The protocol proceeds as a three-move honest-verifier zero-knowledge (HVZK) proof-of-knowledge protocol consisting of the following steps:

1. **Commitment**: The prover sends a commitment (CMT) to the verifier.

2. **Challenge**: The verifier sends a challenge (CHL) to the prover.

3. **Response**: The prover sends a response (RSP) to the verifier. The verifier then decides whether to accept or reject the authentication request.

Figure 1 illustrates the overall framework of a generic CLI scheme, including its key generation and identification phases.

3.2 Security Model

In the CLI security model, adversaries are classified into two main types, namely Type-1 and Type-2, which capture both external and internal threats. A Type-1 adversary does not possess the master secret key but can replace public keys. In contrast, a Type-2 adversary possesses the master secret key but cannot replace the target user's public key.

Beyond these two adversarial types, adversaries can further be categorised based on their capabilities [16]:

1. A **Normal adversary** can only perform identification queries using the original public key.

2. A **Strong adversary** may replace a user's public key but must provide the corresponding secret value associated with the replaced key.

3. A **Super adversary** is the most powerful, as it can perform identification queries even after replacing the public key without needing to provide the corresponding secret value.

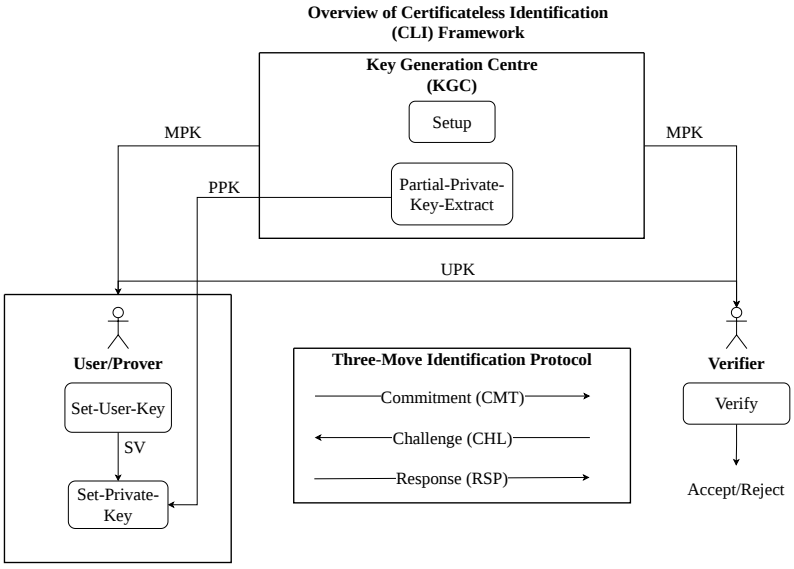


Figure 1: Overview of the certificateless identification (CLI) framework.

We define the security games for CLI techniques against Type-1 and Type-2 impersonators, following the framework established by Chin et al. [10].

Game 1: Type–1 impersonator $\mathcal{I}_1$. The security game between the challenger $\mathcal{C}$ and $\mathcal{I}_1$ proceeds as follows:

- **Setup:** $\mathcal{C}$ runs Setup and gives the system parameters to $\mathcal{I}_1$. The MSK is kept secret.
- **Learning Phase (Phase 1):** $\mathcal{I}_1$ may adaptively issue the following queries:
  1. **ExtrPartKey**(ID). $\mathcal{C}$ returns the partial private key of ID.
  2. **ExtrFullKey**(ID). $\mathcal{C}$ returns the full private key of ID.
  3. **RequestPK**(ID). $\mathcal{C}$ returns the current public key of ID.
  4. **ReplacePK**(ID, UPK'). $\mathcal{I}_1$ replaces the public key of ID with UPK'.
  5. **Identification**(ID). $\mathcal{C}$ plays the role of prover while $\mathcal{I}_1$ plays the role of a cheating verifier.:
     (a) **Normal-Type**: Identification query is not allowed on replaced ID.
     (b) **Strong-Type**: Identification query is allowed on replaced ID only if the $\mathcal{I}_1$ supplies a consistent SV.
     (c) **Super-Type**: Identification query is allowed on replaced ID without supplying the SV.
- **Challenge Phase (Phase 2):** $\mathcal{I}_1$ outputs a target identity ID^* and switch their role where $\mathcal{I}_1$ acts as cheating prover and $\mathcal{C}$ acts as verifier. $\mathcal{I}_1$ wins if the verifier accepts the authentication attempt.

Game 2: Type-2 impersonator, $\mathcal{I}_2$. The security game between the challenger $\mathcal{C}$ and $\mathcal{I}_2$ proceeds as follows:

- **Setup:** $\mathcal{C}$ runs Setup and gives the system parameters and MSK to $\mathcal{I}_2$.
- **Learning Phase (Phase 1):** $\mathcal{I}_2$ may adaptively issue the following queries:

1. **ExtrFullKey**(ID). $\mathcal{C}$ returns the full private key of ID.
2. **RequestPK**(ID). $\mathcal{C}$ returns the current public key of ID.
3. **ReplacePK**(ID, UPK'). $\mathcal{I}_2$ replaces the public key of ID. $\mathcal{I}_2$ is not allowed to replace the target ID. Otherwise, the game is trivial.
4. **Identification**(ID). Same as in Game 1.

- **Challenge Phase (Phase 2):** Same as in Game 1. $\mathcal{I}_2$ wins if the verifier accepts.

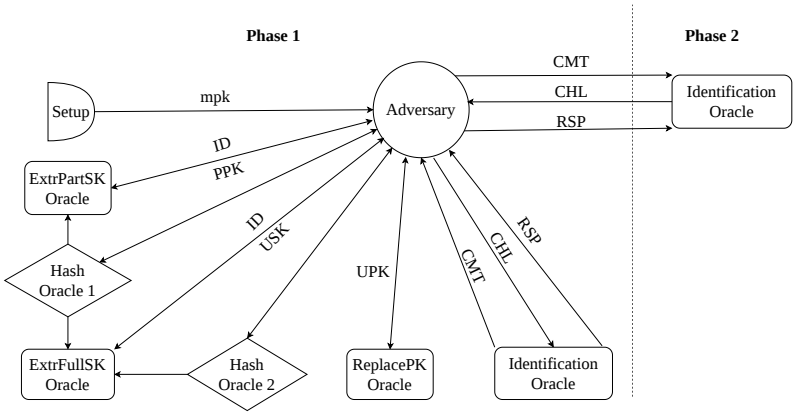


Figure 2: Security game interaction between the challenger and an adversary in certificateless identification schemes, illustrating oracle access and identification queries.

Figure 2 illustrates the security game interaction between the challenger and the adversary in certificateless identification schemes. The adversary is allowed to query various oracles, including Extract, RequestPK, ReplacePK, and Identification. Note that the distinction between Type-1 and Type-2 adversaries lies in the restrictions imposed on these queries. Type-2 adversaries *do not require ExtrPartKey queries* due to possessing MSK, which enables them to generate PPK themselves.

In addition, adversaries can be classified based on their interaction capabilities. A passive adversary can only observe protocol executions, whereas an active adversary can interact with honest participants and attempt impersonation attacks. A stronger setting is the concurrent attack model, where multiple protocol sessions are executed simultaneously.

4 Survey Methodology

This survey follows the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines to ensure a systematic and transparent review process. The paper selection process consisted of four phases: identification, screening, eligibility, and inclusion.

Identification: Relevant papers were collected from databases such as IEEE Xplore, Google Scholar, ScienceDirect, and SpringerLink using keywords related to certificateless cryptography, including "certificateless identification", "CLI", "pairing-free identification", as well as related certificateless primitives to ensure that no relevant CLI studies were overlooked. The search was limited to papers published between 2003 and 2024. Due to the limited body of research on certificateless identification (CLI) schemes, only a few potentially relevant studies were identified.

Screening: Duplicate studies, non-English publications, and studies unrelated to certificateless identification were excluded based on their titles and abstracts.

Eligibility: The remaining studies were evaluated based on their relevance to certificateless identification (CLI), security models, efficiency analysis, and cryptographic constructions. Only studies proposing novel CLI schemes accompanied by formal security analyses were included. Studies focusing on other certificateless primitives, such as certificateless signatures, encryption, signcryption, authentication, and key agreement, were excluded because they fall outside the scope of this survey.

Inclusion: Following the identification, screening, and eligibility assessment, the studies that satisfied the predefined inclusion criteria were selected for the final analysis.

The final set of seven studies represents all currently available CLI schemes that satisfied the predefined inclusion criteria during the review period.

The study selection process is summarised in the PRISMA flow diagram shown in Figure 3. The diagram illustrates the identification, screening, eligibility assessment, and final inclusion of studies considered in this survey.

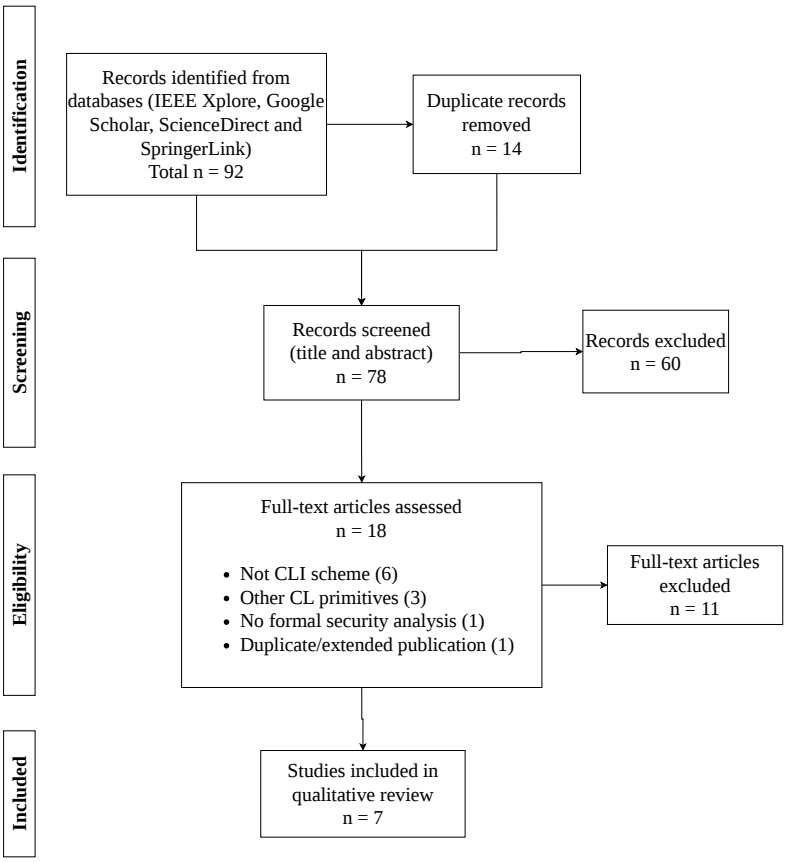


Figure 3: PRISMA flow diagram illustrating the study selection process.

5 Literature Review

Existing CLI schemes can be categorised into three primary types based on their foundational cryptographic constructions: pairing-based schemes, pairing-free schemes, and post-quantum schemes. This classification illustrates the advancement of CLI in tackling efficiency and security issues.

5.1 Certificateless Identification (CLI) Schemes

Al-Riyami and Paterson [1] introduced the first certificateless public key cryptography (CL-PKC) framework. However, subsequent studies by Huang et al. [16] and Au et al. [2] demonstrated that the scheme is vulnerable to public key replacement attacks and malicious-but-passive KGC (Type-2) attacks. Nevertheless, this work laid the foundation for subsequent developments in certificateless cryptography, particularly in signature and identification schemes. Early CLI constructions predominantly utilised bilinear pairings to attain robust security. The first CLI construction was proposed by Dehkordi and Alimoradi in [14], which utilised bilinear pairings to achieve CLI. This work marked the beginning of research on CLI schemes. However, Chin et al. [9] later demonstrated that the scheme is insecure against both Type-1 and Type-2 adversaries

To address these limitations, Chin et al. [10] formalised the notion of CLI and proposed a concrete scheme with security proven in the random oracle model (ROM). Subsequently, Chin et al. [11] extended this approach by constructing a CLI scheme in the standard model (SM), providing stronger security guarantees without relying on random oracles. These schemes rely on bilinear pairing operations. To improve efficiency, Chin et al. [12] extended their earlier work by proposing a pairing-free CLI scheme that eliminates bilinear pairings. Instead of pairings, their construction relies on exponentiation operations in a finite-field discrete logarithm (DLP) group. The scheme achieves Super-Type security against both Type-1 and Type-2 adversaries under passive attacks, as well as under active and concurrent attacks, with proofs based on the discrete logarithm and one-more discrete logarithm assumptions. The introduction of pairing-free CLI marked a significant transition toward more efficient constructions. However, research in this direction remains limited, with only one pairing-free CLI scheme having been reported in the literature.

An alternative approach of CLI is the ECQV-IBI scheme proposed by Braeken et al. [4]. Although ECQV-IBI is not a certificateless identification (CLI) scheme, it is discussed as a related identification paradigm because it addresses the same identification problem using implicit certification rather than the certificateless paradigm. This scheme combines identity-based identification with implicit certification. Unlike CLI, ECQV-IBI does not require users to publish a separate public key. Instead, the public key is reconstructed from the implicit certificate during the identification protocol. Consequently, public key replacement attacks are eliminated, and the security model does not require a ReplacePK oracle. The scheme is proven secure under the discrete logarithm (DLP) assumption using the Reset Lemma. It operates over elliptic curve cryptography (ECC) and has been benchmarked on resource-constrained devices. However, users cannot update their key material without obtaining a new certificate from the trusted authority, whereas CLI allows independent key updates. Therefore, the choice between ECQV-IBI and CLI depends on whether flexibility (CLI) or efficiency (ECQV-IBI) is prioritised.

Beyond ECQV-IBI, recent identity-based identification (IBI) research has also explored tight security reductions. Chia et al. [7] proposed a pairing-free IBI scheme with tight security against active and concurrent adversaries. Although their work focuses on the identity-based setting rather than certificateless identification, it demonstrates that tight security reductions are achievable for identification protocols without relying on rewinding techniques. These results provide valuable design insights for future CLI constructions.

With the emergence of quantum computing, there has been growing interest in developing post-quantum certificateless identification (CLI) schemes. Daghigh and Khodakaramian [13] proposed a CLI scheme based on isogenies between elliptic curves, incorporating quantum-resistant properties and batch verification techniques inspired by Chin et al.'s work [12]. However, the scheme by Daghigh and Khodakaramian lacks a formal security proof and is supported only by illustrative examples using toy values. More recently, Huang et al. [15] proposed a CLI scheme based on supersingular isogenies, integrating the De Feo–Jao–Plût identification scheme and SQISign to achieve post-quantum security.

The scheme can only achieve Normal-Type security, as the adversarial capabilities are restricted. Post-quantum CLI schemes remain in early stages of development.

6 Comparative Analysis

6.1 Security Analysis

Table 1: Security comparison of CLI schemes

Scheme	Year	Model	Assumption	Reduction	Type-1 (A/C)	Type-2 (A/C)
Pairing-based						
Dehkordi and Alimoradi [14]	2013	N/A	N/A	N/A	Insecure [9]	Insecure [9]
Chin et al. [10]	2013	ROM	OMCDHP	Non-tight	Strong	Strong
Chin et al. [11]	2014	SM	OMCDHP	Non-tight	Strong	Strong
Pairing-free						
Chin et al. [12]	2015	ROM	OMDLP	Non-tight	Super	Super
Post-Quantum						
Daghigh and Khodakaramian [13]	2019	N/A	Supersingular Isogeny Problem	N/A	N/A	N/A
Huang et al. [15]	2022	ROM	Supersingular Isogeny Problem	Non-tight	Normal	Normal

Note: A/C denotes the active and concurrent attack models. Passive security is not reported separately because the original works primarily analyse the stronger active and concurrent adversarial settings.

As shown in Table 1, early CLI schemes predominantly relied on pairing-based constructions. More recent works, particularly Chin et al. [12], have shifted towards pairing-free designs due to the high computational cost of bilinear pairings [6]. Unless otherwise stated, the Type-1 and Type-2 security levels reported in Table 1 correspond to the active and concurrent attack models, which represent stronger adversarial settings than passive attacks. Therefore, passive security is not reported separately.

Early CLI constructions were also found to suffer from various security vulnerabilities, as reflected in the tables where the scheme by Dehkordi and Alimoradi [14] was later proven insecure against Type-1 and Type-2 adversaries by Chin et al. [9]. Over time, research has advanced the security of models, with more recent schemes achieving strong, and even super-adversarial, security. Chin et al. [10] proposed the first formal CLI security model and achieved Strong-Type security under active and concurrent attacks. Subsequent work further strengthened the security guarantees by achieving Super-Type security, including security under passive attacks [12]. Nevertheless, simultaneously achieving Super-Type security against both Type-1 and Type-2 adversaries remains a challenging problem.

Another notable observation is that most existing CLI schemes have been proven secure in the random-oracle model (ROM). Nonetheless, the ROM is an idealised abstraction that may not accurately reflect real-world implementations [5]. In particular, security guarantees in the ROM do not necessarily hold when the random oracle is instantiated with a concrete hash function. Although schemes proven under the standard model (SM) offer stronger and more realistic security guarantees, only one existing CLI scheme has been proven in the SM [11], and it introduces additional complexity.

Furthermore, existing schemes rely on non-tight security reductions, often based on techniques such as the reset lemma and the rewinding technique. These approaches

introduce a loss factor in the reduction, potentially requiring larger parameter sizes to achieve equivalent security levels. By contrast, tight security reductions enable more efficient parameter selection and improved performance. Similar efforts have been demonstrated in related cryptographic primitives. For example, variants of the Schnorr and Boneh–Lynn–Shacham (BLS) signature schemes have achieved tight security reductions while maintaining practical efficiency [18, 17]. These results suggest that tight security reductions are achievable in practice and may provide useful design insights for future CLI constructions. However, as observed in Table 1, no CLI scheme currently achieves a tight security reduction.

In terms of underlying assumptions, pairing-based CLI schemes typically rely on the CDHP or the One-More CDHP. The only pairing-free CLI scheme [12] relies on the DLP and the One-More DLP. CLI schemes still face challenges in achieving Super-Type against an active concurrent adversary, even under stronger assumptions. This indicates that security model design is critical beyond the choice of assumptions.

Finally, post-quantum certificateless schemes have attracted increasing attention in recent years. CLI constructions have begun exploring quantum-resistant approaches based on isogeny problems [13, 15]. However, these schemes are still in their early stages. CLI remains significantly underexplored, with only a handful of schemes proposed since 2013.

Despite these promising developments, several operational considerations should also be addressed before CLI schemes can be widely deployed. These include secure user enrolment, key lifecycle management, and public key revocation when users leave the system or their credentials are compromised. Since the Key Generation Centre (KGC) remains responsible for issuing partial private keys, appropriate trust management and accountability mechanisms are also essential. Furthermore, migrating from conventional public key infrastructure (PKI) to certificateless environments requires careful consideration of compatibility with existing authentication infrastructures and deployment procedures. Although these operational aspects have received limited attention in current CLI research, they represent important challenges that should be addressed to facilitate the practical adoption of CLI schemes.

6.2 Efficiency Analysis

Table 2: Notation of computational cost used in CLI schemes

Notation	Description
T_{sm}	Scalar multiplication in an elliptic curve group
T_{pa}	Point addition in an elliptic curve group
T_p	Bilinear pairing operation
T_e	Modular exponentiation
T_m	Multiplication in a finite field
T_{gm}	Group multiplication (e.g., in G_2 or $\mathbb{Z}_p^*$)
T_I	Isogeny computation

The notation used in the computational cost analysis is presented in Table 2. Table 3 presents a comparison of the computational costs of existing CLI schemes. The cost is determined by the number of cryptographic operations in the identification protocol. A clear trend is evident: recent schemes have shifted toward pairing-free constructions, while earlier works predominantly relied on pairings. Pairing-free schemes are generally expected to reduce the overhead associated with bilinear pairing operations. Nevertheless, their actual

Table 3: Computational cost comparison of CLI schemes

Scheme	Year	Prover Cost	Verifier Cost	Total Cost
Pairing-based				
Dehkordi and Alimoradi [14]	2013	$2T_{sm} + T_m + T_{gm}$	$3T_p + T_m + T_e$	$2T_{sm} + 2T_m + T_{gm} + T_e + 3T_p$
Chin et al. [10]	2013	$2T_e$	$2T_{sm} + T_e + 4T_p$	$2T_{sm} + 3T_e + 4T_p$
Chin et al. [11]	2014	$(n+1)T_{gm} + 3T_e + T_p$	$(n+4)T_{gm} + 2T_e + 5T_p$	$(2n+5)T_{gm} + 5T_e + 6T_p$
Pairing-free				
Chin et al. [12]	2015	$T_m + 2T_{gm} + 4T_e$	$3T_{gm} + 5T_e$	$T_m + 5T_{gm} + 9T_e$
Post-quantum				
Daghigh and Khodakaramian [13]	2019	$2T_{sm} + T_I + T_p + 2T_e$	$3T_p$	$2T_{sm} + T_I + 2T_e + 4T_p$
Huang et al. [15]	2022	$T_I + T_m$	$T_I + T_m$	$2T_I + 2T_m$

Note: n denotes the identity hash output length. Hash operations and simple additions are omitted because they incur negligible cost.

performance depends on the number and type of remaining operations, implementation optimisations, security parameters, and hardware platforms.

The pairing-based schemes, such as Dehkordi and Alimoradi [14], Chin et al. [10], and Chin et al. [11], suffer from computational overhead due to the bilinear pairing operation T_p, which is typically 10–20 times slower than an elliptic curve scalar multiplication [6]. These schemes require multiple pairing operations per protocol run, making them less suitable for resource-constrained environments.

In contrast, the pairing-free CLI scheme by Chin et al. [12] entirely removes bilinear pairings. Specifically, unlike most pairing-free certificateless schemes, this scheme uses a finite-field framework (the multiplicative group) and relies on modular exponentiations (T_e) and group multiplications (T_{gm}). Although the scheme requires a total of $T_m + 5T_{gm} + 9T_e$, its operation profile differs from that of pairing-based CLI schemes. For instance, Chin et al. [11] require $6T_p$ in the standard model, whereas the pairing-free scheme eliminates bilinear pairing operations altogether. Among the formally defined CLI schemes considered, Chin et al. [12] therefore appears favourable based on theoretical operation counts. However, because the schemes are constructed over different algebraic settings, a definitive efficiency ranking cannot be established without implementation benchmarks under equivalent security levels and hardware conditions.

No ECC-based pairing-free CLI technique currently exists. This contrasts with CLS, where pairing-free ECC constructions achieve lower computational cost and smaller key sizes. This highlights a notable research gap, since ECC provides reduced key sizes and expedited calculation while maintaining equivalent security levels.

A trade-off between efficiency and security is also evident. Schemes achieving stronger security, particularly in the standard model, incur higher computational cost. For instance, the standard model CLI scheme by Chin et al. [11] requires $6T_p$, whereas the ROM-based scheme by Chin et al. [10] requires $4T_p$. However, neither achieves tight security reduction.

Finally, post-quantum CLI schemes remain limited. Existing isogeny-based constructions by Daghigh et al. [13] and Huang et al. [15] come with high computational costs and lack comprehensive evaluation. For example, Huang et al.' s [15] work is computationally inefficient, as it requires repeated protocol executions to achieve sufficient security levels, resulting in high overall cost.

Although the computational costs reported in Table 3 provide a theoretical comparison, it should be noted that most existing CLI schemes have not been validated through prototype implementations or real-world benchmarking. Consequently, the reported efficiency comparisons remain analytical rather than empirical, highlighting the need for future implementation studies.

7 Research Gap and Future Directions

7.1 Research Gap

Despite substantial progress in certificateless identification (CLI), several important research gaps remain.

First, although certificateless public key cryptography has received considerable attention, research on certificateless identification remains relatively limited. In particular, pairing-free ECC-based CLI schemes are absent. Although Chin et al. [12] eliminates bilinear pairings, it operates in a finite-field setting rather than on elliptic curves. This reveals an important research gap, as ECC is better suited to resource-constrained environments. Furthermore, most CLI schemes have been proven in the random-oracle model, with only limited work in the standard model [11].

Second, CLI remains limited in terms of meaningful practical validation at two levels: evaluation and implementation. At the evaluation level, existing CLI works rely solely on theoretical cost estimations such as counting group operations, exponentiations, and multiplications. At the implementation level, no current CLI system has been developed into a functional prototype. Consequently, there is a lack of practical implementations that reflect real-world deployment scenarios. This limitation restricts the practical assessment of CLI schemes in real-world deployment scenarios. A CLI scheme with no implementation cannot credibly claim suitability for real-world applications or deployment scenarios.

Third, reductions in tight security remain largely underexplored in CLI schemes. Most existing constructions rely on non-tight reductions, which may require larger security parameters to achieve equivalent levels of security. No CLI scheme to date has attempted to address loose security reductions or to develop constructions that strive for tightness.

Fourth, the majority of existing CLI schemes exhibit an over-reliance on strong or non-standard cryptographic assumptions. For example, the One-More CDH or One-More ECDLP problems [14, 10, 11]. These assumptions are considered stronger than standard ones, such as CDH and ECDLP, and their application undermines the security assurances of the respective schemes, as they do not reduce to well-studied hard problems [3]. While this reliance is technically understandable given the interactive nature of identification protocols, which introduce inherent challenges in achieving the same level of security, particularly in concurrent challenge–response models. Nevertheless, it also reveals an important research opportunity. Specifically, there is a need for novel constructions that handle concurrency without non-standard assumptions. This represents a theoretically interesting open problem rather than a limitation of any specific construction in the certificateless cryptography literature.

Finally, post-quantum CLI research remains in its early stages. Only a handful of post-quantum CLI studies have been reported, with only two proposing post-quantum CLI constructions [13, 15]. However, these schemes entail significantly higher computational costs and lack comprehensive evaluation. This limitation is particularly important given the anticipated impact of quantum computing on current public-key cryptography, underscoring the need for further research into practical, efficient post-quantum CLI schemes.

7.2 Future Direction

Based on the identified gaps, several directions for future research can be outlined. First, future research should focus on developing pairing-free ECC-based CLI schemes under standard assumptions, such as the ECDLP, and achieve Super-Type-1 and Super-Type-2 security in the CLI setting. Although Chin et al. [12] proposed a pairing-free CLI scheme, it uses finite-field DLP rather than elliptic curve cryptography. An ECC-based CLI would offer smaller key sizes and faster operations. Adapting the CLI to an ECC framework opens additional opportunities, given its suitability for practical deployment in resource-constrained environments. This direction should also include investigating CLI security under the standard model rather than the random-oracle model, thereby overcoming the random-oracle model's known limitations and providing stronger theoretical guarantees. To date, only one study has addressed standard model security in the CLI setting [11].

Second, future work should focus on implementing CLI prototypes and conducting empirical performance evaluations. No implementation study exists for CLI. This limits the ability to evaluate CLI schemes under realistic deployment scenarios, including latency, memory consumption, communication overhead, and energy efficiency. Since identification protocols are intended for practical authentication applications, empirical validation should be regarded as an essential component of future CLI research. Future work should implement CLI schemes using established cryptographic libraries to facilitate reproducible performance evaluation. Performance evaluation should measure execution time, communication cost, memory usage, and energy consumption on platforms such as Raspberry Pi, ARM processors, and standard desktops.

Third, future work should pursue tight security reductions for CLI. While Zhang et al. [20] have established the feasibility of tight security reductions in the CLS setting, this remains largely unaddressed in the CLI setting. Future CLI constructions should aim to achieve tight security reductions by eliminating reliance on techniques such as the forking lemma, reset lemma, and rewinding. Another promising direction is the development of reset-secure CLI schemes. While reset-secure identity-based identification schemes without pairings have been proposed in the IBI setting [8], comparable constructions have not yet been explored for CLI. Extending reset security to the certificateless setting would strengthen resilience against adversaries capable of resetting provers while preserving the benefits of certificateless cryptography.

Finally, future research should focus on the development of post-quantum certificateless identification schemes. Existing post-quantum CLI research has primarily focused on supersingular isogeny-based constructions [13, 15]. Although isogeny-based cryptography offers resistance against quantum attacks, these schemes generally involve higher computational complexity and remain relatively less mature than several other post-quantum cryptographic approaches. In contrast, lattice-based cryptography has emerged as a leading post-quantum paradigm, particularly following the standardization of lattice-based digital signature algorithms such as ML-DSA (formerly CRYSTALS-Dilithium) and FN-DSA (derived from Falcon). While these algorithms are designed for digital signatures rather than certificateless identification, their underlying lattice assumptions demonstrate the practical viability and maturity of lattice-based cryptography. To date, lattice-based techniques have not been explored in the CLI setting. Investigating the feasibility of lattice-based CLI schemes, inspired by the design principles of standardized lattice-based constructions, therefore represents a promising direction for future research.

8 Conclusion

This paper has surveyed CLI schemes, classifying them into pairing-based, pairing-free, and post-quantum constructions. The analysis covered security models, adversarial types,

assumptions, reduction tightness, and computational cost.

Several key observations emerged from this survey. First, early CLI schemes predominantly relied on bilinear pairings, which incur significant computational overhead. This led to the development of pairing-free constructions, although only one such scheme has been identified to date. Second, tight security reductions remain rare, and most CLI schemes are proven under the random-oracle model. Third, practical validation is absent in the CLI literature, where no implementation exists for CLI and all performance claims remain theoretical. Finally, post-quantum CLI schemes remain in their early stages, with only a limited number of constructions explored.

CLI remains an underexplored area of research. Future research should focus on designing pairing-free ECC-based CLI schemes that achieve tight security reductions, are simultaneously secure, efficient, and practically deployable, and advance post-quantum constructions. This survey serves as a useful reference for researchers and is intended to motivate further advances in certificateless identification research.

Use of AI Declaration

This work made limited use of AI-assisted tools for language refinement and formatting support. Specifically, AI tools were used to improve grammar, enhance sentence clarity, and assist in LaTeX formatting for tables and figures.

All technical content, including the analysis of certificateless identification schemes, security models, and efficiency comparisons, was developed and verified independently by the author. The AI tools did not contribute to the conceptual design, technical derivations, or research findings of this work.

Acknowledgments

This work was supported by the Ministry of Higher Education Malaysia through the Fundamental Research Grant Scheme under Grant
FRGS/1/2023/ICT07/MMU/03/1. The third author wishes to thank Malaysia Cryptology Technology and Management Centre (PTPKM) for housing his attachment in June 2026, where most of this manuscript was discussed and written. He also wishes to thank EPSRC for the Overseas Travel Grant UKRI4485 that supports his work with Malaysia, in particular PTPKM.

References

[1] Sattam S. Al-Riyami and Kenneth G. Paterson. Certificateless public key cryptography. In *International conference on the theory and application of cryptology and information security*, pages 452–473. Springer, 2003.

[2] Man Ho Au, Yi Mu, Jing Chen, Duncan S Wong, Joseph K Liu, and Guomin Yang. Malicious kgc attacks in certificateless cryptography. In *Proceedings of the 2nd ACM symposium on Information, computer and communications security*, pages 302–311, 2007.

[3] Xavier Boyen. The uber-assumption family. In *Pairing-Based Cryptography – Pairing 2008*, pages 39–56. Springer, 2008.

[4] An Braeken, Ji-Jian Chin, and Syh-Yuan Tan. Ecqv-ibi: Identity-based identification with implicit certification. *Journal of Information Security and Applications*, 63:103027, 2021.

[5] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited. *Journal of the ACM (JACM)*, 51(4):557–594, 2004.

[6] Liqun Chen, Zhaohui Cheng, and Nigel P Smart. Identity-based key agreement protocols from pairings. *International Journal of Information Security*, 6(4):213–241, 2007.

[7] Jason Chia and Ji-Jian Chin. An identity based-identification scheme with tight security against active and concurrent adversaries. *IEEE Access*, 8:61711–61725, 2020.

[8] Ji-Jian Chin, Hiroaki Anada, and Syh-Yuan Tan. Reset-secure identity-based identification schemes without pairings. In *International Conference on Provable Security*, pages 227–246. Springer, 2015.

[9] Ji-Jian Chin, Rouzbeh Behnia, Swee-Huay Heng, and Raphael C.-W. Phan. Cryptanalysis of a certificateless identification scheme. *Security and Communication Networks*, 8(2):122–125, 2015.

[10] Ji-Jian Chin, Raphael C.-W. Phan, Rouzbeh Behnia, and Swee-Huay Heng. An efficient and provably secure certificateless identification scheme. In *Proceedings of the 10th International Conference on Security and Cryptography (SECRYPT-2013)*, pages 371–378. SCITEPRESS (Science and Technology Publications, Lda.), 2013.

[11] Ji-Jian Chin, Raphael C.-W. Phan, and Swee-Huay Heng. An efficient and provable secure certificateless identification scheme in the standard model. *KSII Transactions on Internet and Information Systems*, 8(7):2532–2553, 2014.

[12] Ji-Jian Chin, Syh-Yuan Tan, Swee-Huay Heng, Raphael C.-W. Phan, and Rouzbeh Behnia. A provable secure pairing-free certificateless identification scheme. *International Journal of Computer Mathematics*, 92(8):1520–1535, 2015.

[13] Hassan Daghigh and Gilan Ruholla Khodakaramian. Isogeny-based certificateless identification scheme. *Algebraic Structures and Their Applications*, 2019.

[14] Massoud Hadian Dehkordi and Reza Alimoradi. Certificateless identification protocols from supersingular elliptic curve. *Security and Communication Networks*, 7(6):979–986, 2014.

[15] Keteng Huang, Guoqing Zhou, and Maozhi Xu. Certificateless identification schemes using supersingular isogenies. In *2022 3rd Information Communication Technologies Conference (ICTC)*, pages 127–134, 2022.

[16] Xinyi Huang, Willy Susilo, Yi Mu, and Futai Zhang. On the security of certificateless signature schemes from asiacrypt 2003. In *Cryptology and Network Security*, pages 13–25. Springer, 2005.

[17] Tiong-Sik Ng, Syh-Yuan Tan, and Ji-Jian Chin. A variant of bls signature scheme with tight security reduction. In *International Conference on Mobile Networks and Management*, pages 150–163. Springer, 2017.

[18] Tiong-Sik Ng, Syh-Yuan Tan, and Ji-Jian Chin. A variant of schnorr signature scheme with tight security reduction. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 411–415. IEEE, 2017.

[19] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Workshop on the theory and application of cryptographic techniques*, pages 47–53. Springer, 1984.

[20] Zhenfeng Zhang, Duncan S Wong, Jing Xu, and Dengguo Feng. Certificateless public-key signature: security model and efficient construction. In *International Conference on Applied Cryptography and Network Security*, pages 293–308. Springer, 2006.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 149–165.

Efficient Threshold Signatures: A Runtime-Based Comparative Study of FROST Variants

Nusrat Sultana[1], Sook-Chin Yip[2], Ji-Jian Chin[3], and Ivan Ku[1]

[1] Faculty of Artificial Intelligence and Engineering, Multimedia University, Cyberjaya, Malaysia
nusrat.sultana@student.mmu.edu.my,kccivan@mmu.edu.my
[2] School of Artificial Intelligence and Robotics, Xiamen University Malaysia, Sepang, Malaysia
sookchin.yip@xmu.edu.my
[3] School of Engineering, Computing and Mathematics, University of Plymouth, Plymouth, United Kingdom ji-jian.chin@plymouth.ac.uk

Abstract. Threshold signature schemes enable a group of participants to produce a valid signature when a subset of t out of n members participate in signing, distributing trust, and strengthening security in decentralized systems. Among recent designs, the Flexible Round-optimised Schnorr Threshold (FROST) protocol offers a practical balance of efficiency, low communication cost, and strong guarantees. This paper first situates FROST within its cryptographic foundations, including Schnorr signatures and the discrete logarithm assumption, and outlines its security setting and robustness considerations against malicious or aborting signers. We describe its main components, including the two-round signing flow, distributed key generation (DKG), and the connection between such protocols and secure multiparty computation. We then survey representative extensions and adaptations, such as proactive and dynamic groups, compatibility with verifiable secret sharing, and use of alternative curves and primitives, together with standardization via the Internet Research Task Force Crypto Forum Research Group (IRTF CFRG) (RFC 9591) and application contexts such as blockchain and distributed systems. Complementing this overview, we report a runtime-oriented experimental study of several FROST-style variants and Robust Asynchronous Schnorr Threshold Signatures (ROAST) in a unified implementation over Ristretto255 with SHA-512, summarising mean, median, and dispersion of phase timings under multiple (n, t, α) configurations. The measurements provide a controlled implementation-level comparison of the evaluated protocols, clarifying their relative runtime trade-offs under a unified experimental environment. The results should be interpreted as evidence from controlled computational benchmarking.

Keywords: Threshold signatures · FROST · ROAST · Ristretto255 · Performance evaluation · Schnorr signatures

1 Introduction

The rapid shift toward distributed computation and multiparty collaboration has sharpened the need for cryptographic techniques that support secure, reliable joint operations without a single point of trust [16], [23]. In these environments, even basic tasks such as digital signing must be performed collaboratively. At the same time, these tasks must protect the underlying secret key material. Threshold signature schemes (TSS) address this by distributing a private signing key among (n) parties. Any authorized subset of size (t) can produce a standard signature, while no smaller coalition can do so [22]. This decentralization of trust substantially improves resilience to key compromise and hardware failure. It also provides robustness when some participants are unreliable or even adversarial.

As a result, TSS is a central primitive in secure multiparty computation and modern cryptographic infrastructures [10]. Schnorr signatures form the foundation of FROST, providing a simple and efficient way to generate digital signatures. Their security is based on the hardness of the discrete logarithm problem. This well-established assumption makes it computationally infeasible to recover the secret key from the public key in a cyclic group of prime order [5].

Schnorr-based threshold signatures have emerged as especially compelling because they are algebraically simple and efficient. They rest on the well-studied discrete logarithm assumption [27]. Their linear structure enables straightforward aggregation [5] of local contributions and underpins a broad class of optimised protocols [8]. Yet early threshold implementations of Schnorr signatures suffered from high interaction and communication costs. They also displayed fragile behaviour in the presence of malicious or aborting signers, which limited their use in realistic distributed systems [18]. The Flexible Round-optimised Schnorr Threshold (FROST) scheme was introduced to overcome these drawbacks. It reduces signing to two rounds, cuts network overhead, and supports identifiable aborts, thereby enabling much more practical threshold Schnorr deployments [7].

FROST's security in heterogeneous environments remains delicate. The protocol depends on tightly coordinated nonce generation and interaction patterns. This coordination can create subtle channels for misuse or attack if implementations deviate from the ideal model. Meanwhile, many extensions have emerged, such as proactive key refresh, dynamic committees, and compositions with verifiable or proactive secret sharing. These are proposed in disparate frameworks [6] and are often evaluated in isolation for security and performance, without a common basis for comparison. Additionally, concrete aspects such as runtime and scaling behaviour across different (t, n) configurations are rarely measured in a unified experimental setting. To address this gap, the present work undertakes a structured analysis of FROST and its main variants. It provides a unified experimental evaluation across the Ristretto255 group with SHA-512, clarifying relative performance trade-offs under controlled benchmark conditions.

The main contributions of this work are summarized as follows:

1. A structured analysis of Schnorr-based threshold signatures and the FROST protocol.
2. A unified comparative study of multiple FROST variants and their extensions.
3. A runtime-based experimental evaluation across multiple FROST variants under a unified Rust implementation across (n, t, α) settings, highlighting relative efficiency trade-offs.
4. A discussion of practical considerations, with explicit scope limits for single-process benchmarking.

The remainder of this paper is organized as follows. Sections 2 and 3 present the mathematical preliminaries, the FROST protocol and its variants, and the ROAST protocol. Sections 4 and 5 detail the implementation setup and report the runtime evaluation results. Section 6 discusses the practical implications of the findings, and Section 7 concludes the paper.

2 Preliminaries

The FROST protocol standardizes an efficient two-round Schnorr threshold-signing workflow. Several design variants have been proposed that adjust binding factors, incorporate token-based authentication, or simplify nonce structures, thereby trading implementation complexity for varying levels of robustness and simplicity. Robust Asynchronous Schnorr Threshold Signatures (ROAST) extend this approach by composing a semi-interactive

threshold scheme with additional coordinator logic, ensuring protocol progress even in the presence of malicious or slow signers [16, 23].

2.1 Notation and Mathematical Background

Let $\mathbb{G}$ denote a cyclic group of prime order q, written multiplicatively unless stated otherwise. Let $g \in \mathbb{G}$ be a fixed generator; every element $h \in \mathbb{G}$ can be written as $h = g^x$ for a unique exponent $x \in \mathbb{Z}_q$, where $\mathbb{Z}_q$ denotes the ring of integers modulo q. Scalars such as secret keys, coefficients, and challenges are elements of $\mathbb{Z}_q$.

When considering elliptic-curve groups, scalar multiplication is often expressed in additive notation. In this case, $Y = [x]B$ for base point B corresponds to the multiplicative notation $Y = g^x$. In this work, $\mathbb{G}$ is instantiated as Ristretto255, a prime-order subgroup derived from Curve25519 [11, 14].

Let H_1 and H_2 denote cryptographic hash functions modeled as random oracles, mapping arbitrary-length inputs to $\mathbb{Z}_q$. In practice, we instantiate these using SHA-512 with domain separation.

Discrete Logarithm Assumption: Finally, given $Y = g^x$ for unknown uniformly random $x \in \mathbb{Z}_q$, it is computationally infeasible for any polynomial-time adversary to recover x. This crucial assumption underpins Schnorr signatures and their threshold variants [8].

2.2 Schnorr Signature Scheme

Schnorr signatures are among the most elegant and widely studied constructions in public key cryptography, valued for both conceptual simplicity and practical efficiency. Defined over a cyclic group G of prime order q with generator g, the scheme derives its security from the hardness of the discrete logarithm problem. A signer holds a secret key $x \in \mathbb{Z}_q$, while the corresponding public key is given by $Y = g^x$[25]. The Schnorr signing procedure samples a fresh random nonce to generate a commitment, derives a hash-based challenge from the message, and computes a linear response using the nonce and secret key [27].

These characteristics have made Schnorr signatures a natural foundation for threshold constructions. The ability to decompose the signing operation into independent contributions, each tied to a participant's local secret share, enables efficient aggregation into a single valid signature. Prior work has also studied Schnorr and related signature variants with tighter security reductions and stronger unforgeability notions [20, 21, 1]. Consequently, many modern threshold protocols, including FROST, leverage the Schnorr paradigm for efficient multiparty signing [10].

Key Generation: A signer selects a secret key $x \in \mathbb{Z}_q$ uniformly at random and computes the public key $Y = g^x$.

Signing: To sign a message m, the signer samples a nonce $r \in \mathbb{Z}_q$, computes the commitment $R = g^r$, and derives the challenge

$$c = H_2(R, Y, m). \tag{1}$$

The signature is the pair $\sigma = (R, s)$, where $s \in \mathbb{Z}_q$ is a linear response that combines the ephemeral nonce r with the secret key x weighted by the challenge c

$$s = r + c \cdot x \pmod{q}. \tag{2}$$

Verification: A signature $\sigma = (R, s)$ on message m under public key Y is accepted if

$$g^s = R \cdot Y^c, \quad \text{where } c = H_2(R, Y, m). \tag{3}$$

Equivalently, in elliptic-curve notation:

$$[s]B = R + [c]Y. \tag{4}$$

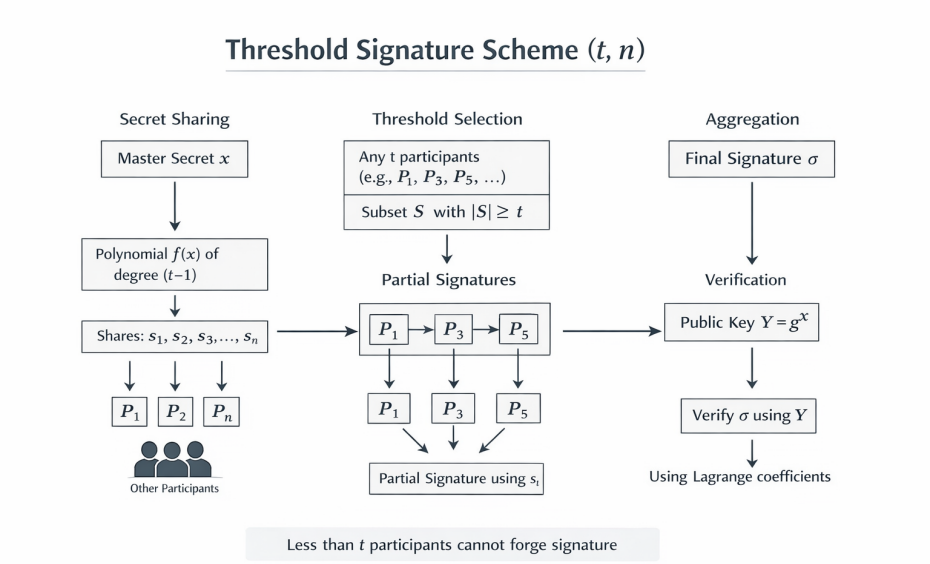


Figure 1: Workflow of a (t, n)-threshold signature scheme from secret sharing to final signature generation and verification.

2.3 Threshold Signature Schemes

A (t, n)-threshold signature scheme distributes a signing key among n parties such that any subset of at least t participants can jointly produce a valid signature, while fewer than t parties cannot forge. The resulting signature is indistinguishable from a standard signature and can be verified using a single public key. Figure 1 illustrates this workflow from secret sharing through partial signing to final verification.

This is commonly achieved using Shamir secret sharing [26]. A secret $x \in \mathbb{Z}_q$ is embedded as the constant term of a polynomial f of degree $t-1$, and each participant i receives a share $s_i = f(i)$. Any subset S of size t can reconstruct the secret via Lagrange interpolation:

$$x = f(0) = \sum_{i \in S} \lambda_i s_i, \tag{5}$$

where λ_i are publicly computable coefficients.

In practice, threshold signing avoids the need to reconstruct x. Instead, participants compute partial signatures using their shares, which are then combined to produce a valid Schnorr signature under $Y = g^x$.

Security is defined through unforgeability; even an adversary that corrupts up to $t-1$ parties and observes protocol executions cannot generate a valid signature on a new message with non-negligible probability [12].

In threshold signing, the secret x is never reconstructed. Instead, participants combine partial signatures in [26] using these coefficients to produce a valid Schnorr signature under $Y = g^x$.

The preliminaries above establish the foundations for notation and protocol. This work does not introduce new cryptographic constructions; instead, it focuses on a runtime-based comparison of FROST variants and ROAST under a unified experimental framework using Ristretto255 and SHA-512. Prior work establishes the theoretical foundations of FROST and ROAST [16, 23]. This work evaluates multiple implementations of FROST

variants and ROAST within a unified framework, filling the gap in empirical comparisons across variants under consistent conditions. We provide a unified experimental framework, comparative runtime analysis across protocol phases, and insights into performance trade-offs, particularly highlighting the cost of robustness mechanisms such as ROAST.

3 Overview of FROST Protocol

FROST is a modern threshold signature scheme that enables multiple participants to jointly produce a standard Schnorr signature without reconstructing the secret key [16, 8]. It leverages the linearity of Schnorr signatures to achieve efficient aggregation while maintaining strong security guarantees. The protocol is particularly suitable for distributed systems where trust must be decentralized and communication overhead minimized.

3.1 FROST Protocol

FROST is designed to enable a set of distributed participants to collaboratively produce a standard Schnorr signature under a single group public key, without ever reconstructing the full signing secret at any single location [16, 5]. Each participant holds a share of the secret, and only a quorum of at least t participants is required to generate a valid signature. The protocol is termed round-optimised because it reduces the signing interaction to two primary rounds under typical conditions, making it suitable for environments with frequent signing operations or high network latency. This efficiency is enabled by the linear structure of Schnorr signatures, which allows both the secret key and the nonce to be combined additively.

3.2 Two-Round Signing

The signing procedure in FROST consists of two main rounds at the protocol level, although implementations may optimize or pipeline certain steps. In the first round, each participating signer generates fresh randomness and publishes commitments derived from their nonce shares. These commitments are exchanged among all signers to ensure that each participant has a consistent view of the signing session before any partial signature is revealed. This step prevents adversarial behaviour where nonces could otherwise be adaptively chosen.

In the second round, participants compute their partial signature shares using the message, the set of participating signers, and the previously exchanged commitments. These partial signatures are then combined, typically by an aggregator, into a single group signature. The aggregator requires no access to any secret information and merely performs a deterministic combination of the shares [16, 8]. The resulting signature verifies in the same way as a standard Schnorr signature under the group public key, making it fully compatible with existing verification procedures.

3.3 Distributed Key Generation (DKG)

Before the signing phase, participants must establish a shared public key and corresponding secret shares. While a trusted dealer could distribute shares using schemes such as Shamir Secret Sharing, this introduces a central point of trust. Distributed Key Generation (DKG) eliminates this dependency by allowing all participants to jointly generate the secret in a decentralized manner [26].

In a typical DKG protocol, each participant contributes randomness by generating a polynomial and broadcasting commitments to its coefficients [13]. Participants exchange shares privately and verify their correctness using these commitments. At the end of the

protocol, each participant obtains a secret share, and the group public key is derived from all contributions. Importantly, no individual participant learns the full secret key.

3.4 FROST Variants

FROST1 is commonly treated as the baseline two-round Schnorr signing protocol. In this design, each signer prepares nonce-related material, publishes commitments, and then produces a partial signature using the appropriate Lagrange coefficient for the selected signing set [16].

The Binding variant follows the same general flow but introduces an explicit binding factor. This factor hashes all participants' nonce commitments, and typically also the message and group public key, into a scalar before the aggregate commitment and challenge are formed. As a result, the protocol binds each signer's contribution to the full signing context, reducing the risk that a malicious participant can adapt its response after observing partial information from others.

FROST-2+ preserves the two-round structure but adds authenticated preprocessing. In addition to standard nonce commitments, signers derive auxiliary scalars and group elements, often together with Schnorr-style consistency checks, so that tokens can be validated before the signing phase begins. This improves protection against misuse of preprocessing material, but it also increases the amount of hashing and scalar computation required [8].

FROST-2# follows a similar authenticated-token approach, but with a simpler construction [30]. Depending on the design, it may reduce the number of nonce components or simplify the hashing structure used during preprocessing and signing. In practice, both FROST-2+ and FROST-2# introduce greater preprocessing and signing overhead than the baseline variants when evaluated under the same curve and hash setting.

3.5 ROAST Protocol

ROAST extends FROST to asynchronous settings [23]. It introduces a coordinator who manages multiple signing attempts and ensures progress even in the presence of offline or malicious participants. ROAST acts as a wrapper over FROST and does not change the underlying signature equation. ROAST is not a different elliptic-curve signature equation; rather, it wraps a semi-interactive threshold scheme such as FROST. The coordinator schedules signing sessions, collects commitments, and can run sessions in parallel until enough honest signers complete the protocol. Share validation (identifiable abort) allows the coordinator to drop misbehaving participants and retry without trusting them to keep secrets. These robustness guarantees come at the cost of additional latency and implementation complexity. In our evaluation, ROAST is exercised in a single-process simulation with only honest and responsive signers; the benchmark therefore measures coordinator and session-management overhead, but not the protocol's progress guarantees under faulty, slow, or malicious participants. Because this benchmark aggregates coordinator activity into the signing phase, ROAST records a noticeably higher signing time than a single FROST session executed under the same conditions.

4 Implementation Details

Implementing Schnorr-based FROST requires selecting the elliptic-curve group and hash function, in accordance with RFC 9591 [5]. The standard specifies ciphersuites, including Ed25519, Ristretto255, and secp256k1, with corresponding hash functions and test vectors for correctness and interoperability. The protocol comprises setup, key generation,

preprocessing, signing, combining, and verification, with the last step using Schnorr-style verification [5].

We implemented all schemes in a unified codebase to ensure that differences in measured runtime reflect protocol structure rather than unrelated library choices. This implementation, written in Rust, uses curve25519-dalek [9] for Ristretto255 [17, 11, 2] group operations and sha2 for SHA-512 hashing [24]. All variants follow the same workflow for decentralized key generation (DKG), preprocessing, leader request construction, partial signing, aggregation, and verification, building upon the standardized stages introduced previously. Our work complements prior implementation-oriented studies of cryptographic authentication on mobile and hybrid-PKI platforms [31, 4, 3].

Building on this common implementation foundation, the experiments reported in Section 5 were conducted using the Rust release binary (cargo build –release) on a single workstation [15, 19]. All variants use identical implementation settings to ensure a fair comparison of runtimes.

Cryptographic parameters are fixed: Ristretto255 [17] (prime-order, 32-byte encodings) as the group, and SHA-512 for deriving challenges and binding scalars with domain separation. Benchmarking randomness is seeded for reproducibility. All measurements use the same scheduler, timing API, and build config, ensuring observed differences reflect protocol behaviour rather than artifacts.

4.1 Implemented Variants

FROST1 specifies the canonical two-round threshold Schnorr signing protocol, employing per-signer nonce pairs and standard aggregation. FROST-2+ (referred to as a "FROST-2"-style augmented protocol) incorporates authenticated preprocessing, hash-derived auxiliary inputs, and pre-signing token verification mechanisms. FROST-2# applies these extensions with a modified or simplified nonce/token framework in a pedagogical construction, thereby altering preprocessing and signing efficiency relative to the baseline. ROAST encapsulates a FROST-derived core within coordinator-driven session management, shared verification, and signature orchestration, trading enhanced robustness for increased control-flow complexity and support for repeated signing attempts.

The purpose of the implementation is to enable a comparative runtime evaluation; no novel security proofs are presented. We precisely measure the runtime for identical phases, fixing the (n, t, α) parameters and the hash/group structures. The principal distinguishing factors among variants are: (i) the nature and structure of binding and authentication mechanisms, and (ii) ROAST's incremental coordinator and session-related overhead.

4.2 Experimental Parameters

Table 1 summarises the primary benchmark configuration used for the Rust experiments reported in Section 5. In addition to the primary configuration, two supplementary settings of (n, t, α) are evaluated to examine the impact of different committee sizes and threshold values, as shown in Table 4. The threshold scheme uses n participants, threshold t, and signing set size α (the number of active signers in each signing session).

Each configuration is evaluated over T independent trials. In every trial, a fresh DKG execution and one or more signing sessions are performed. The signed message is a fixed-length octet string. FROST variants generate fresh randomness during each preprocessing phase, whereas ROAST reuses the same random sequence across its internal signing attempts. To ensure a fair comparison, all variants use the same implementation, timing methodology, phase boundaries, compiler settings, and statistical analysis (mean, median, and sample standard deviation).

All reported configurations use $T = 30$ independent trials with seed 42 to ensure a fair comparison across parameter settings.

Table 1: Implementation and benchmark configuration.

Item	Setting
Language	Rust
Libraries	`curve25519-dalek`, `sha2`, `rand`
Group / hash	Ristretto255 / SHA-512
Participants n	20
Threshold t	7
Signing set size α	7
Trials T	30
Seed	42

4.3 Measurement Methodology

Execution is divided into protocol-aligned phases: setup (initialisation and random number generation (RNG)), key generation (DKG rounds), preprocessing (nonce and token material), signing (partial signatures and ROAST coordinator operations), combining (aggregation of partial signatures), and verification (final Schnorr verification). For preprocessing and signing, the per-participant metric is computed as the total contributions from participating signers in these phases divided by α, the number of participants, yielding the average contribution per participant.

Runtimes are measured in milliseconds using high-resolution clocks. For each phase and variant, the mean (representing typical cost), median (a robust measure of central tendency), and sample standard deviation (capturing variability across trials) are reported. The mean reflects the average computational cost, the median reduces the influence of outliers, and the standard deviation quantifies run-to-run variability under the controlled single-workstation benchmarking environment. Table 2 summarises the runtime metrics and protocol phases consistently applied across all evaluated schemes.

The implementation is publicly available and archived with a DOI to support reproducibility [28, 29].

Table 2: Phases and reported statistics.

Phase / metric	Description
Setup	Initialisation and RNG setup
Key generation	DKG (broadcasts, share verification, final keys)
Preprocessing	Nonces and variant-specific token material
Signing	Partial signatures; ROAST includes coordinator/session work
Combining	Aggregation of partial signatures
Verification	Final Schnorr verification
Per-participant	Mean time per signer within a phase (where traced)
Mean / median / std	Central tendency and dispersion over T trials

5 Runtime Evaluation

This section evaluates the runtime performance and overhead of FROST variants and ROAST, implemented using Ristretto255 and SHA-512. We compare phase-wise execution time to highlight efficiency and practical trade-offs across schemes.

5.1 Evaluation Criteria

The evaluation focuses on runtime efficiency and relative overhead for FROST1, FROST-2+, FROST-2#, and ROAST using Ristretto255 with SHA-512. Security properties are taken from the literature; here, we compare practical CPU times within a single implementation on a single workstation. To ensure fairness, each variant is measured with identical curve and hash choices, phase boundaries, and statistical aggregation. These constraints are satisfied for the results in Tables 3 and 4.

5.2 Comparative Results

Table 3 summarises mean, median, and standard deviation for the Rust release build with n=20, t=7, α=7, T=30 trials, and seed 42. In signing, FROST1, Binding, FROST-2+, and FROST-2# cluster between approximately 3.66–4.09 ms, while ROAST shows a substantially higher mean of 10.89 ms. This gap reflects ROAST's coordinator logic and session handling on top of threshold-signing steps. Key-generation means are comparable across protocols (207–239 ms) because the same DKG path runs before variant-specific signing. FROST-2+ and FROST-2# display higher preprocessing means than FROST1 (0.75–0.85 ms versus 0.52–0.54 ms), reflecting added hashing and token material. In our evaluation framework, ROAST's aggregate preprocessing and verification costs are included in the signing cost rather than measured separately. Therefore, the zero values in those columns do not indicate that no computation is required. The signing and per-participant results should serve as the primary indicators of ROAST's wrapper overhead.

Table 3: Runtime summary (ms): Rust, $n = 20$, $t = 7$, $\alpha = 7$, $T = 30$, seed 42. Mean (μ), median (mdn), and sample standard deviation (σ) over trials. Abbreviations: DKG = Distributed Key Generation; Preproc. = preprocessing; Verify = verification; Sign./part. = signing time per participant.

	FROST1			Binding			FROST-2+			FROST-2#			ROAST		
Phase	μ	mdn	σ	μ	mdn	σ	μ	mdn	σ	μ	mdn	σ	μ	mdn	σ
Key gen	216.54	209.02	26.36	227.41	210.49	57.79	224.59	208.00	45.63	239.43	211.72	68.21	232.74	207.74	71.40
Preproc.	0.54	0.52	0.11	0.53	0.45	0.22	0.75	0.68	0.15	0.85	0.69	0.40	0.00	0.00	0.00
Signing	3.76	3.63	0.55	3.77	3.53	1.30	3.66	3.45	0.65	4.09	3.59	1.40	10.89	10.33	1.75
Verify	0.10	0.08	0.04	0.09	0.08	0.01	0.09	0.08	0.03	0.29	0.08	1.06	0.00	0.00	0.00
Sign./part.	0.54	0.49	0.11	0.54	0.47	0.29	0.52	0.47	0.11	0.58	0.48	0.30	1.09	1.03	0.17

Table 4 reports mean key-generation and signing times across three (n, t, α) configurations, each measured over T=30 trials. Signing cost increases with α, as expected from additional partial signatures (e.g., FROST1 signing rises from 1.83 ms at $(10, 5, 5)$ to 3.76 ms at $(20, 7, 7)$ and 7.68 ms at $(20, 10, 10)$), while the relative ordering among variants is preserved: FROST-family schemes remain close to one another, and ROAST consistently incurs the highest signing overhead. Key generation scales with n and t because DKG performs more share exchanges and verifications; the choice of variant has a limited effect on this phase, consistent with Table 3.

To complement Table 3 with histogram-based plots of runtime behaviour, we approximate per-trial distributions using normal models derived from the measured mean and standard deviation at the primary configuration (T=30). These figures offer an intuitive view of overlap and separation among variants; they are illustrative and do not replace the tabulated statistics.

The histogram visualizations confirm several key trends from Table 3. Key generation shows significant overlap across all variants, suggesting execution times are largely independent of protocol variant. All schemes rely on similar cryptographic operations during this phase. High runtime in this stage indicates that key generation remains the dominant contributor to workflow execution time.

Table 4: Mean key generation and signing times (ms) across parameter configurations. All configurations use T=30 trials and seed 42.

Configuration	Phase	FROST1	Binding	FROST-2+	FROST-2#	ROAST
(n, t, α)=(10, 5, 5)	Key gen	43.54	45.59	43.47	45.89	45.15
	Signing	1.83	1.95	1.91	1.94	6.09
(n, t, α)=(20, 7, 7)	Key gen	216.54	227.41	224.59	239.43	232.74
	Signing	3.76	3.77	3.66	4.09	10.89
(n, t, α)=(20, 10, 10)	Key gen	290.67	296.10	293.02	298.22	294.31
	Signing	7.68	7.57	7.40	7.72	19.37

The preprocessing phase reveals a more nuanced pattern, with two distinct clusters emerging. FROST1 and Binding form a lower-cost group, while FROST-2+ and FROST-2# occupy a higher range due to additional operations such as hash computations and token-related processing. This difference illustrates the trade-off between efficiency and enhanced security mechanisms introduced in the augmented variants.

Overall, the histogram visualizations complement the numerical results and confirm the observed runtime trends across the protocol variants.

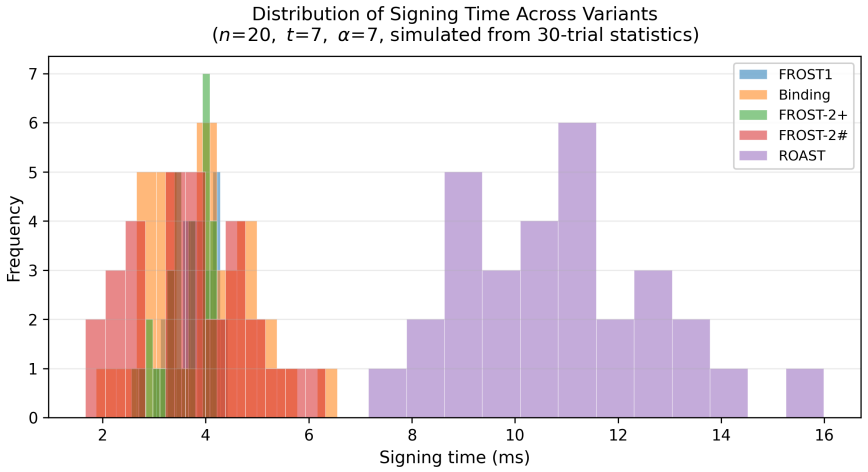


Figure 2: Distribution of signing phase duration (ms) across FROST1, Binding, FROST-2+, FROST-2#, and ROAST.

Figure 2 shows that FROST1, Binding, FROST-2+, and FROST-2# have similar signing runtimes, whereas ROAST exhibits a clearly higher runtime. This confirms the results in Table 3, showing that ROAST incurs additional coordinator-related overhead in the single-process benchmark. The FROST variants cluster between approximately 3.0–4.5 ms, whereas ROAST is centered near 10.9 ms, demonstrating the additional runtime overhead of its coordinator-driven robustness mechanisms.

By removing ROAST from the plot, Figure 3 zooms into the FROST-family signing range. FROST-2# shows a higher mean (4.09 ms) than FROST1 (3.76 ms), while FROST-2+ remains close to the baseline. FROST1 and Binding largely overlap, confirming that the binding-factor mechanism adds negligible signing overhead in this configuration. The slight spread among augmented variants is consistent with the extra scalar and hash operations in authenticated preprocessing.

Figure 4 shows that all five distributions overlap heavily, centered near 208–239 ms. This confirms that DKG cost is largely independent of the signing variant, since the same

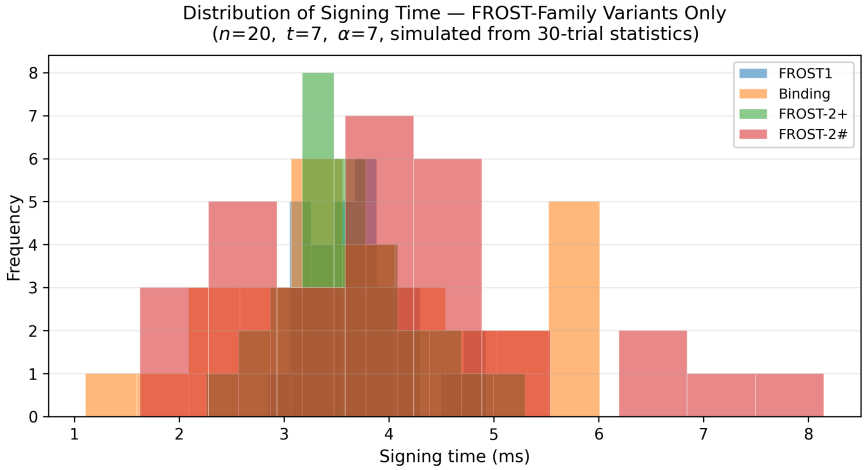


Figure 3: Distribution of signing phase duration (ms) for FROST variants only, namely FROST1, Binding, FROST-2+, and FROST-2#, excluding ROAST to reveal finer differences within the FROST family.

polynomial commitment, share exchange, and Feldman verification steps are executed before any variant-specific logic begins. The reported dispersion reflects run-to-run variability observed over T=30 trials.

Figure 5 shows that preprocessing runtimes form two visible groups. FROST1 and Binding form a lower band near 0.45–0.55 ms, while FROST-2+ and FROST-2# occupy a higher band near 0.75–0.85 ms. This separation reflects the additional hash derivations and token point operations required by authenticated preprocessing in the augmented variants.

Taken together, Figures 2–5 complement Tables 3 and 4 by showing overlap and separation among variants at the primary configuration. The histogram views support three observations: key generation remains similar across variants, FROST-family schemes stay close in signing cost, and ROAST introduces a visibly higher signing burden due to coordinator logic in our benchmark harness.

5.3 Phase-wise Discussion

Key generation dominates total wall-clock time for all variants in Table 3, as expected from many group operations and hashes in DKG. Preprocessing and signing are secondary, but distinguish variants: augmented paths (FROST-2+, FROST-2#) increase preprocessing relative to FROST1. Combining and verification contribute negligibly to the total cost for all FROST variants. Combining requires only a single scalar addition per partial signature. It records a mean of approximately 0.002 ms across FROST1, Binding, FROST-2+, and FROST-2#, while verification reduces to a single Schnorr check, with mean values between 0.08 ms and 0.29 ms. The trade-off is familiar: optimised minimal paths (FROST1) reduce per-session computation, while robustness-oriented designs (ROAST) spend extra time on coordinator logic; however, our benchmark does not evaluate ROAST under actual faults or asynchrony.

5.4 Practical Interpretation

Distributed key generation (DKG) is typically performed during system setup and can therefore be amortized across many signing sessions, whereas preprocessing and signing

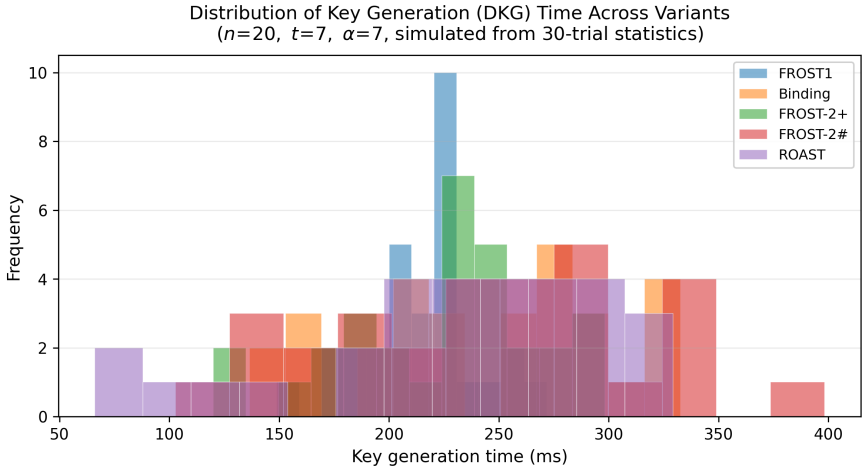


Figure 4: Distribution of key generation (DKG) runtime (ms) across all five variants. The distributions are derived from the measured mean and standard deviation over 30 trials at (n, t, α)=$(20, 7, 7)$.

contribute to the runtime of every signature. The reported standard deviations (e.g., ROAST signing (σ) $\approx$ 1.75 ms in Table 3) indicate measurable run-to-run variability under the controlled single-workstation benchmark. These results quantify computational performance only and do not include communication overhead, network latency, message serialisation, memory usage, or energy consumption. Consequently, the findings should be interpreted as evidence of implementation-level benchmarking.

5.5 Summary of Findings

Under a unified implementation of Ristretto255 with SHA-512, distributed key generation (DKG) remains the dominant computational cost across all evaluated variants. Among the measured schemes, FROST1 and Binding achieve the lowest per-session runtime. FROST-2+ and FROST-2# introduce moderate preprocessing overhead. ROAST incurs the highest signing cost due to coordinator and session-management operations across the three evaluated (n, t, α) configurations, and the same relative ordering is observed. However, the reported runtimes should be interpreted only within the benchmark settings used for testing.

This work provides a unified implementation and benchmarking framework for multiple FROST variants and ROAST, reporting mean, median, and standard deviation for each protocol phase. The primary contribution is a controlled implementation-level comparison across a common codebase and multiple parameter configurations.

Threats to Validity The reported results depend on the benchmark hardware, operating system scheduling, and compiler optimizations. All experiments were conducted on a single workstation in a single-process environment and therefore do not capture network latency, message serialisation, distributed execution, or adversarial behaviour. The reported ROAST timings quantify coordinator overhead only among honest participants and do not assess its robustness in asynchronous or faulty environments. Furthermore, communication overhead, memory usage, and energy consumption were not measured. To support reproducibility, the implementation uses fixed random seeds, documented (n, t, α) configurations, archived source code, compiler version, commit hash, and `cargo build -release` settings.

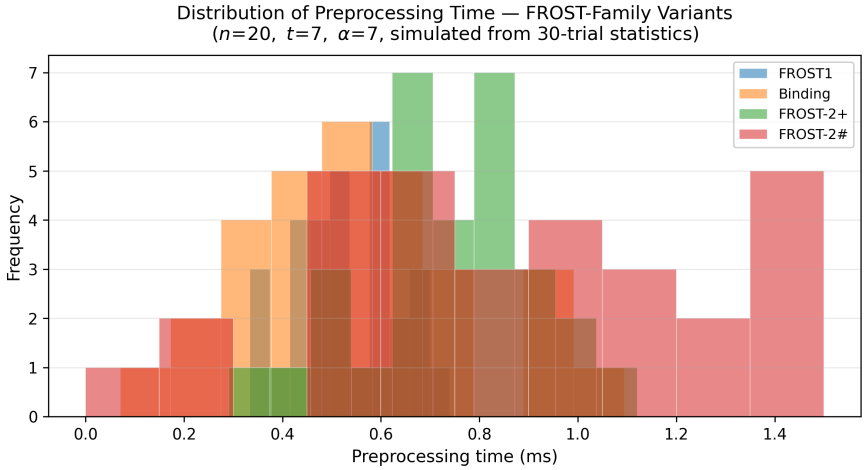


Figure 5: Distribution of preprocessing time (ms) for FROST-family variants only (ROAST excluded because its harness attributes preprocessing cost to the signing phase).

5.6 Limitations

Several limitations of this study should be acknowledged to properly interpret the reported results. Although all three (n, t, α) configurations use 30 trials, the experimental space remains limited and does not establish general laws of scalability. Communication overhead, memory usage, energy consumption, and adversarial robustness were not measured directly, even though these factors matter for deployment-oriented claims.

All benchmarks were collected from a Rust release binary running on a single workstation. Because signers and the ROAST coordinator ran within the same process with honest participants, the measurements reflect only computational cost and local control-flow overhead. They do not capture network latency, message serialisation, concurrent scheduling, or ROAST's progress guarantees in the presence of faulty or malicious signers. As a result, although the relative ordering among variants is informative, absolute timings would differ in distributed deployments, and the present experiments do not demonstrate ROAST's robustness benefit.

Despite these limitations, a unified implementation and consistent measurement methodology enable a fair and controlled comparison across all variants. The results, therefore, provide meaningful insight into the computational trade-offs among FROST-style designs across the tested configurations. At the same time, deployment guidance for mobile, IoT, or asynchronous networks remains outside the scope of this study.

6 Discussion

The experimental results reveal clear computational differences among the evaluated protocols. FROST1 and Binding achieve the lowest signing overhead under the benchmark conditions, whereas FROST-2+ and FROST-2# introduce moderate preprocessing overhead through additional authenticated preprocessing operations. ROAST records the highest signing runtime, approximately 2.9 times that of FROST1 under the primary benchmark configuration $(n, t, \alpha) = (20, 7, 7)$, reflecting the additional coordinator and session-management logic incorporated by the protocol. Because the evaluation was conducted using only honest participants in a single-process environment, these measurements quantify computational overhead rather than ROAST's robustness under adversarial or asynchronous conditions.

From a computational perspective, distributed key generation (DKG) is typically performed during system initialisation and can therefore be amortized across multiple signing sessions. In contrast, preprocessing and signing directly contribute to the latency of each signature operation. Across the evaluated benchmark configurations, the FROST-family variants consistently incur lower signing costs than ROAST. However, communication overhead, network latency, memory usage, and energy consumption were not measured.

The reported runtimes were obtained under a controlled single-workstation, single-process benchmark. In practical distributed deployments, additional factors such as communication latency, message serialisation, concurrent scheduling, and network variability will influence overall end-to-end performance. Although the additional coordinator processing required by ROAST is inherent to its protocol design, evaluating its robustness and overall performance in realistic distributed environments remains an important direction for future work.

7 Conclusion

This work presents a unified implementation-level runtime evaluation of FROST1, Binding, FROST-2+, FROST-2#, and ROAST using Ristretto255 with SHA-512. By applying identical implementation settings, phase boundaries, and measurement methodology across all variants, the study isolates protocol-level computational differences from implementation artifacts. It enables a fair comparison under a controlled benchmarking environment.

The results show that distributed key generation (DKG) remains the dominant computational cost across all evaluated schemes, while preprocessing and signing distinguish the protocol variants. FROST1 and Binding achieve the lowest per-session runtime, whereas FROST-2+ and FROST-2# introduce moderate preprocessing overhead due to additional authenticated preprocessing operations. ROAST incurs the highest signing cost due to coordinator and session management overhead in the single-process benchmark; however, its robustness to faulty, slow, or malicious participants was outside the scope of this evaluation.

Overall, the reported results provide implementation-level evidence of the computational trade-offs among contemporary FROST-style threshold signature schemes. They should be interpreted as controlled benchmarking results. Future work includes evaluating the protocols in distributed environments, conducting fault-injection experiments for ROAST, measuring communication, memory, and energy costs, and profiling performance on resource-constrained ARM-based platforms.

Use of AI Declaration

In this paper, Artificial Intelligence (AI) tools were used only for language refinement, grammar correction, and manuscript clarity. All research work, including the Rust implementation, benchmarking, and data collection, was conducted entirely by the authors. No AI was used to generate or modify any experimental results. All AI-assisted content was reviewed for accuracy.

Code and Data Availability

The implementation and benchmark code used in this study are publicly available. The source repository, including the Rust codebase for all five threshold signing variants (FROST1, Binding, FROST-2+, FROST-2#, and ROAST), along with the benchmarking framework and CSV export scripts, is available at:

https://github.com/NusratTule/FROST-2_rust [28]

To ensure long-term availability and reproducibility, an archived version of the repository has been deposited on Zenodo and assigned a digital object identifier (DOI):
https://doi.org/10.5281/zenodo.19589050 [29]
Researchers can reproduce the reported results using the provided implementation and benchmark configuration.

Acknowledgments

This work was supported by the Ministry of Higher Education Malaysia through the Fundamental Research Grant Scheme under Grant Number: FRGS/1/2023/ICT07/MMU/03/1. Dr. Ji-Jian Chin wishes to thank the Malaysia Cryptology Technology and Management Center (PTPKM) for hosting his attachment, during which most of this manuscript was discussed and written. He also wishes to thank EPSRC for the Overseas Travel Grant UKRI4485, which supports his work with Malaysian partners MySEAL and PTPKM.

References

[1] Jason Chia, Ji-Jian Chin, and Sook-Chin Yip. Digital signature schemes with strong existential unforgeability. *F1000Research*, 10, 2021. Art. no. 931.

[2] Jason Chia, Ji-Jian Chin, and Sook-Chin Yip. Evaluating pairing-free identity-based identification using Curve25519. In *Advances in Cyber Security (ACeS 2020)*, volume 1347 of *Communications in Computer and Information Science*, pages 179–193, 2021.

[3] Jason Chia, Ji-Jian Chin, and Sook-Chin Yip. Pluggable authentication module meets identity-based identification. In *Advances in Cyber Security (ACeS 2021)*, volume 1487 of *Communications in Computer and Information Science*, pages 155–175, 2022.

[4] Jason Chia, Swee-Huay Heng, Ji-Jian Chin, Syh-Yuan Tan, and Wei-Chuen Yau. An implementation suite for a hybrid public key infrastructure. *Symmetry*, 13(8), 2021. Art. no. 1535.

[5] Ji-Jian Chin, Syh-Yuan Tan, Sook-Chin Yip, and Nur Hanis Sabrina Suhaimi. A technical review of FROST threshold signatures: Variants, extensions, security, and RFC 9591. In *2025 3rd International Conference on Cyber Resilience (ICCR)*, pages 1–7, 2025.

[6] Annalisa Cimatti, Francesco De Sclavis, Giuseppe Galano, Sara Giammusso, Michela Iezzi, Antonio Muci, Matteo Nardelli, and Marco Pedicini. Dynamic-FROST: Schnorr threshold signatures with a flexible committee. *Journal of Mathematical Cryptology*, 19(1), 2025. Art. no. 20240045.

[7] Deirdre Connolly, Chelsea Komlo, Ian Goldberg, and Christopher A. Wood. The flexible round-optimized Schnorr threshold (FROST) protocol for two-round Schnorr signatures. RFC 9591, RFC Editor, 2024.

[8] Elizabeth Crites, Chelsea Komlo, and Mary Maller. How to prove Schnorr assuming Schnorr: Security of multi- and threshold signatures. IACR Cryptology ePrint Archive, Report 2021/1375, 2021.

[9] Dalek Cryptography. curve25519-dalek: A pure-Rust implementation of group operations on Ristretto and Curve25519. [Online]. Available: https://github.com/dalek-cryptography/curve25519-dalek, 2024. Accessed: Apr. 14, 2026.

[10] Ivan Damgård, Claudio Orlandi, Akira Takahashi, and Mehdi Tibouchi. Two-round n-out-of-n and multi-signatures and trapdoor commitment from lattices. *Journal of Cryptology*, 35(2), 2022. Art. no. 14.

[11] Henry de Valence, Jack Grigg, George Tankersley, Filippo Valsorda, and Isis Lovecruft. The Ristretto255 group. Internet-draft, Internet Engineering Task Force, 2019.

[12] Sinan Ergezer, Holger Kinkelin, and Filip Rezabek. A survey on threshold signature schemes. *Network*, 49(1):49–53, 2020.

[13] Paul Feldman. A practical scheme for non-interactive verifiable secret sharing. In *28th Annual Symposium on Foundations of Computer Science*, pages 427–438, 1987.

[14] S. Josefsson and I. Liusvaara. Edwards-curve digital signature algorithm (EdDSA). RFC 8032, RFC Editor, 2017.

[15] Steve Klabnik and Carol Nichols. *The Rust Programming Language*. No Starch Press, 2023.

[16] Chelsea Komlo and Ian Goldberg. FROST: Flexible round-optimized Schnorr threshold signatures. In *Selected Areas in Cryptography (SAC)*, pages 34–65, 2020.

[17] Eva Kupcova, Patrik Zelenak, Matus Pleva, and Milos Drutarovsky. Optimization of Ristretto255 group implementation for Cortex-M4-based cryptographic applications. In *2024 34th International Conference Radioelektronika (RADIOELEKTRONIKA)*, pages 1–5, 2024.

[18] Yehuda Lindell. Simple three-round multiparty Schnorr signing with full simulatability. IACR Cryptology ePrint Archive, Report 2022/374, 2022.

[19] Denisa Muntean and Ciprian Oprişa. Software composition analysis on Rust binaries. In *2025 IEEE 21st International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 1–8, 2025.

[20] Tiong-Sik Ng, Syh-Yuan Tan, and Ji-Jian Chin. A variant of Schnorr signature scheme with tight security reduction. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 411–415, 2017.

[21] Tiong-Sik Ng, Syh-Yuan Tan, and Ji-Jian Chin. A variant of BLS signature scheme with tight security reduction. In *Mobile Networks and Management (MONAMI 2017)*, volume 235 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 150–163, 2018.

[22] Jonas Nick, Tim Ruffing, and Yannick Seurin. MuSig2: Simple two-round Schnorr multi-signatures. In *Annual International Cryptology Conference (CRYPTO)*, pages 189–221, 2021.

[23] Tim Ruffing, Viktoria Ronge, Elliott Jin, Jonas Schneider-Bensch, and Dominique Schröder. ROAST: Robust asynchronous Schnorr threshold signatures. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2551–2564, 2022.

[24] RustCrypto Developers. RustCrypto: Hashes—collection of cryptographic hash functions in pure Rust. [Online]. Available: https://github.com/RustCrypto/hashes, 2026. Accessed: Apr. 14, 2026.

[25] C. P. Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, 4(3):161–174, 1991.

[26] Adi Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.

[27] Victor Shoup. The many faces of Schnorr. IACR Cryptology ePrint Archive, Report 2023/1019, 2023.

[28] Nusrat Sultana. FROST-2_rust: Unified Rust implementation and runtime benchmark of FROST threshold signature variants and ROAST over Ristretto255. [Online]. Available: https://github.com/NusratTule/Frost2_rust, 2026. Accessed: Apr. 15, 2026.

[29] Nusrat Sultana. FROST-2_rust: Unified Rust implementation and runtime benchmark of FROST threshold signature variants and ROAST over Ristretto255 (archived). Zenodo, 2026.

[30] Syh-Yuan Tan, Will Wang, and Ryan Chow. From TS-SUF-2 to TS-SUF-4: Practical security enhancements for FROST2 threshold signatures. IACR Cryptology ePrint Archive, Report 2026/075, 2026.

[31] Thong-Yun Teh, Yik-Shu Lee, Zi-Yik Cheah, and Ji-Jian Chin. IBI-mobile authentication: A prototype to facilitate access control using identity-based identification on mobile smart devices. *Wireless Personal Communications*, 94:127–144, 2017.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 166–187.

Measuring Neural Attackability of Lightweight Authenticated Encryption: Towards Robust Cipher Design

Norziana Jamil[1], Hazim Noman Abed[2,3], Je Sen Teh[4], Yasir Amer Abbas[5], Muhammad Reza Z'aba[6] and Noor Hasan Hassoon[7]

[1] College of IT and Cyber Security Research Lab, UAE University, P.O.Box 15551, Al Ain, Abu Dhabi, United Arab Emirates Norziana@uaeu.ac.ae
[2] College of Graduate Studies, Universiti Tenaga Nasional, Malaysia
[3] College of Science, University of Diyala, Diyala, Iraq Hazim_numan@uodiyala.edu.iq
[4] School of Information Technology, Deakin University, Geelong Waurn Ponds Campus, Australia j.teh@deakin.edu.au
[5] College of Engineering for Artificial Intelligence Technology, University of Diyala, Diyala, Iraq dr.yasiral-zubaidi@uodiyala.edu.iq
[6] Malaysia Cryptology Technology and Management Centre (PTPKM), Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia, rezazaba@upm.edu.my
[7] College of Education for Pure Science, University of Diyala, Diyala, Iraq noor.hassan@uodiyala.edu.iq

Abstract. Neural cryptanalysis has emerged as a promising technique for identifying statistical patterns in symmetric primitives, particularly in reduced-round configurations. Despite a growing body of work demonstrating the effectiveness of neural distinguishers, there remains a lack of a systematic method to quantify the learnability of ciphertext structures. In this work, we introduce the *Neural Attackability Index (NAI)*, a metric that evaluates the susceptibility of AEAD schemes to machine-learning-based distinguishers under a fixed-key model. The NAI combines three parameters, distinguishability advantage, convergence speed, and feature attribution, into a single score that reflects structural learnability. We instantiate the NAI metric on three representative lightweight AEAD schemes: Ascon-128, GIFT-COFB, and Grain-128AEAD. Our evaluation shows a consistent degradation of NAI with increasing rounds, that supports the hypothesis that neural learnability is closely related to early-round diffusion. Furthermore, we use SHAP-based attribution to identify which ciphertext bits have the strongest influence on the neural distinguisher. Our results show that when only a small number of bits dominate the attribution, the cipher tends to be more vulnerable to neural attacks. Our experiments also show that Ascon-128 exhibits strong neural bias in very early rounds, while GIFT-COFB shows weaker but still detectable statistical bias. Under the reduced-round configuration, Grain-128AEAD produces the strongest signal, achieving high distinguishability advantage ($\delta_{\max} \approx 0.488$), near-perfect separability (AUC ≈ 0.996), and concentrated SHAP attribution over a subset of ciphertext bytes. Based on these observations, we formulate structural propositions linking neural convergence behaviour to diffusion coverage and attribution concentration, and discuss design considerations for reducing learnable signal in ciphertext outputs. We conclude by positioning NAI as a metric for evaluating structural learnability in symmetric designs beyond what is captured by traditional cryptanalytic methods.

Keywords: Neural cryptanalysis · Neural Attackability Index · SHAP explainability · Structural leakage · Machine learning-based distinguishers

1 Introduction

Authenticated Encryption with Associated Data (AEAD) is a cryptographic primitive for securing data in bandwidth- and power-constrained environments, including embedded systems, wireless sensor networks, and the Internet of Things (IoT). AEAD schemes provide both confidentiality and authenticity in a single algorithm to ensure message integrity and resistance against replay and forgery attacks. The CAESAR competition [1], followed by the NIST Lightweight Cryptography (LWC) project [2], provided avenues for the design and analysis of several lightweight AEAD schemes such as Ascon [9], GIFT-COFB [5], TinyJAMBU [29], Elephant [10], and Romulus [18].

These schemes reflect diverse structural paradigms: sponge-based, tweakable block cipher (TBC)-based and feedback-based designs. A lot of cryptanalytic efforts have been done to assess their structure weaknesses using classical techniques such as differential and linear cryptanalysis [13, 22, 28], cube and algebraic attacks [8, 30], fault attacks [20, 23], and forgery distinguishers [15, 3]. For instance, over 20 attack variants have been proposed on Ascon [21], the NIST LWC winner, across different rounds and models.

More recently, machine learning (ML)-based distinguishers have emerged as a new class of cryptanalytic tools in symmetric cryptanalysis. Gohr's seminal work [14] showed that neural networks can learn structural leakage from reduced round versions of Speck and achieve higher distinguishability than conventional differential attacks. Subsequent works extended this approach to block ciphers such as PRESENT [19], Simon [24], and AEAD primitives including Ascon and GIFT-COFB [12, 27]. They showed that convolutional networks can detect learnable features in ciphertext distributions that arise from weak early-round diffusion or biased output mappings. In the context of neural cryptanalysis, neural vulnerability describes the learnability of statistical patterns that a cryptographic primitive displays and can be used by a distinguisher based on a neural network[14, 12]. It offers an alternative approach to traditional cryptanalysis, focused on the vulnerability to data-driven analysis.

Despite these advances, the field lacks a methodology to quantify the *neural vulnerability* of symmetric cryptographic primitives such as AEAD schemes. Existing approaches typically evaluate performance using classification accuracy, convergence time, or by identifying salient bits. However, these approaches are dependant to architectural choices, preprocessing steps or hyperparameter tuning, and therefore it makes cross-cipher comparisons unreliable. In addition to that, high test accuracy alone does not necessarily indicate structural leakage unless contextualized with explainability metrics [25] or adversarial generalization capacity [17]. Consequently, there is a need for a common metric to measure and compare neural attackability of the various AEAD constructions in a consistent and interpretable manner.

Adversarial Model and Scope. We consider a setting in which a passive adversary attempts to distinguish structured ciphertexts, those produced from fixed input differences, from random ciphertexts using a neural network trained under a fixed-key model. This setting captures *structural learnability* rather than classical cryptanalytic attacks. Our goal is not to forge messages, recover keys, or mount chosen-ciphertext attacks, but to study whether neural models can detect internal structural dependencies in the cipher's round function, following the direction explored in neural cryptanalysis [14, 12, 7, 19].

Our analysis focuses on early-round behaviour, where diffusion is still incomplete and structural differences between cipher designs are more visible. Similar to early-round studies in classical differential cryptanalysis, this setting allows statistical patterns to be observed before full diffusion is reached.

Contribution

This work presents the following contributions:

- We introduce the *Neural Attackability Index (NAI)*, a metric that quantifies the susceptibility of AEAD schemes to machine-learning-based distinguishers. The NAI combines three components: distinguishability advantage, convergence speed, and feature attribution.
- We instantiate the NAI metric on three structurally distinct AEAD schemes: Ascon-128, GIFT-COFB, and Grain-128AEAD [16]. For each scheme, we train neural distinguishers under a differential setting, compute SHAP-based attribution maps and derive empirical NAI scores.
- Our analysis shows a clear relationship between early-round diffusion and neural learnability. Ascon exhibits strong neural bias in 2-round configurations with sparse attribution patterns, GIFT-COFB shows weaker but still observable leakage, while Grain-128AEAD produces the highest NAI score among the evaluated ciphers, indicating stronger resistance to neural distinguishers.
- We propose two structural theorems, the *Early-Round Vulnerability Theorem* and the *Structural Sensitivity Theorem*, which relate neural convergence behaviour to how quickly diffusion spreads and how the signal is concentrated over ciphertext bits.
- Finally, we discuss the implications of neural attackability for cipher design and outline how the NAI can support future cryptographic evaluation and standardization processes.

Organization of the Paper

The remainder of this paper is organized as follows. Section 2 introduces notation, adversarial models, and evaluation metrics. Section 3 defines the Neural Attackability Index and its computational methodology. Section 4 presents the experimental configuration, including cipher parameters, neural architectures, and training strategies. Section 5 reports NAI measurements across selected AEAD schemes and interprets them using SHAP-based attribution. Section 6 discusses binary distinguishability as an adversarial decision problem. Section 7 discusses the implications and guidelines of the design. Section 8 concludes the paper.

2 Preliminaries and Notation

In this section, we provide the foundation and notation used throughout our analysis. We begin by introducing the AEAD security model and notation, followed by a formalization of the machine learning-based adversarial model. We also describe the neural network architecture, training configuration, and evaluation metrics used in computing the Neural Attackability Index (NAI).

2.1 Authenticated Encryption with Associated Data (AEAD)

An AEAD scheme $\Pi = (\mathsf{Enc}, \mathsf{Dec})$ consists of two algorithms defined over a key space $\mathcal{K}$, nonce space $\mathcal{N}$, associated data space $\mathcal{A}$, message space $\mathcal{M}$, and ciphertext space $\mathcal{C}$. Given a secret key $K \in \mathcal{K}$, a nonce $N \in \mathcal{N}$, associated data $A \in \mathcal{A}$, and a message $M \in \mathcal{M}$, the encryption function produces a ciphertext $C = \mathsf{Enc}_K(N, A, M) \in \mathcal{C}$. The decryption function $\mathsf{Dec}_K(N, A, C)$ returns either the original message M or a failure symbol $\perp$.

We focus on AEAD schemes of different design paradigms: permutation-based (Ascon [9]), tweakable block cipher-based (GIFT-COFB [5]) and feedback-based designs (Grain-128AED [16]. These ciphers differ significantly in their internal state structure and diffusion behavior that make them suitable targets for comparative analysis under neural distinguishability attacks.

2.2 Neural Cryptanalysis Framework

Following the machine learning based cryptanalytic paradigm introduced by Gohr [14] and extended by Gerault et al. [12], we consider an adversary who trains a neural network to distinguish between ciphertexts produced from structured (differential) and random plaintext inputs.

Let $\mathcal{D}_0$ denote the distribution of ciphertexts obtained from random plaintext pairs encrypted under a fixed key K, and $\mathcal{D}_1$ denote the distribution of ciphertexts from structured plaintext pairs with a fixed XOR difference Δ. The adversary receives labeled samples (C_i, y_i) drawn from $\mathcal{D}_0 \cup \mathcal{D}_1$, where $y_i = 1$ if $C_i \sim \mathcal{D}_1$, and $y_i = 0$ otherwise.

The neural model f_θ with parameters θ is trained to solve a binary classification problem. The loss function minimized during training is the binary cross-entropy:

$$\mathcal{L}(\theta) = -\sum_i [y_i \log f_\theta(C_i) + (1 - y_i) \log (1 - f_\theta(C_i))] \tag{1}$$

The goal of the adversary is to find a model f_θ such that the classification accuracy exceeds the random guessing threshold of 0.5 on unseen data.

2.3 Neural Architecture and Training Setup

Convolutional Neural Networks (CNNs) are a model of deep learning neural network that automatically extracts hierarchical feature representations from structured input data by using convolution and pooling. CNNs are also effective in identifying subtle statistical dependencies and distinguishable patterns in cryptographic outputs, making them suitable for neural cryptanalysis[4]. For our experiments, we use a lightweight convolutional neural network (CNN) architecture designed for bit-level cryptanalytic learning [14, 12]. This architecture is used to learns discriminative features from ciphertext–tag representations, eliminating the need for manual feature engineering. The architecture includes three 1D convolutional layers followed by two fully connected layers. Each convolutional layer is followed by a ReLU activation function to enhance training stability and convergence. The final layer outputs logits, and binary classification is performed using the `BCEWithLogitsLoss` objective. The ciphertext is encoded at the bit level, so that an input dimension proportional to the ciphertext length.

The models are trained using the Adam optimizer with a learning rate of 0.001 and a batch size of 256. Convergence is determined in this way: the convergence epoch E_c is defined as the first epoch at which the distinguishability advantage $\delta(e) = \mathrm{Acc}(e) - 0.5$ exceeds 0.05. Each experiment is repeated across multiple random seeds to account for variability due to random initialization and data shuffling.

2.4 Dataset Generation

For each AEAD scheme, we generate datasets by encrypting n pairs of plaintexts under a fixed key K. Half of the pairs (M, M') satisfy $M' = M \oplus \Delta$ (differential samples), and the other half are independently chosen (random samples). Ciphertexts are computed under a fixed nonce N and labeled accordingly. A fixed nonce value is used to isolate the internal structural effects of the cipher from the randomness provided by that value.

For each plaintext pair, we compute the corresponding ciphertext–tag outputs $(C \parallel T)$ and $(C' \parallel T')$. The neural input is $X = (C \parallel T) \parallel (C' \parallel T')$ which constructed as the concatenation, where C and C' present ciphertexts and T and T' represent authentication tags. The outcome is converted into the bit level representation and fed into a neural model. We denote the bitwise ciphertext–tag representation as $C_i \in \{0,1\}^m$, where m is determined by the ciphertext length of the scheme.

We fix $n = 100{,}000$ for training and $n = 20{,}000$ for testing per cipher configuration, to give balanced class labels in each split.

2.5 Notation and Metrics

Throughout this paper, we use the following notation and evaluation criteria:

- f_θ: Trained neural classifier with parameters θ.
- The distinguishability advantage measured at epoch e, computed as the validation accuracy deviation from random guessing, is defined as

$$\delta^{(e)} = \Pr[f_\theta(C_i) = y_i] - 0.5 \tag{2}$$

- The maximum distinguishability advantage achieved during training is defined as

$$\delta_{\max} = \max_e \delta^{(e)} \tag{3}$$

- The convergence epoch, defined as the first epoch at which the distinguishability advantage exceeds 0.05, is given by

$$E_c = \min \left\{ e \mid \delta^{(e)} > 0.05 \right\} \tag{4}$$

- Let ϕ_j denote the absolute mean SHAP attribution of ciphertext feature j. We define the threshold τ as the 90th percentile of $\{\phi_j\}$.
- The Feature Sensitivity Index (FSI), computed as the number of ciphertext features whose SHAP attribution exceeds the 90th percentile threshold, is defined as

$$\text{FSI} = |\{j \mid \phi_j > \tau\}| \tag{5}$$

  SHAP values in our implementation are first computed at the bit level and then aggregated per ciphertext byte. Therefore, FSI is evaluated at the byte level.

- NAI: The Neural Attackability Index, formally defined in Eq. (6) in Section 3.

We denote a plaintext difference by Δ, ciphertexts by C, and the number of cipher rounds by r. All keys and nonces are drawn uniformly at random unless explicitly fixed.

The schemes selected for evaluation are:

- **Ascon-128** [9]: A NIST-standardized AEAD scheme based on a 320-bit permutation with sponge based duplexing. Encryption consists of 12 permutation rounds, divided across initialization, message absorption, and finalization phases.
- **GIFT-COFB** [5]: An AEAD scheme built from the GIFT-128 tweakable block cipher, employing a COFB (Ciphertext Offset Feedback) mode of operation that combines permutation and feedback chaining with provable bounds under nonce misuse.
- **Grain-128AEAD** [16]: A lightweight stream cipher based AEAD design derived from the Grain family, incorporating nonlinear feedback shift registers and internal MAC computations. Its bitwise operation model differs significantly from permutation-based ciphers, offering complementary structural features.

2.6 SHAP-Based Attribution

SHapley **A**dditive ex**P**lanations (SHAP) is a technique initially generated to clarify the conduct of machine learning models by giving a sum of worth to each input features that is relative to the effect it had on the model output [25]. In this study, SHAP is not utilized as a cryptanalytic method, but as a tool which is directed towards the determination of which components of the ciphertext neural distinguishers are using. Local structural dependencies are reflected as concentrated SHAP patterns, whereas good diffusion or randomness results in more uniform SHAP patterns.

3 Neural Attackability Index (NAI)

This section presents the *Neural Attackability Index (NAI)*, a metric designed to measure the vulnerability of lightweight AEAD schemes to neural network-based distinguishers. The NAI provides an interpretable score that reflects the learnability of the structural properties of a cipher from ciphertexts under controlled differential conditions.

3.1 Definition of the NAI

Following the notation adopted in the previous section, the Neural Attackability Index (NAI) is a measure that quantifies the neural distinguishability of a lightweight AEAD construction and structural learnability of a lightweight AEAD construction.

Cipher C evaluated under an input difference Δ, the NAI index is defined as

$$\mathrm{NAI}(C, \Delta) = \frac{\delta_{\max} \cdot \mathrm{FSI}}{E_c} \tag{6}$$

If $E_c = \infty$, we define $\mathrm{NAI} = 0$.

The NAI integrates three important aspects of neural vulnerability:

- **Statistical Separability** ($\delta_{\max}$): Larger value indicates clearer statistical bias in ciphertext distribution. It captures the maximum deviation from random guessing achieved by the neural distinguisher.
- **Learning Efficiency** (E_c): Smaller values indicate that the structural bias is easier to learn. It measures how quickly this separability is achieved during training.
- **Feature Sensitivity Index** (FSI): Measures the number of ciphertext features that contribute to the classification decision to indicate the extent of structural dependence exploited by the neural model. This is computed using SHAP.

These terms are combined multiplicatively, the index chooses constructions that are easy to distinguish, quick to learn, and depend on a subset of ciphertext features. A high NAI value indicates higher neural attackability, while a low value indicates better structural robustness.

The motivation behind NAI is that the accuracy of a classifier does not necessarily reflect neural exploitability. Therefore, NAI incorporates three properties: $\delta_{\max}$, which measures statistical separability; E_c, which reflects learning efficiency; and FSI, which quantifies the ciphertext features. When combined, these properties provide a more comprehensive characterization of neural learnability than classification accuracy alone.

To compare, we normalize the NAI score relative to the highest observed score in a given batch. Let $\{C_1, \ldots, C_m\}$ be a set of AEAD schemes evaluated under the same architecture and Δ. Algorithm 1 illustrates the general procedure for calculating the Neural Attackability Index.

Algorithm 1 Computation of Neural Attackability Index (NAI)

Require: Cipher C, difference Δ, model f_θ, attribution method A, percentile threshold τ
Ensure: $\mathrm{NAI}(C, \Delta)$

1: Generate training dataset $D = \{(C_i, y_i)\}$ using C and Δ
2: Train f_θ on D; for each epoch e compute $\delta^{(e)} = \mathrm{val\_acc}(e) - 0.5$
3: Set $\delta_{\max} \leftarrow \max_e \delta^{(e)}$
4: Determine convergence epoch $E_c \leftarrow \min\{e \mid \delta^{(e)} > 0.05\}$
5: Compute attribution scores $\phi_i = A(f_\theta, C_i)$ for each input bit i
6: Aggregate bit-level attributions into byte-level scores $\phi_j \leftarrow \sum_{i \in \text{byte } j} |\phi_i|$
7: Set percentile threshold $\tau \leftarrow$ 90th percentile of $\{\phi_j\}$
8: Identify sensitive bytes $B \leftarrow \{j \mid \phi_j \geq \tau\}$
9: Set FSI $\leftarrow |B|$
10: **if** $E_c = \infty$ **then**
11: **return** 0
12: **end if**
13: **return** $\mathrm{NAI}(C, \Delta) = \frac{\delta_{\max} \cdot \mathrm{FSI}}{E_c}$

The attribution method A is DeepSHAP [25] which is a method for deep neural networks for approximating the contribution of input features to the output of a neural classifier (Algorithm 1). The parameter ϕ_i of the attribute is the mean contribution of the ith byte of the ciphertext to the output of the classifier. The threshold, denoted by the variable τ, is a fixed percentage of attribution distribution (e.g., 90th percentile) that is applied to determine the bits that are the most sensitive and whose contribution is large enough to be disproportionate of the average performance. The e in line 3 denotes the training epoch, and E_c is the first epoch at which the model achieves a classification accuracy above random guessing by more than 0.05. The threshold $\delta^{(e)} > 0.05$ indicates that the classification accuracy is at least 5 percentage points higher than random guessing. This value provides a benchmark for determining when a distinguishable signal has emerged during training, while reducing the influence of random fluctuations that may occur in the early stages of the learning process.

$$\mathrm{NAI}_{\mathrm{rel}}(C_j) = \frac{\mathrm{NAI}(C_j, \Delta)}{\max_{1 \leq i \leq m} \mathrm{NAI}(C_i, \Delta)} \tag{7}$$

This results in a normalized value $\mathrm{NAI}_{\mathrm{rel}} \in [0, 1]$ to facilitate comparative analysis and visualizations such as attackability profiles.

A high NAI score signals a configuration in which neural distinguishers:

- Achieve statistically significant distinguishability ($\delta_{\max}$),
- Converge quickly through training (small E_c),
- Base decisions on a structurally non-trivial subset of ciphertext features (high FSI).

The NAI index does not indicate a failure of cryptographic security in the traditional sense of IND-CPA or INT-CTXT, but it exposes structural imbalances that neural models can exploit. This makes it particularly useful for evaluations.

Consider two examples of AEAD, C_A and C_B:

$$\begin{aligned} C_A: &\quad \delta_{\max} = 0.08,\ E_c = 3,\ \mathrm{FSI} = 15 \Rightarrow \mathrm{NAI} = 0.4 \\ C_B: &\quad \delta_{\max} = 0.06,\ E_c = 6,\ \mathrm{FSI} = 8 \Rightarrow \mathrm{NAI} = 0.08 \end{aligned}$$

Although both are secure under classical assumptions, the NAI indicates that C_A is easier for a neural model to distinguish. This suggests that C_A may require further analysis or additional rounds to improve its resistance.

4 Experimental Setup

This section describes the steps to compute the Neural Attackability Index (NAI) for three AEAD schemes: GIFT-COFB, Ascon-128, and Grain128-AEAD. We detail the problem definition, dataset generation process, model architecture, training protocol, and SHAP-based attribution analysis.

All experiments are conducted in a fixed-key setting, consistent with prior work [14, 12], to isolate structural biases from key-related randomness. The key is fixed and plaintext pairs follow a controlled difference Δ, so any observed distinguishability reflects structural properties of the cipher.

4.1 Problem Definition

The goal of this work is to quantify the *learnability* of ciphertext structures by neural models, without assuming key recovery or forgery capabilities. Specifically, we ask: to what extent can a passive adversary distinguish ciphertexts generated from structured differentials versus random pairs under a fixed secret key?

Adversarial Model. We assume a passive adversary with known plaintext and ciphertext access, where the encryption is performed under a fixed and randomly chosen secret key K. The key remains fixed across dataset generation to isolate structural biases. For nonce-based schemes, we fix the nonce across all samples. The adversary's goal is to build a classifier $f : \mathcal{C} \to \{0, 1\}$ that can correctly distinguish the origin of the ciphertext based on whether it was produced from a pair of related or unrelated plaintexts.

Let $\mathcal{C}$ be an AEAD cipher embodied with fixed key K, and let Δ be a plaintext difference. The adversary is provided ciphertexts C_i and labels y_i where:

- $y_i = 1$ if C_i results from encrypting a pair $(M, M \oplus \Delta)$.
- $y_i = 0$ if C_i results from two independent messages.

The adversary constructs a classifier f such that:

$$\Pr[f(C_i) = y_i] > \frac{1}{2}.$$

The evaluation metric is the model's advantage over random guessing, measured on a separate test set.

$$\delta^{(e)} = \text{val\_acc}(e) - 0.5$$

4.2 Dataset Generation

Each encryption configuration produces a fixed size of categorized ciphertext samples (120,000 per round) with a fixed secret key and a fixed nonce value. The dataset is balanced with half structured plaintext that are of fixed input difference 0 and the other half random plaintext pairs. Structure samples are generated as plaintext pairs(M, M') using a fixed non-zero XOR input difference Δ, such that $M' = M \oplus \Delta$.In all experiments, Δ is defined as a 16-byte vectore whose last byte is $00xFF$. Random samples are generated from two independent and uniformly sampled plaintext messages. The size of the training sample is 100,000 and the size of the test sample is 20,000.

The fixed-key and fixed-nonce configuration has been intentionally chosen to isolate cipher-internal structural effects and evaluate neural distinguishability under controlled conditions. Fixed-parameter settings are widely used in neural cryptanalysis to assess distinguishability in the absence of nonce variation. The results should therefore be interpreted as diagnostic indicators of neural learnability rather than practical attack capabilities.The data for:

- **GIFT-COFB** and **Ascon-128** was generated using Python wrappers interfacing with their respective C reference implementations.
- **Grain128-AEAD** required integration using CFFI and a custom wrapper around the LWC reference code[26].

The plaintext, ciphertext, and authentication-tag lengths used in the experiments follow the specifications and reference implementations of the corresponding AEAD schemes. Since the input dimensionality supplied to the neural distinguisher is determined by the output structure of each cipher, it is not treated as an externally configurable parameter.

The data were stored in the form of a `pkl` file. Each record contains the ciphertext-generated feature vector (bit-expanded in training) and its binary class label. Depending on the experimental setup, ciphertext vectors can contain concatenated outputs (e.g., multiple ciphertext blocks and/or authentication tags).

4.3 Neural Architecture and Training

We use a 1D convolutional neural network (CNN) with the following configuration:

- **Input layer**: Ciphertext input converted first to bit-level representation and provided as a 1D sequence of length $8n$, where n is the ciphertext length in bytes.
- **Feature extraction layers**: Three 1D convolutional layers with channel sizes 32, 64, and 64 respectively, each followed by ReLU activation.
- **Hidden layers**: Two dense layers with 64 and 32 neurons respectively, each followed by ReLU activation.
- **Output layer**: A single neuron with sigmoid activation for binary classification.

Training is performed in PyTorch using the Adam optimizer, with the following settings:

- **Learning rate**: 10^{-3}, with step-wise decay (StepLR, decay factor 0.9 every epochs).
- **Epoch**: 50 with early stopping (patience = 5).
- **Batch size**: 128.
- **Validation split**: 20%.

Each model is trained on 80% of the dataset and validated on the remaining 20%. Final accuracy is described on a held-out test set of 20,000 samples per round configuration.

Experiments are performed across multiple reduced-round configurations to observe how neural distinguishability changes as the number of rounds increases. As the number of rounds increases, ciphertext distributions become closer to random. This makes it harder for the neural model to distinguish them and reducing SHAP attribution values. This is consistent with earlier work by Gohr et al. [14], which shows that early rounds reveal more structural bias, while additional rounds tend to hide these patterns.

4.4 Computing the Neural Attackability Index (NAI)

The NAI is calculated using three components:

- **Maximum Distinguishability Advantage** ($\delta_{\max}$): The maximum validation-based advantage
$$\delta^{(e)} = \text{val\_acc}(e) - 0.5$$
achieved through training. This is a measure of deviation from random guessing and not raw test accuracy.

- **Convergence Epoch** (E_c): The first epoch at which $\delta^{(e)} > 0.05$, indicating statistically meaningful learnability. If such an epoch does not exist, E_c is assigned to infinity and the NAI is set to zero.
- **Feature Sensitivity Index** (FSI): The number of bytes with SHAP values above the 90th percentile. Despite the fact that FSI is reported as an absolute number of sensitive features, the threshold is calculated based on the distribution of attribution of the evaluated cipher. Therefore, FSI value reflects the strength of informative features in each cipher rather than their absolute attribution.

These measures are combined into a single value to indicate attackability across both model performance and feature attribution. Although we take the 90th percentile as a threshold to determine which bytes are considered significant in the model's decision-making process, this approach may have limitations. Since this threshold is determined by the distribution of SHAP values, it may behave differently depending on how the values are distributed. For example, even when all SHAP values are low and nearly equal, many bytes may still exceed the 90th percentile. This causes the FSI to appear larger than expected. Alternatively, when only a few bytes have very large SHAP values, those bytes may strongly influence the model, yet because they are few, the FSI may be smaller. This shows a limitation of percentile-based thresholding: it reflects how widely the model relies on different features, rather than how strong the importance of each feature is.

4.5 SHAP-Based Attribution Analysis

SHAP DeepExplainer[25] is used to understand the contribution that specific ciphertext bytes had on the model decision. DeepSHAP is used in the trained PyTorch model with a background set of training data and an explanation set of validation data. For each model:

- We sample 512 ciphertexts from the training set as background data and 200 ciphertexts from the validation set for explanation.
- SHAP values are computed at the bit level and absolute values are averaged across all explained samples.
- Top 5 bytes by mean absolute SHAP value are identified.

These bytes indicate where distinguishable structure appears in the ciphertext. Since the classifier uses a bit-level representation of the ciphertext, attribution analysis helps us link the learned patterns to specific locations in the ciphertext. High SHAP values concentrated in a small number of bytes mean limited diffusion or biased state propagation. In contrast, low and evenly distributed SHAP values indicate stronger diffusion and lower structural learnability.

5 Results and Findings

Through this section, we examine how the NAI varies among three lightweight AEAD schemes: Ascon-128, GIFT-COFB and Grain-128AEAD. Our evaluation tests reduced-round configurations of each cipher under a fixed-key differential setting. We describe the results of raw and normalized NAI scores, feature-level attribution, and interpret the neural distinguishers' behavior using SHAP-based explainability.

Table 1 shows the measured NAI averaged over three independent training runs. For evaluation at 5 rounds, Ascon-128 exhibits a high NAI compared to the evaluated schemes GIFT-COFB and Grain-128AEAD. This matches what we observe during neural evaluation: when early rounds have weaker diffusion, the model can more easily find patterns and this

leads to higher distinguishability and more features being important. As more rounds are added, these patterns are mixed more thoroughly, and that makes the ciphertext closer to random. As a result, NAI values decrease, showing that the neural model can no longer distinguish the data easily.

Table 1: NAI Scores for AEAD Schemes.

Cipher	Rounds	NAI	$\mathbf{NAI_{rel}}$
Ascon-128	5	1.29	0.88
GIFT-COFB	5	0.00	0.00
Grain-128AEAD	5	1.46	1.00

These findings validate that Grain-128AEAD (5-round configuration) exhibits the highest neural learnability (NAI) among the considered ciphers, whereas Ascon-128 shows relatively lower though still significant distinguishability.

Let $f : \mathbb{F}_2^n \to [0, 1]$ denote the trained neural classifier used to distinguish structured ciphertexts from random ones. The NAI is defined as:

$$\mathrm{NAI}(C, \Delta) = \frac{\delta_{\max} \cdot \mathrm{FSI}}{E_c}.$$

where $\delta_{\max} = \max_e (\mathrm{val\_acc}(e) - 0.5)$ represents the maximum distinguishability advantage observed during training, E_c is the first epoch at which meaningful distinguishability is observed, and FSI denotes the number of ciphertext bytes whose SHAP attribution lies in the 90th percentile or higher. This definition captures the interaction between separability strength, learning speed, and structural feature utilization, in contrast to formulations based on AUC or normalized training horizons.

When $\delta_{\max} \approx 0$, the product collapses and NAI ≈ 0, regardless of test accuracy. It means that classifier accuracy alone is not sufficient for neural exploitability, as high accuracy may arise from overfitting or dataset imbalance without sustained structural advantage.

5.1 Classifier Metrics

Table 2 shows detailed classifier-level metrics: accuracy, maximum distinguishability advantage $\delta_{\max}$, SHAP-derived Feature Sensitivity Index (FSI), convergence epochs E_c, and the Area Under Curve (AUC).

Table 2: Classifier Metrics Across AEAD Schemes (Reduced-Round).

Cipher	Rounds	Accuracy	$\boldsymbol{\delta_{\max}}$	FSI	$\boldsymbol{E_c}$	AUC
Ascon-128	5	0.984	0.484	8	3	0.997
GIFT-COFB	5	0.500	0.000	11	∞	0.502
Grain-128AEAD	5	0.988	0.488	6	2	0.996

For GIFT-COFB, the SHAP values across all bytes were near-uniform and low in magnitude, resulting in no meaningful distinguishability ($\delta_{\max} = 0$) and $E_c = \infty$. Although small attribution differences may arise due to percentile thresholding, the classifier performs nearly at random guessing (accuracy ≈ 0.5 and AUC ≈ 0.5), indicating that there is no structural signal to learn. This tells that the model does not extract differential bias under the evaluated configuration.

On the other hand, both Grain-128AEAD and Ascon-128 show high distinguishability ($\delta_{\max}$), very strong AUC values (> 0.99), and non-zero FSI, indicating the presence of

learnable structure in reduced-round settings. Grain-128AEAD achieves the highest NAI due to fast convergence ($E_c = 2$) and consistent attribution across certain ciphertext bytes. Ascon-128 also shows strong neural learnability, but requires slightly more training to converge.

These results show that classification accuracy alone is not enough to measure neural attackability. Instead, it should be considered together with distinguishability, AUC, convergence speed, and how attribution is distributed. Although these experiments are performed on reduced-round versions, the goal is to understand whether the observed distinguishability comes from stable structural patterns rather than random effects.

5.2 Explainability via SHAP Attribution

In this work, SHAP is used as an interpretability tool to identify which parts of the ciphertext have the strongest influence on neural distinguishability. Figure 1 shows the SHAP mean importance plot of Grain-128AEAD (5-round configuration) and shows the 20 most significant ciphertext bytes by their average SHAP values. The plot indicates a focused attribution scheme where certain collections of bytes are having disproportionately significant impact to the output of the classifier. This indicate that Grain-128AEAD, under reduced-round evaluation, exhibits localized structural dependencies that are strongly exploitable by neural models. In contrast to a diffuse attribution profile, this focus implies that there are certain transitions of internal state which have a greater contribution to the difference in distinguishability.

This behaviour can be explained by the bit-serial structure of Grain, where limited mixing in early rounds can create localized patterns that are easier for the neural model to learn. The high $\delta_{\max} \approx 0.488$, strong AUC ($> 0.996$), and fast convergence ($E_c = 2$) together explain the high NAI observed in this setting.

Rather than indicating robustness, these results show that reduced-round Grain-128AEAD can be learned by the neural model under fixed-key differential evaluation, especially when diffusion is not sufficient to hide structural dependencies.

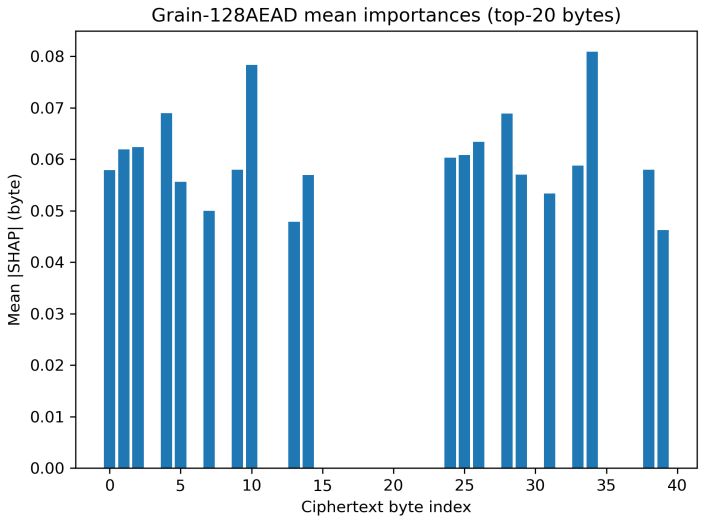


Figure 1: Grain-128AEAD (5-round) Mean SHAP Importances (Top 20 Bytes).

Figure 2 compares the top five SHAP-attributed ciphertext bytes for Ascon-128 and GIFT-COFB under 5-round configurations. Ascon exhibits higher SHAP values across its most influential bytes, indicating that a small subset of ciphertext positions strongly

influences the neural classifier. This is consistent with its high distinguishability ($\delta_{\max} \approx 0.484$) and moderate FSI = 8.

In contrast, GIFT-COFB does not show a clear attribution structure, as its classifier fails to achieve distinguishability ($\delta_{\max} = 0$, $E_c = \infty$). The SHAP values are low and spread more evenly. It means that the model cannot rely on specific ciphertext locations i.e., the ciphertext is closer to random under this configuration, making it harder for the neural model to learn useful patterns.

From these results, we found that when only a few ciphertext bytes dominate the attribution, the cipher is easier for the neural model to distinguish; when attribution is weak and spread out, neural distinguishability is reduced.

- The early-round structure of Ascon results in concentrations of features which are local and learnable.
- The structure of GIFT-COFB does not cause any statistically exploitable bias due to neural discrimination.

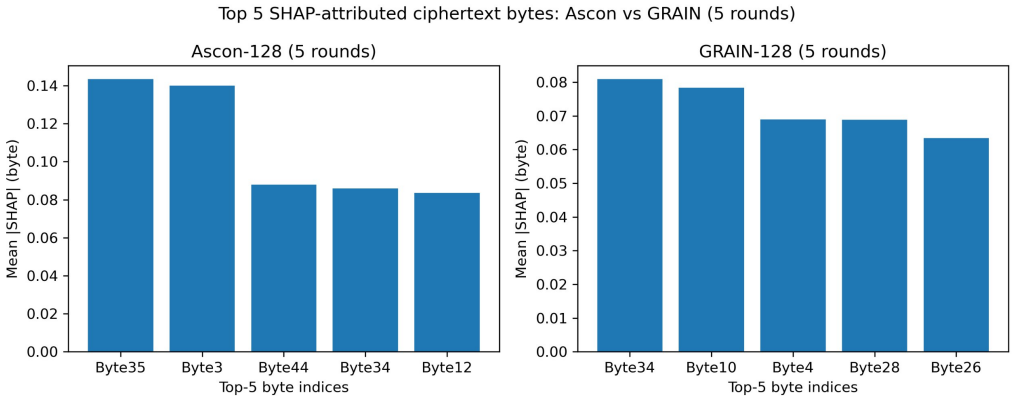


Figure 2: Top 5 SHAP attributed ciphertext bytes: Ascon vs Grain-128 (5-rounds).

5.3 Discussions

The experimental results across the three AEAD schemes enable interpretation of their early-round robustness under neural distinguishers:

- **Ascon-128 (5 rounds):** The high Neural Attackability Index (NAI = 1.29), a strong distinguishability advantage $\delta_{\max} = 0.484$, and a moderate Feature Sensitivity Index (FSI = 8) indicate that Ascon's early rounds reveal statistically learnable structure. The SHAP analysis shows a tight concentrating effect of the attribution to a subgroup of ciphertext bytes, showing local structure dependency. The convergence epoch ($E_c = 3$) implies that the neural model requires limited, though non-trivial, training to capture this bias. These findings shows that, even in reduced-round evaluation, the permutation structure of Ascon has learnable differential traces before complete diffusion.
- **GIFT-COFB (5 rounds):** GIFT-COFB does not exhibit measurable distinguishability ($\delta_{\max} = 0$), with AUC = 0.5, no convergence ($E_c = \infty$), and NAI = 0. SHAP attributions are low and fairly consistent across ciphertext bytes i.e., lack of a learnable pattern. This means that under the same fixed-key differential condition, GIFT-COFB achieves effective early diffusion such that neural models cannot extract structural bias.

- **Grain-128AEAD (5 rounds):** Grain-128AEAD achieves the highest NAI (1.46), with high distinguishability ($\delta_{\max} = 0.488$) and fast convergence ($E_c = 2$), while relying on a relatively small set of influential bytes (FSI = 6). The SHAP distribution shows that attribution is concentrated on specific byte positions, consistent with localized dependencies from its feedback-based structure.

To further quantify how attribution is distributed across the ciphertext output, we introduce a metric derived from SHAP scores.

$$\phi_i = \mathbb{E}[\,|\mathrm{SHAP}_i|\,]$$

Let be the mean absolute attribution of ciphertext byte i, and define the attribution sparsity α as

$$\alpha = \frac{1}{n}\,|\{i \mid \phi_i \geq \tau\}| \tag{8}$$

where n is the ciphertext length in bytes and τ is the 90th percentile SHAP score. This metric measures the proportion of ciphertext bytes that contribute to the neural model's decision.

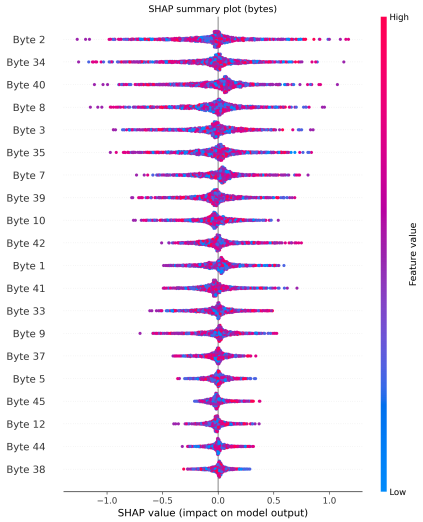
The SHAP scores for AScon-128, GIFT-COFB and Grain-128 are given below:

- Ascon-128 (5R): $\alpha \approx 0.083$
- GIFT-COFB (5R): $\alpha \approx 0.115$
- Grain-128 (5R): $\alpha \approx 0.125$

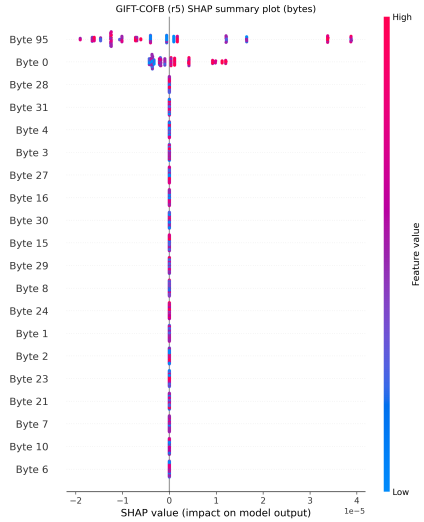
This shows that neural exploitability is not directly proportional to attribution sparsity (α). Although Grain-128A has the highest sparsity ($\alpha \approx 0.125$), followed by GIFT-COFB ($\alpha \approx 0.115$) and Ascon-128 ($\alpha \approx 0.083$), only configurations with non-zero distinguishability advantage ($\delta_{\max} > 0$) gives good NAI scores. Specifically, both Grain-128AEAD and Ascon-128 produce positive $\delta_{\max}$ values and therefore non-zero NAI, whereas GIFT-COFB has $\delta_{\max} = 0$ and NAI = 0, despite exhibiting sparse attribution due to percentile-based thresholding.

Figure 3 presents the visualizations of the three ciphers Ascon-128, GIFT-COFB, and Grain-128AEAD using SHAP. These include summary plots and mean-importance bar charts obtained using DeepSHAP. Ascon-128 shows concentrated attribution on a subset of important ciphertext features, GIFT-COFB shows weak and dispersed attribution that aligns with near-random distinguishability, while Grain-128AEAD shows very strong attribution on a few, important ciphertext features, as shown by its high AUC and NAI values.

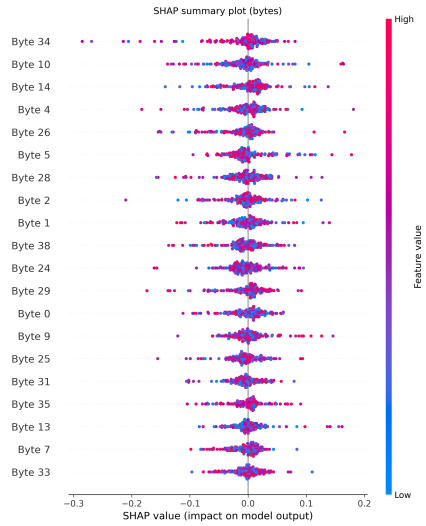
This comparison of AUC and FSI values further shows that the number of sensitive features is not the only factor that makes a neural distinguishable. The FSI value for GIFT-COFB is the highest, but the AUC is relatively low and is close to a random guess, suggesting that these features attributed to GIFT are not useful for discrimination information. However, Ascon-128 and Grain-128AEAD obtain near-perfect AUC values with smaller FSI values, indicating that distinguishability does not depend on the number of attributed features alone but also on a small number of highly informative features.



(a) Ascon-128 summary plot.



(b) GIFT-COFB summary plot.



(c) Grain-128 summary plot.

Figure 3: SHAP-based interpretation panels for three AEAD schemes under 5-round configurations.

6 NAI as a Binary Decision Tool

Based on this interpretation, we study distinguishability as a binary decision problem, without introducing additional assumptions. To evaluate the Neural Attackability Index (NAI), we define a task where a neural classifier must distinguish between ciphertexts generated from plaintext pairs with a fixed difference and ciphertexts generated from independently chosen plaintext pairs, both under the same fixed key. This setting is similar in spirit to the ND-RAND game [6], but is adapted to a fixed-key, reduced-round neural analysis setting rather than a full adaptive security model.

Let $\mathcal{C}$ denote an AEAD scheme instantiated with a fixed key K. The adversary is given

a ciphertext C and must determine whether:

- C was produced from the encryption of a structured plaintext pair $(M, M \oplus \Delta)$ using the fixed-key instantiation $\mathcal{C}_K$.
- C was generated from randomly sampled plaintexts under the same fixed key K.

The adversary is modeled as a neural classifier f_θ that outputs a score $s(C) \in [0, 1]$. A higher score indicates that the ciphertext is more likely to come from structured input. The classifier is trained on labeled samples (C_i, y_i), where $y_i \in 0, 1$ denotes structured or random ciphertexts.

To evaluate the adversary's performance, we use standard binary classification metrics [11]:

- **True Positive Rate (TPR)**: Fraction of structured ciphertexts correctly identified.
- **False Positive Rate (FPR)**: Fraction of random ciphertexts misclassified as structured.
- **ROC Curve**: Plot of TPR vs. FPR across threshold values.
- **AUC**: Area under the ROC curve, representing the probability that a structured ciphertext scores higher than a random one.

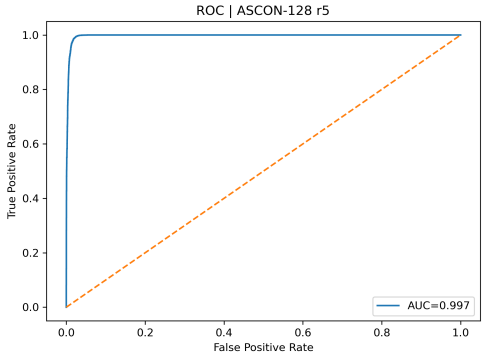
$$\mathrm{AUC}(f_\theta) = \Pr(s(C_i) > s(C_j) \mid y_i = 1,\, y_j = 0)$$

AUC: Area under the ROC curve, representing the probability that a structured ciphertext scores higher than a random one.

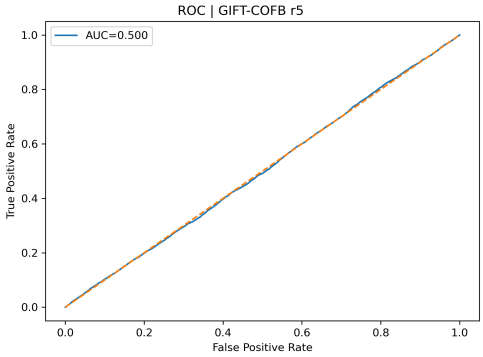
$$\mathrm{AUC}(f_\theta) = \Pr(s(C_i) > s(C_j) \mid y_i = 1,\, y_j = 0)$$

The Area Under the ROC Curve (AUC) measures the probability that the classifier assigns a higher score to a structured sample than to a random sample. An AUC value of 0.5 corresponds to random guessing, while higher values indicate better separation between structured and random ciphertexts. This is depicted in Figure 4.

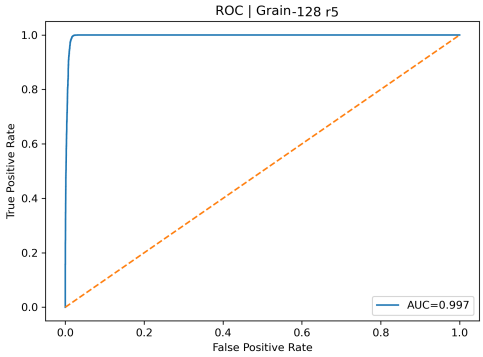
Table 3 shows the AUC values for the three AEAD schemes. These values were extracted from the neural classifier's final confidence scores evaluated on 20,000 test samples per class.



(a) Ascon-128 ROC curve.



(b) GIFT-COFB ROC curve.



(c) Grain-128AEAD ROC curve.

Figure 4: Empirical ROC curves for the evaluated schemes under 5-round configurations.

Table 3: Classifier AUC scores across AEAD schemes under 5-round configurations.

Cipher	Rounds	AUC
Ascon-128	5	0.997
GIFT-COFB	5	0.5
Grain-128AEAD	5	0.996

The results show that:

- Ascon-128 exhibits strong learnable structure at 5 rounds, with near-perfect separability (AUC ≈ 0.997).
- GIFT-COFB is not distinguishable in this setting, with (AUC ≈ 0.5), indicating no learnable signal.
- Grain-128AEAD also shows strong neural separability (AUC ≈ 0.996).

We observe that AUC closely follows $\delta_{\max}$, while neural attackability cannot be explained by attribution sparsity (α) alone. In particular, high AUC values are always associated with non-zero $\delta_{\max}$, whereas configurations with $\delta_{\max} = 0$ show near-random performance (AUC ≈ 0.5), regardless of the value of α.

The AUC values under the 5-round setting are:

- Ascon-128: $\alpha \approx 0.083$, FSI $= 8$, AUC $= 0.997$.
- GIFT-COFB: $\alpha \approx 0.115$, FSI $= 11$, AUC $= 0.500$.
- Grain-128AEAD: $\alpha \approx 0.125$, FSI $= 6$, AUC $= 0.996$.

These results show that NAI, which combines $\delta_{\max}$, E_c, and FSI, provides a more complete measure of neural distinguishability than α or AUC alone. In particular, AUC reflects separability, while FSI captures how the model relies on ciphertext features, and E_c captures how quickly this behaviour emerges during training. By combining them, they provide a clearer picture of neural attackability than any single metric. Specifically, ciphers with high distinguishability and concentrated attribution pattern have a higher NAI, while cipher with near random AUC and low concentration of attribution have a low NAI, even if classification accuracy is high. This is a confirmation that NAI is measuring something beyond just prediction performance, namely learnability.

7 Design Implications and Recommendations

The results in Section 5 and Section 5.3 show that reduced-round variants of lightweight AEAD schemes can be distinguished by neural models in some cases. The NAI is not only a way to rank ciphers based on distinguishability, but also a tool to study how design features affect neural learnability in early rounds. It helps explain observations such as where attribution is concentrated and how diffusion behaves.

In this section, we summarize these findings and provide design recommendations to reduce neural learnability in early rounds.

7.1 Key Observations from NAI Evaluation

Our analysis shows consistent patterns across the schemes, which we summarize in the following propositions.

Proposition 1 (Early-Round Neural Sensitivity)**.** *Let $\mathcal{C}_r$ be a reduced-round AEAD instance. For a fixed-key model and known input difference Δ, the Neural Attackability Index* $\mathrm{NAI}(\mathcal{C}_r)$ *tends to be higher for small values of r, and generally decreases as r increases, assuming non-trivial diffusion is present in later rounds.*

Proposition 2 (Structural Sensitivity Principle)**.** *Let f_θ be a trained neural distinguisher for an AEAD scheme $\mathcal{C}$ under a fixed key. If the classifier achieves non-trivial distinguishability advantage $\delta_{\max} > 0$, then there exists a subset $B^* \subseteq \{1, \ldots, n\}$ of ciphertext features such that:*

$$\sum_{b_i \in B^*} \phi_i > \sum_{b_j \notin B^*} \phi_j,$$

where ϕ_i denotes the mean absolute SHAP attribution of feature i. In other words, most model attribution is concentrated within B^*.

In configurations demonstrating high distinguishability (e.g., reduced-round Ascon-128), SHAP values concentrate on a subset of ciphertext bytes, yielding non-zero FSI and elevated NAI. Conversely, when $\delta_{\max} \approx 0$ (e.g., GIFT-COFB under certain configurations), attribution becomes diffuse or statistically insignificant, resulting in low or zero NAI. This suggests that structural sensitivity, as captured by SHAP concentration, is conditional upon measurable statistical separability.

7.2 Design Guidelines to Minimize Neural Attackability

Based on our findings, we suggest design guidelines to reduce neural learnability in AEAD schemes. These guidelines do not replace standard cryptographic analysis (e.g., differential or linear attacks), but serve as an additional check during early-round evaluation or post-design testing.

1. Maximize early-round diffusion. Let $\phi_r : \mathbb{F}_2^n \to \mathbb{F}_2^n$ denote the transformation after r rounds. Define the output support of a differential as:

$$\delta_r(M, \Delta) = |\mathrm{Supp}\left(\phi_r(M \oplus \Delta) \oplus \phi_r(M)\right)|\,.$$

Our results show that limited diffusion in early rounds is linked to higher $\delta_{\max}$ and NAI. For example, GIFT-COFB and Grain-128AEAD both have weak early diffusion, but differ in how quickly diffusion spreads. Designs should aim to increase $\delta_r(M, \Delta)$ as early as possible to reduce learnable structure.

2. Avoid symmetry and repetition in round transformations. Repeated or uniform operations across rounds can introduce patterns that neural models can learn. In GIFT-COFB, the similar round structure may lead to consistent feature focus in reduced-round settings, even though distinguishability still depends on $\delta_{\max}$. Using round constants, nonlinear steps, or key-dependent changes can help break this regularity and reduce learnable patterns in early rounds.

3. Minimize feature attribution concentration. Let the SHAP scores for ciphertext bits $b_1, \ldots, b_n$ be $\phi_1, \ldots, \phi_n$. Define the concentration ratio:

$$\kappa = \frac{\sum_{i=1}^{k} \phi_{(i)}}{\sum_{i=1}^{n} \phi_i} \tag{9}$$

where $\phi_{(i)}$ are the top-k SHAP scores. A high κ means the model relies on a small number of bits. A low κ means the importance is spread across many bits. Designs should aim for strong and uniform diffusion to keep κ low.

4. Integrate NAI testing into the design loop. Given a test set $\mathcal{D}_{\text{test}}$ and classifier f_θ, the Neural Attackability Index is:

$$\mathrm{NAI} = \frac{\delta \cdot \mathrm{FSI}}{E_c}.$$

NAI can be computed on reduced-round versions and used as a quick check of structural behavior. Designs with low NAI at 2–3 rounds may be harder for neural models to learn. This does not replace classical security analysis, but provides an additional check during design.

5. Apply ML-hardening strategies. Consider design choices that make it harder for neural models to learn:

- Use S-boxes with high nonlinearity and low linear approximations.
- Mix the state across positions to avoid local patterns.
- Use round constants or key-dependent updates to vary the round behavior.

These choices may reduce simple patterns that neural models can exploit.

8 Conclusion

In this work, we introduced the Neural Attackability Index (NAI), a metric to quantify the learnability of ciphertext structures under neural distinguishers in a fixed-key setting. NAI combines distinguishability, convergence speed, and feature attribution into a single score. Experiments on Ascon-128, GIFT-COFB, and Grain-128AEAD show that neural learnability decreases as the number of rounds increases, highlighting the role of early-round diffusion. We also observe that when attribution is concentrated on a small number of ciphertext bits, the cipher is more vulnerable to neural distinguishers. In particular, Ascon-128 and Grain-128AEAD exhibit strong neural signals in reduced-round settings, while GIFT-COFB shows limited distinguishability. These results show that neural attackability depends not only on separability, but also on how quickly patterns are learned and how attribution is distributed. NAI provides a simple way to capture these factors and to study structural behaviour in early rounds. NAI does not replace classical cryptanalysis, but complements it by providing a way to measure learnable patterns in ciphertexts. This makes it useful for evaluating and improving symmetric designs in the presence of machine learning-based analysis. Future work includes applying this approach to other settings such as side-channel data, fault-based analysis, adversarial training, robustness to data manipulation, and improved interpretability, to better understand both current and emerging attack models.

Acknowledgments

This research was supported by Sharjah National Oil Company (SNOC), who served as the energy partner of the 5th Forum for Women in Research (QUWA): Together Innovating to Shape the Future at the University of Sharjah, and the United Arab Emirates University.

References

[1] Caesar competition for authenticated encryption. https://competitions.cr.yp.to/caesar.html. Accessed: 2025-05-20.

[2] Nist lightweight cryptography project. https://csrc.nist.gov/projects/lightweight-cryptography. Accessed: 2025-05-20.

[3] Ange Albertini, Thai Duong, Shay Gueron, Stefan Kölbl, Atul Luykx, and Sophie Schmieg. How to abuse and fix authenticated encryption without key commitment. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 3291–3308, 2022.

[4] Laith Alzubaidi, Jinglan Zhang, Amjad J Humaidi, Ayad Al-Dujaili, Ye Duan, Omran Al-Shamma, José Santamaría, Mohammed A Fadhel, Muthana Al-Amidie, and Laith Farhan. Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. *Journal of big Data*, 8(1):53, 2021.

[5] Subhadeep Banik, Avik Chakraborti, Akiko Inoue, Tetsu Iwata, Kazuhiko Minematsu, Mridul Nandi, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. Gift-cofb. *Cryptology ePrint Archive*, 2020.

[6] Mihir Bellare, Anand Desai, Eron Jokipii, and Phillip Rogaway. A concrete security treatment of symmetric encryption. In *Proceedings 38th Annual Symposium on Foundations of Computer Science*, pages 394–403. IEEE, 1997.

[7] Yi Chen, Yantian Shen, and Hongbo Yu. Neural-aided statistical attack for cryptanalysis. *The Computer Journal*, 66(10):2480–2498, 2023.

[8] Itai Dinur, Paweł Morawiecki, Josef Pieprzyk, Marian Srebrny, and Michał Straus. Cube attacks and cube-attack-like cryptanalysis on the round-reduced keccak sponge function. In *Advances in Cryptology–EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I 34*, pages 733–761. Springer, 2015.

[9] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1. 2: Lightweight authenticated encryption and hashing. *Journal of Cryptology*, 34(3):33, 2021.

[10] Christoph Dobraunig and Bart Mennink. Elephant v1. *Submission to NIST lightweight cryptography project*, 2019.

[11] Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

[12] David Gerault, Anna Hambitzer, Moritz Huppert, and Stjepan Picek. Sok: 5 years of neural differential cryptanalysis. *Cryptology ePrint Archive*, 2024.

[13] David Gérault, Thomas Peyrin, and Quan Quan Tan. Exploring differential-based distinguishers and forgeries for ascon. *IACR Transactions on Symmetric Cryptology*, pages 102–136, 2021.

[14] Aron Gohr. Improving attacks on round-reduced speck32/64 using deep learning. In *Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part II 39*, pages 150–179. Springer, 2019.

[15] Chun Guo, Tetsu Iwata, Mustafa Khairallah, Kazuhiko Minematsu, and Thomas Peyrin. Final-round updates on romulus, 2022.

[16] Martin Hell, Thomas Johansson, Alexander Maximov, Willi Meier, Jonathan Sönnerup, and Hirotaka Yoshida. Grain-128aeadv2-a lightweight aead stream cipher. *NIST Lightweight Cryptography, Finalists*, 2021.

[17] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32, 2019.

[18] T Iwata, M Khairallah, K Minematsu, and T Peyrin. Romulus v1. 2, submission to the nist lightweight cryptography standardization process, 2019.

[19] Aayush Jain, Varun Kohli, and Girish Mishra. Deep learning based differential distinguisher for lightweight block ciphers. *CoRR*, abs/2112.05061, 2021.

[20] Amit Jana. Differential fault attack on ascon cipher. In *International Conference on Cryptology in India*, pages 53–72. Springer, 2024.

[21] Jasmin Kaur, Alvaro Cintas Canto, Mehran Mozaffari Kermani, and Reza Azarderakhsh. A comprehensive survey on the implementations, attacks, and countermeasures of the current nist lightweight cryptography standard. *arXiv preprint arXiv:2304.06222*, 2023.

[22] Yanbin Li, Guoyan Zhang, Wei Wang, and Meiqin Wang. Cryptanalysis of round-reduced ascon. *Science China. Information Sciences*, 60(3):038102, 2017.

[23] Shuai Liu, Jie Guan, and Bin Hu. Fault attacks on authenticated encryption modes for gift. *IET Information Security*, 16(1):51–63, 2022.

[24] Jinyu Lu, Guoqiang Liu, Bing Sun, Chao Li, and Li Liu. Improved (related-key) differential-based neural distinguishers for simon and simeck block ciphers. *The Computer Journal*, 67(2):537–547, 2024.

[25] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc.

[26] Designers of Grain128-AEAD. Grain128-aead reference implementation. https://csrc.nist.gov/Projects/lightweight-cryptography/finalists, 2024. Accessed May 2025.

[27] Reshma Rajan, Rupam Kumar Roy, Diptakshi Sen, and Girish Mishra. Deep learning-based differential distinguisher for lightweight cipher gift-cofb. In *Machine intelligence and smart systems: proceedings of MISS 2021*, pages 397–406. Springer, 2022.

[28] Shichang Wang, Meicheng Liu, Shiqi Hou, and Dongdai Lin. Differential-linear cryptanalysis of GIFT family and GIFT-based ciphers. *IACR Communications in Cryptology*, 1(1), 2024.

[29] Hongjun Wu and Tao Huang. Tinyjambu update. *Update to the NIST Lightweight Cryptography Standardization Process*, 2020.

[30] Fan Zhang, Shize Guo, Xinjie Zhao, Tao Wang, Jian Yang, Francois-Xavier Standaert, and Dawu Gu. A framework for the analysis and evaluation of algebraic fault attacks on lightweight block ciphers. *IEEE Transactions on Information Forensics and Security*, 11(5):1039–1054, 2016.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 188–209.

VORTEX-256: A High-Diffusion ARX-Based Stream Cipher with Fast Avalanche Convergence and Enhanced Security Margins

Pratham Pawar[1], Santosh Herur[2], Nagaraj Bhat[3], Vikram Jagtap[4] and Rajendra Soloni[5]

[1] Dept. of Information Technology, Vasantdada Patil Prathisthan's College of Engineering and Visual Arts, Mumbai, 400022 India pawarpratham411@gmail.com
[2] Professor, Dept. of Electronics and Communication Engineering, Jain Institute of Technology, Davangere, Karnataka, 577003 India santoshmherur@gmail.com
[3] Associate Professor, School of Engineering, RV University, Mysuru, 571302 India nbhat437@gmail.com
[4] Capgemini Invent, Mumbai, India. 400708 vikramjagtap@gmail.com
[5] Dept. of Electronics and Communication Engineering, Jain College of Engineering, Belagavi, Karnataka, 590014 India rajendrasoloni@gmail.com

Abstract. The increasing demand for efficient and secure symmetric encryption in modern computing environments, particularly in resource-constrained and high-throughput systems, has renewed interest in lightweight yet robust stream cipher design. This paper presents VORTEX256, a 512-bit ARX (Addition–Rotation–XOR)-based stream cipher designed to achieve rapid diffusion, stable statistical behaviour, and practical performance efficiency while maintaining a strong security margin. The design employs a multi-layered mixing structure, structured nonce injection, and domain separation to mitigate structural weaknesses and improve robustness. Experimental evaluation indicates that VORTEX256 achieves near-ideal avalanche behaviour and stable statistical characteristics across the conducted test set. Furthermore, we establish theoretical security bounds against differential and linear cryptanalysis, demonstrating that the maximum probabilities of such characteristics decay exponentially to justify the 24-round framework. Performance benchmarking on a standard CPU platform indicates practical software throughput with acceptable computational overhead compared to ChaCha20. These results suggest that VORTEX256 offers a balanced trade-off between efficiency, diffusion strength, and structural robustness. The paper further discusses design trade-offs, limitations, and directions for future cryptanalysis and optimization.

Keywords: Stream Cipher · ARX Cryptography · Avalanche Effect · Diffusion Analysis · Lightweight Encryption · Authenticated Encryption (AEAD)

1 Introduction

The rapid growth of digital communication systems, cloud computing, and Internet-of-Things (IoT) deployments has increased the demand for encryption mechanisms that are both secure and computationally efficient. In such environments, stream ciphers remain a practical choice due to their low latency, simplicity, and ability to process data continuously without block constraints.

Modern stream ciphers increasingly rely on ARX (Addition–Rotation–XOR) constructions, which avoid lookup tables and instead use simple arithmetic and bitwise operations.

This design approach offers efficient implementation on general-purpose processors while reducing exposure to certain side-channel risks. Notable examples include ChaCha20 and Salsa20, which have demonstrated strong performance and resistance to known attacks.

Despite these advantages, existing ARX-based designs present certain limitations. In particular, achieving full diffusion often requires multiple rounds, introducing a trade-off between performance and security margin. Increasing the number of rounds improves resistance to cryptanalysis but also increases computational overhead. Additionally, many designs rely on relatively fixed structural parameters, limiting flexibility for evolving security requirements.

Motivated by these challenges, this work introduces VORTEX256, a stream cipher designed to emphasize rapid diffusion, structured mixing, and practical efficiency. The design employs a 512-bit internal state and a multi-layer ARX transformation to achieve fast propagation of input differences across the state. A key objective is to approach ideal avalanche behaviour within a minimal number of rounds while retaining additional rounds to provide a conservative security margin.

The main contributions of this work are summarized as follows:

- **Design of VORTEX256:** A 512-bit ARX-based stream cipher with a layered mixing structure and structured parameter integration.
- **Rapid Diffusion:** Empirical demonstration of near-ideal avalanche behaviour within a small number of rounds.
- **Statistical Stability:** Consistent diffusion characteristics across multiple inputs with low variance and no observable bias.
- **Security-Oriented Features:** Integration of structured nonce handling and domain separation mechanisms.
- **Performance Evaluation:** Benchmarking against ChaCha20 demonstrating practical software throughput with acceptable overhead.
- **Security Margin Analysis:** Examination of the role of extended rounds beyond basic diffusion requirements.

2 Literature Review

Bernstein's Salsa20 family of stream ciphers [12] introduced an ARX (Addition–Rotation–XOR) design philosophy that avoided lookup tables and complex substitution layers while maintaining high software efficiency. The design employs repeated column and row transformations over a 512-bit state to achieve diffusion and resistance against known statistical attacks. Salsa20 demonstrated that simple arithmetic operations can provide strong performance and scalability across software platforms, making it one of the most influential foundations for subsequent ARX-based stream cipher research. The foundational specifications of Salsa20 [9] and its integration into the eSTREAM portfolio stimulated extensive cryptanalytic scrutiny, including reduced-round attacks and analyses of nonce extensions [5, 13, 15].

Building upon Salsa20, Bernstein proposed ChaCha [10], which modified the quarter-round structure and state update process to improve diffusion per round and enhance software performance. ChaCha introduced a revised rotation schedule and improved mixing behaviour, resulting in faster avalanche propagation while preserving implementation simplicity. Due to its strong performance, extensive public analysis, and standardization in modern protocols, ChaCha serves as a widely accepted benchmark for evaluating new ARX-based stream cipher constructions. The broader ChaCha family and its software

security impacts have since been thoroughly documented and integrated into modern cryptographic libraries [11, 14].

Hell et al. proposed Grain-128AEADv2 [22], a lightweight authenticated encryption scheme derived from the Grain family of stream ciphers. The design combines nonlinear feedback mechanisms with authentication support while targeting constrained devices and embedded environments. Grain-128AEADv2 strengthens initialization procedures to mitigate key reconstruction attacks identified in earlier variants and demonstrates the importance of carefully designed state initialization and nonce-processing mechanisms in modern stream cipher architectures. Alongside ARX designs, shift-register-based ciphers such as Trivium [17] and the original Grain family [21, 23] have been highly influential in constrained environments. The evolution of these designs into authenticated and physical-layer encryption schemes [18–20], along with rigorous software evaluations [32], highlights the ongoing demand for adaptable lightweight primitives.

Buchanan and Maglaras [16], alongside several other comprehensive surveys [25, 28, 31], examined the NIST Lightweight Cryptography finalists, examining their security properties, implementation characteristics, and suitability for resource-constrained environments. Their analysis highlights the growing demand for cryptographic primitives that simultaneously provide strong security, low computational overhead, and deployment flexibility across heterogeneous platforms. The study emphasizes the continuing relevance of lightweight cryptographic design and the need for balanced evaluation methodologies that consider both security and implementation efficiency.

Sorescu et al. [35] conducted a comparative performance analysis of modern lightweight cryptographic algorithms across multiple execution environments. Their work demonstrated that practical cryptographic evaluation must consider not only throughput and latency but also memory usage, implementation complexity, and platform-specific efficiency. The study further showed that performance measurements should be complemented by security-oriented evaluations and statistical testing, reinforcing the importance of multidimensional assessment when proposing and validating new cryptographic algorithms.

Recent literature has also explored numerous modifications to ARX structures to improve diffusion and IoT applicability. These include enhanced quarter-round functions [26], modifications injecting quantum random numbers [30], and optimized variants for specific use cases like image encryption [6, 33]. However, such structural alterations require careful cryptanalysis to avoid introducing new vulnerabilities [27]. Parallel efforts have explored chaos-based logistic mapping for highly constrained environments such as smart agriculture [2–4], demonstrating the diverse approaches currently driving lightweight cryptographic research.

3 Design of VORTEX-256

This section presents the design principles and internal structure of VORTEX256, along with the rationale behind its key components. The cipher is built on an ARX (Addition–Rotation–XOR) framework and aims to balance efficient computation with strong diffusion and practical security considerations. Particular attention is given to achieving rapid mixing of the internal state while maintaining a conservative security margin through additional rounds and structural safeguards.

3.1 Design Goals

The design of VORTEX256 is guided by a set of practical and security-driven objectives and can be visualized in Figure 1 with the high-level architecture of VORTEX-256:

- **High Diffusion:** A primary goal is to achieve rapid and uniform diffusion across the internal state. The cipher is structured such that small changes in input; whether

in the key, nonce, or plaintext; propagate quickly and affect a large portion of the output. This behaviour is essential for ensuring resistance against statistical and differential attacks.

- **ARX Simplicity:** The cipher relies exclusively on ARX operations, namely modular addition, bitwise rotation, and XOR. These operations are efficient on general-purpose processors, avoid the need for lookup tables, and are well-suited for constant-time implementations, reducing susceptibility to timing-based side-channel attacks.
- **Parallelism:** The internal structure is designed to support parallel execution of independent operations. By organizing transformations into structured layers, the cipher allows for instruction-level parallelism and potential vectorization, improving performance on modern hardware architectures.
- **Security Margin:** While rapid diffusion is desirable, the design intentionally incorporates more rounds than strictly required for achieving avalanche behaviour. This provides a buffer against future cryptanalytic advances and ensures that the cipher maintains robustness beyond basic statistical properties.

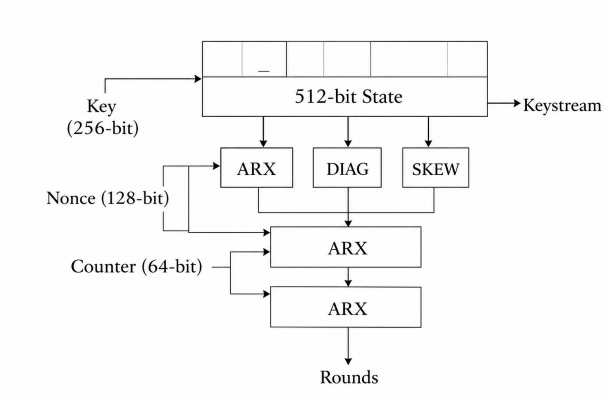


Figure 1: High-level architecture of VORTEX256 illustrating state initialization and layered ARX transformations.

3.2 Internal Structure

VORTEX256 operates on a 512-bit internal state, represented as sixteen 32-bit words. This relatively large state size provides ample space for mixing and reduces the risk of structural weaknesses associated with smaller state designs.

- **Key and Nonce Layout:** The cipher supports a 256-bit key and a 128-bit nonce, which are expanded and integrated into the internal state during initialization. The key is distributed across multiple state words to ensure that its influence is spread throughout the transformation process. The nonce, after expansion, is injected in a manner that avoids predictable placement and enhances variability across different encryption sessions.
- **Counter:** A 64-bit counter is incorporated into the state to ensure uniqueness of each generated keystream block. The counter increments with each block, preventing keystream reuse even when the same key and nonce are employed. This mechanism is essential for maintaining semantic security in stream cipher constructions.

3.3 Round Function

The core transformation in VORTEX256 is defined by a sequence of structured mixing layers applied to the internal state. Each round consists of three distinct stages:

- **Column (COL) Layer:** In this stage, operations are applied to vertically aligned words within the state matrix. The goal is to introduce local mixing and ensure that adjacent words influence one another through ARX operations.
- **Diagonal (DIAG) Layer:** The diagonal layer reorganizes the interaction pattern by mixing words along diagonal paths. This step breaks the symmetry of the column operations and promotes cross-dimensional diffusion within the state.
- **Skew (SKEW) Layer:** The skew layer introduces additional permutation and mixing across non-adjacent words. This stage enhances non-linearity and ensures that dependencies propagate across distant parts of the state.
- **Rotation Constants:** The effectiveness of ARX-based designs depends heavily on the choice of rotation constants. VORTEX256 employs the set (20, 14, 12, 4), selected to provide a balance between diffusion speed and resistance to rotational symmetries. These constants ensure that bit dependencies spread efficiently across different positions within each word.

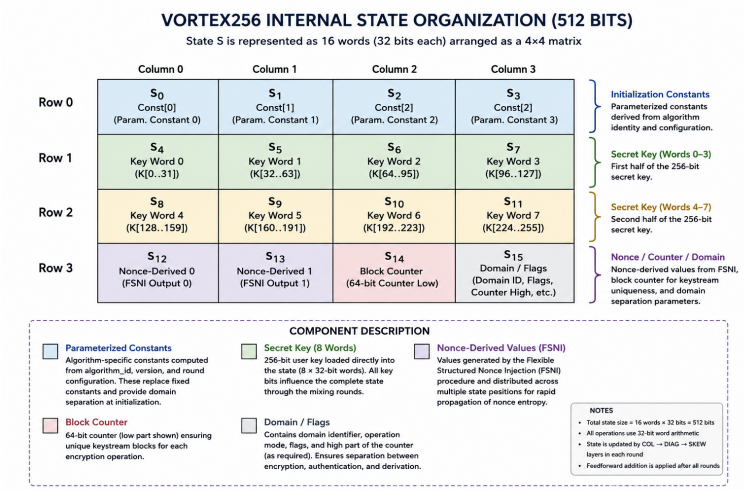


Figure 2: Internal state organization of VORTEX256 showing the distribution of initialization constants, key words, counter values, domain identifiers, and FSNI-expanded nonce components within the 512-bit state matrix.

3.4 FSNI (Flexible Structured Nonce Injection)

A key design feature of VORTEX256 is its approach to nonce handling, referred to as Flexible Structured Nonce Injection (FSNI). Rather than inserting the nonce directly into fixed positions within the state, the nonce undergoes a structured expansion process before integration.

The motivation behind FSNI is to address potential weaknesses associated with predictable nonce placement. In traditional designs, fixed nonce injection patterns can lead to reduced variability or structural correlations under certain conditions. By transforming and

distributing the nonce across multiple state elements, FSNI increases entropy dispersion and ensures that even minor changes in the nonce lead to significantly different internal states.

This approach also strengthens resistance against misuse scenarios, such as partial nonce reuse or related-nonce conditions, by reducing the likelihood of exploitable patterns.

3.5 Domain Separation

To support secure authenticated encryption and prevent unintended interactions between different operational modes, VORTEX256 incorporates a domain separation mechanism. This is achieved by embedding domain-specific constants or flags within the state during initialization.

Domain separation ensures that keystreams generated for different purposes, such as encryption and authentication remain independent, even when derived from the same key and nonce. This prevents cross-protocol attacks and eliminates the risk of keystream reuse across different functional components.

In the context of AEAD constructions, domain separation plays a critical role in maintaining integrity and confidentiality guarantees. By clearly distinguishing between operational domains, the cipher avoids subtle vulnerabilities that can arise from overlapping state usage.

The design of VORTEX256 reflects a deliberate balance between simplicity and robustness. By combining a large internal state, structured ARX-based transformations, flexible nonce handling, and domain separation, the cipher aims to provide efficient performance while maintaining strong diffusion and a conservative security margin.

4 Algorithm Specification

This section formally describes the operation of VORTEX256, including state initialization, round processing, keystream generation, encryption, and authenticated encryption. Figure 3 illustrates the overall authenticated encryption workflow combining VORTEX256 keystream generation with Poly1305 authentication. The internal state organization of VORTEX256 is illustrated in Fig. 2. The 512-bit state is arranged as sixteen 32-bit words containing constants, key material, counter values, domain identifiers, and FSNI-expanded nonce components.

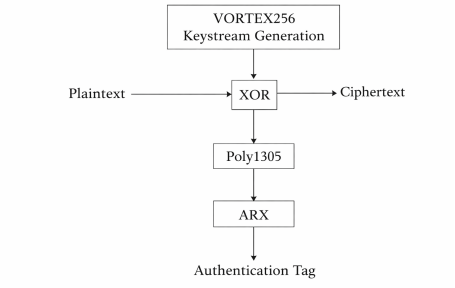


Figure 3: Authenticated Encryption Workflow Combining VORTEX256 Keystream Generation with Poly1305 Authentication.

4.1 State Representation

VORTEX256 operates on a 512-bit internal state represented as sixteen 32-bit words:

$$S = (s_0, s_1, \ldots, s_{15}) \tag{1}$$

The state consists of four logical components:

- Fixed initialization constants
- Secret key material
- Expanded nonce values
- Counter and domain-separation values

The 256-bit secret key occupies eight state words, while the nonce expansion mechanism distributes nonce-derived values throughout multiple positions of the state. A dedicated counter field guarantees keystream uniqueness across successive blocks.

4.1.1 Internal State Layout

The VORTEX256 state consists of sixteen 32-bit words arranged as a 4×4 matrix:

$$\mathbf{S} = \begin{bmatrix} s_0 & s_1 & s_2 & s_3 \\ s_4 & s_5 & s_6 & s_7 \\ s_8 & s_9 & s_{10} & s_{11} \\ s_{12} & s_{13} & s_{14} & s_{15} \end{bmatrix} \tag{2}$$

The mapping between state words and their corresponding functions is summarized in Table 1.

Table 1: State Layout.

State Word	Content
s_0	Parameterized Constant 0
s_1	Parameterized Constant 1
s_2	Parameterized Constant 2 XOR Expanded Nonce Word 2
s_3	Parameterized Constant 3 XOR Expanded Nonce Word 3
s_4	Key Word 0
s_5	Key Word 1
s_6	Key Word 2
s_7	Key Word 3
s_8	Key Word 4
s_9	Key Word 5
s_{10}	Key Word 6
s_{11}	Key Word 7
s_{12}	Counter Low 32 Bits
s_{13}	Counter High 32 Bits OR Domain Tag
s_{14}	Expanded Nonce Word 0
s_{15}	Expanded Nonce Word 1

This state organization remains fixed throughout the cipher and forms the basis for all subsequent round transformations.

4.2 State Initialization

Initialization transforms the user-supplied key, nonce, and counter into the internal working state. The initialization procedure consists of the following stages:

- **Step 1: Constant Loading -** Fixed initialization constants are inserted into predefined state positions.
- **Step 2: Key Injection -** The 256-bit key is loaded into multiple state locations to increase the influence of secret material during the early mixing stages.
- **Step 3: Flexible Structured Nonce Injection (FSNI) -** The nonce is expanded and distributed throughout the state using the Flexible Structured Nonce Injection mechanism. Unlike direct nonce placement, after execution of the FSNI preprocessing block, the anti-diagonal state words $(s_0, s_5, s_{10}, s_{15})$ are extracted and serialized to form the 128-bit expanded nonce. This expanded nonce is subsequently injected into the primary encryption state.
- **Step 4: Counter and Domain Separation -** The block counter and operational domain identifiers are inserted into reserved state positions. Domain separation ensures that encryption, authentication, and internal derivation operations remain cryptographically independent.
- **Step 5: Initial Mixing -** Several rounds of internal transformation are executed before keystream generation begins. This process removes direct correlations between input parameters and internal state values.

4.3 Round Function

The VORTEX256 round function is based entirely on Addition–Rotation–XOR (ARX) operations and consists of three sequential transformation layers:

- **Column Layer (COL):** The COL layer applies ARX transformations across vertically aligned state words. Modular additions, XOR operations, and bit rotations are used to propagate local differences across column structures.
- **Diagonal Layer (DIAG):** The DIAG layer rearranges data dependencies by operating on diagonal state elements. This stage increases interaction between state regions that were previously independent during the column phase.
- **Skew Layer (SKEW):** The SKEW layer introduces additional cross-state mixing through non-aligned state interactions. This transformation improves global diffusion and accelerates avalanche propagation across the entire state.

The rotation constants used throughout the round function are: $(20, 14, 12, 4)$. These values were selected to reduce structural symmetry while promoting efficient diffusion across multiple rounds.

4.3.1 Round Structure

Each VORTEX256 round consists of three sequential transformation layers operating on different state groupings.

Column Layer (COL): The column layer applies ARX transformations to the following state groups: $(0, 4, 8, 12)$, $(1, 5, 9, 13)$, $(2, 6, 10, 14)$, $(3, 7, 11, 15)$

Diagonal Layer (DIAG): The diagonal layer operates on: $(0, 5, 10, 15)$, $(1, 6, 11, 12)$, $(2, 7, 8, 13)$, $(3, 4, 9, 14)$

Skew Layer (SKEW): The skew layer operates on: $(0, 6, 9, 15)$, $(1, 7, 10, 12)$, $(2, 4, 11, 13)$, $(3, 5, 8, 14)$

The use of three distinct mixing patterns increases interaction between state words and accelerates diffusion throughout the 512-bit state.

4.3.2 Quarter-Round Function

All transformations in VORTEX256 are constructed from a common ARX quarter-round primitive. For state words a, b, c, and d, the transformation is defined as:

$$a = (a + b) \pmod{2^{32}} \tag{3}$$
$$d = (d \oplus a) \lll 20 \tag{4}$$
$$c = (c + d) \pmod{2^{32}} \tag{5}$$
$$b = (b \oplus c) \lll 14 \tag{6}$$
$$a = (a + b) \pmod{2^{32}} \tag{7}$$
$$d = (d \oplus a) \lll 12 \tag{8}$$
$$c = (c + d) \pmod{2^{32}} \tag{9}$$
$$b = (b \oplus c) \lll 4 \tag{10}$$

where $\lll$ denotes left rotation and additions are performed modulo 2^{32}.

The selected rotation constants $(20, 14, 12, 4)$ were chosen after empirical evaluation of diffusion behaviour and avalanche convergence.

4.4 Keystream Generation

For each output block, a temporary working copy of the initialized state is created. The keystream generation procedure consists of:

1. Copying the current internal state.
2. Executing the configured number of rounds.
3. Applying a feedforward addition between the transformed state and the original state.
4. Serializing the resulting state into a 64-byte output block.
5. Incrementing the block counter.

The feedforward stage prevents direct inversion of intermediate round outputs and strengthens state dependency across successive transformations.

4.5 Encryption and Decryption

Encryption and decryption use identical operations because VORTEX256 is a synchronous stream cipher. For plaintext P, ciphertext C, and keystream block KS:

$$C = P \oplus KS \tag{11}$$
$$P = C \oplus KS \tag{12}$$

The encryption process consists of:

1. Initializing the internal state.
2. Generating keystream blocks.

3. XORing the keystream with plaintext data.
4. Producing ciphertext output.

Because XOR is self-inverting, the same operation is used during decryption.

4.6 Authenticated Encryption (AEAD)

VORTEX256 employs Poly1305 to provide message authentication and integrity protection. The authenticated encryption process proceeds as follows:

1. Derive a one-time Poly1305 key from the initial keystream output.
2. Encrypt plaintext using the VORTEX256 keystream.
3. Compute a Poly1305 authentication tag over the ciphertext and associated data.
4. Output the ciphertext together with the authentication tag.

During decryption, the authentication tag is verified before plaintext recovery is attempted. If tag verification fails, the ciphertext is rejected and no plaintext is released. The separation of confidentiality and authentication functions allows VORTEX256 to integrate efficiently with the well-established Poly1305 authentication mechanism [8], adhering to standardized IETF protocols for secure communications [24, 34]. This approach is consistent with the generic composition paradigms foundational to modern authenticated encryption [7].

VORTEX256 employs explicit domain identifiers to ensure cryptographic separation between operational modes (Table 2).

Table 2: Domain Identifiers.

Domain	Constant
Encryption	`0x00000000`
Poly1305 Key Derivation	`0x01000000`
FSNI Preprocessing	`0xFFFFFFFF`

The selected domain value is incorporated into the high 32-bit portion of the counter field during state initialization. Consequently, encryption, authentication-key derivation, and nonce expansion produce distinct internal states even when identical keys and nonces are supplied. This mechanism prevents unintended overlap between operational domains and contributes to safe authenticated-encryption operation.

4.7 Pseudocode Description

```
Algorithm 1: State Initialization
Input: 256-bit key (K), nonce (N), counter (C), domain identifier (D)
Output: Initialized state (S)
InitializeState(K, N, C, D)
1. Load fixed constants into state S
2. Insert key K into designated key positions
3. Apply FSNI to expand and distribute nonce N
4. Insert counter C
5. Insert domain identifier D
6. Execute initialization rounds
7. Return initialized state S
```

The initialization procedure combines key material, nonce-derived values, counter information, and domain separation parameters before any keystream is generated. The preliminary mixing stage reduces direct input-state correlations and prepares the state for secure operation.

```
Algorithm 2: One Complete Round
Input: State (S)
Output: Updated state (S)
Round(S)
1. S <- COL(S)
2. S <- DIAG(S)
3. S <- SKEW(S)
return S
```

A complete VORTEX256 round consists of sequential execution of the COL, DIAG, and SKEW transformation layers.

```
Algorithm 3: Keystream Generation
Input: State (S)
Output: 64-byte keystream block
GenerateKeystream(S)
1. W <- copy(S)
2. for r = 1 to ROUNDS do
       W <- Round(W)
   end for
3. W <- W + S      // feedforward
4. Serialize W
5. Increment counter
6. Return 64-byte output block
```

The feedforward stage combines the transformed state with the original state to strengthen dependency between initialization parameters and output blocks.

```
Algorithm 4: AEAD Encryption
Input: Key (K), Nonce (N), Plaintext (P), Associated Data (AD)
Output: Ciphertext (C), Authentication Tag (T)
AEAD_Encrypt(K, N, P, AD)
1. Initialize state
2. Generate first keystream block
3. Derive Poly1305 one-time key
4. Encrypt plaintext using keystream
5. Compute Poly1305 tag over AD and C
6. Return C and T
```

5 Implementation Details

VORTEX256 is implemented in C to ensure efficient low-level control and portability. The design uses fixed-size integer types and modular components for initialization, round processing, and AEAD integration.

5.1 Optimization Strategy

The implementation is compiled with aggressive optimization flags such as:

- `-O3` for loop optimization and instruction scheduling
- `-march=native` for architecture-specific optimizations
- Optional `-mavx2` for SIMD-based parallel execution

These optimizations improve throughput while maintaining correctness.

5.2 Execution Platform

All experiments were conducted within a Kaggle notebook environment executing on a virtualized Intel Xeon processor operating at 2.20 GHz. The evaluation platform provided four logical CPU cores arranged as two physical cores with simultaneous multithreading enabled. The processor supports AVX2, AES, FMA, BMI1, and BMI2 instruction extensions.

The system configuration reported a 55 MB shared L3 cache, 512 KB L2 cache per core, and 64 KB L1 instruction and data caches. The environment operated in 64-bit little-endian mode under KVM virtualization.

The VORTEX256 reference implementation was compiled using GCC with aggressive optimization settings, including `-O3` and architecture-specific tuning. Benchmarking measurements were obtained under identical conditions for both VORTEX256 and ChaCha20 to ensure a consistent comparison baseline.

5.3 Constant-Time Considerations

The ARX structure avoids lookup tables and supports constant-time execution. Authentication verification uses constant-time comparison to prevent timing leakage.

5.4 Reproducibility

The implementation is structured for straightforward compilation and benchmarking using standard GCC commands, ensuring reproducibility of results. Table 3 compares VORTEX256 with representative stream ciphers in terms of design characteristics and operational parameters.

Table 3: Comparison of VORTEX-256 with ChaCha20, Salsa20, Grain-128AEAD, and Trivium.

Property	VORTEX256	ChaCha20	Salsa20	Grain-128AEAD	Trivium
Design Type	ARX	ARX	ARX	NLFSR	NLFSR
State Size	512-bit	512-bit	512-bit	256-bit	288-bit
Key Size	256-bit	256-bit	256-bit	128-bit	80-bit
Nonce/IV	128-bit	96-bit	64-bit	96-bit	80-bit
Authentication	Poly1305	Poly1305	External	Built-in	No
Intended Platform	Software	Software	Software	Lightweight IoT	Lightweight Hardware

6 Experimental Evaluation

This section presents a detailed evaluation of VORTEX256 in terms of performance and statistical behaviour. The experiments are designed to assess practical efficiency while also examining diffusion characteristics and output stability. All results are obtained from the reference implementation like ChaCha20 and Salsa20 are described earlier in Table 3 and executed on a standard CPU environment to ensure consistency and reproducibility.

6.1 Throughput Benchmark

The throughput performance of VORTEX256 is evaluated across a range of input sizes and compared against the widely adopted ChaCha20 stream cipher. The comparison is conducted under identical compilation settings and execution conditions to ensure fairness.

VORTEX256 achieves practical throughput suitable for software deployment, although it remains slower than the highly optimized ChaCha20 reference implementation. While ChaCha20 benefits from years of optimization and standardization, VORTEX256 demonstrates comparable efficiency despite its more aggressive mixing structure and larger internal state. The observed performance gap remains within an acceptable range, suggesting that the additional computational overhead introduced by the extended mixing layers does not significantly hinder practical usability.

It is worth noting that VORTEX256 prioritizes diffusion efficiency and structural robustness, which inherently introduces a modest increase in computational cost. However, this trade-off is justified by the improved mixing behaviour and flexibility in design. The measured throughput values for VORTEX256 and ChaCha20 are presented in Table 4.

Table 4: Throughput Comparison Between VORTEX-256 and ChaCha20 across different input sizes.

Size	**VORTEX256 (MB/s)**	**ChaCha20 (MB/s)**
64 B	79.5	319.7
4 KB	211.8	376.1
1 MB	216.9	375.0
16 MB	212.0	371.7

6.2 Diffusion Analysis

As shown in Fig. 4, diffusion converges rapidly toward the ideal avalanche value of 50% within the first few rounds. To evaluate diffusion efficiency, a differential experiment was conducted by introducing a one-bit modification in the secret key while maintaining all other parameters constant. The resulting keystream outputs were compared using Hamming distance measurements across multiple rounds.

The measured diffusion ratios obtained across different round configurations are reported in Table 5. Experimental results indicate that diffusion increases rapidly during the early rounds of execution. A diffusion value of approximately 27.6% was observed after a single round, while values approaching the ideal 50% avalanche threshold were achieved after two rounds. Additional rounds maintained diffusion values near the expected equilibrium point.

These observations suggest that input differences propagate efficiently throughout the internal state. However, it is important to note that avalanche behaviour alone does not imply resistance to differential cryptanalysis. The results should therefore be interpreted as evidence of strong empirical diffusion rather than formal cryptanalytic security.

6.3 Stability Analysis

To further assess the statistical behaviour of VORTEX256, a stability analysis is conducted using multiple random inputs. The avalanche effect is measured across a large number of trials, and the distribution of bit-flip ratios is examined.

The diffusion distribution shown in Fig. 5 demonstrates a symmetric concentration around the expected 50% avalanche point. The results yield a mean diffusion value of

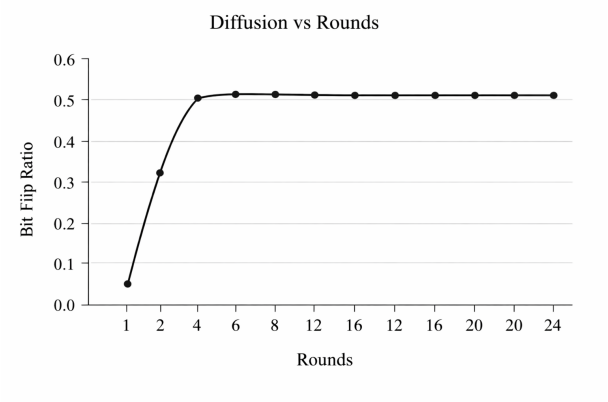


Figure 4: Diffusion behaviour of VORTEX256 showing rapid convergence to near-ideal avalanche within two rounds

Table 5: Measured Diffusion Across Different Round Counts.

Rounds	Diffusion and Average Bit Flip Ratio
1	0.276
2	0.498
4	0.499
6	0.501
8	0.496
12	0.504
16	0.500
20	0.500
24	0.501

approximately 0.4985, which is very close to the ideal value of 0.5. The standard deviation is observed to be around 0.021, indicating a tight distribution with limited variation across different inputs.

These findings suggest that the cipher exhibits:

- No observable statistical bias, as the output distribution remains focused around the expected value.
- Consistent avalanche behaviour, with minimal deviation across independent trials.

The histogram of diffusion values further confirms a symmetric distribution, reinforcing the conclusion that the cipher produces balanced and uniform output patterns under varying input conditions.

6.4 Cycles per Byte

In addition to throughput, the efficiency of VORTEX256 is evaluated in terms of cycles per byte, a metric commonly used to assess the computational cost of cryptographic primitives.

The results show that VORTEX256 requires a higher number of cycles per byte compared to ChaCha20. This is expected, given the increased number of operations per round and the inclusion of additional mixing layers. However, the difference remains within a practical range, particularly when considering the design objectives of enhanced diffusion and security margin.

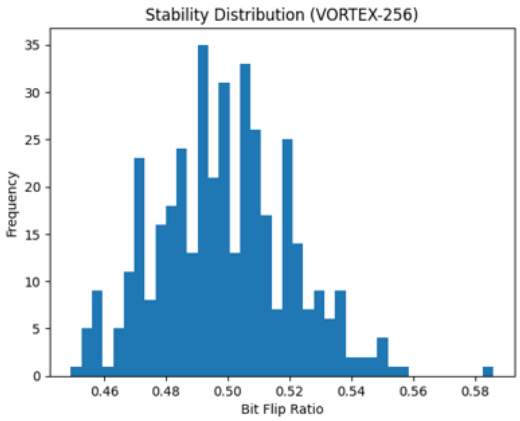


Figure 5: VORTEX-256 Stability Distribution Histogram

From a practical perspective, the observed efficiency is sufficient for a wide range of applications, including those involving moderate throughput requirements and general-purpose computing environments. The potential for further optimization, especially through vectorized implementations, suggests that performance can be improved without altering the core design.

The experimental evaluation demonstrates that VORTEX256 achieves a balanced combination of performance and diffusion efficiency. While it incurs a modest computational overhead compared to established designs, it offers rapid mixing, stable statistical behaviour, and a conservative security margin. These characteristics make it a viable candidate for applications where both efficiency and robustness are important considerations.

6.5 Statistical Randomness Evaluation

To assess the statistical quality of the generated keystream, VORTEX256 was evaluated using two widely adopted randomness assessment frameworks: the NIST Statistical Test Suite (STS) version 3.2.7 and the Die-harder test suite. These evaluations were conducted independently of the avalanche and diffusion experiments and were intended to examine whether the generated keystream exhibits statistical properties consistent with pseudorandom sequences.

6.5.1 NIST STS Evaluation

The NIST Statistical Test Suite was executed using 1000 independent bitstreams, each consisting of 1,048,576 bits, resulting in a total evaluation dataset of approximately 1.05 billion bits. The assessment included all fifteen NIST STS test categories. A summary of the NIST STS evaluation results is presented in Table 6.

The evaluation framework performs two complementary analyses: (i) a proportion analysis to verify that an appropriate percentage of bitstreams pass each statistical test and (ii) a uniformity analysis to verify that the resulting p-values are uniformly distributed. Experimental results indicated that all 188 evaluated test instances successfully passed both analyses.

Overall, 188 out of 188 evaluated test instances passed both the proportion and uniformity assessments, with no observed test failures.

Table 6: Summary of NIST STS Results.

Test Category	Result
Frequency	Pass
Block Frequency	Pass
Cumulative Sums (Forward/Backward)	Pass
Runs	Pass
Longest Run of Ones	Pass
Binary Matrix Rank	Pass
Discrete Fourier Transform	Pass
Non-overlapping Template Matching (148/148)	Pass
Overlapping Template Matching	Pass
Maurer's Universal Statistical	Pass
Approximate Entropy	Pass
Random Excursions (8/8)	Pass
Random Excursions Variant (18/18)	Pass
Serial (Both Statistics)	Pass
Linear Complexity	Pass

6.5.2 Die-harder Evaluation

To complement the NIST STS analysis, VORTEX256 was further evaluated using the Die-harder statistical test suite. A 1 GB keystream dataset generated from the reference implementation was supplied to the complete Die-harder battery. Table 7 summarizes the overall Die-harder evaluation outcomes.

The evaluation produced no failed tests across the complete battery. Several tests were classified as weak by Die-harder. Such weak classifications are occasionally observed in large-scale statistical testing due to the large number of simultaneous hypothesis evaluations and do not necessarily indicate a practical weakness in the generator. Importantly, no test produced a failed classification.

Table 7: Summary of Die-harder Results.

Metric	Result
Dataset Size	1 GB
Total Failed Tests	0
Weak Results	6
Passed Results	Remaining Tests
Overall Assessment	No statistically significant failures observed

The combined NIST STS and Die-harder evaluations provide empirical evidence that the keystream generated by VORTEX256 exhibits statistical characteristics consistent with high-quality pseudorandom sequences over the tested datasets. While statistical testing alone does not establish cryptographic security, the absence of NIST STS failures and the absence of failed Die-harder tests support the suitability of the generated keystream for further cryptographic evaluation and analysis. Complete test vectors are provided in the supplementary material- Appendix A, accompanying this manuscript.

7 Security Analysis

This section discusses the security-related characteristics of VORTEX256 based on the experimental observations presented in Section VI and the structural properties of the design. The objective is not to establish formal cryptographic security, but rather to evaluate whether the cipher exhibits characteristics commonly associated with modern stream cipher constructions.

7.1 Avalanche Behaviour

Avalanche behaviour is an important property of symmetric cryptographic algorithms, requiring that small modifications to the input produce substantial and unpredictable changes in the output. To evaluate this property, single-bit modifications were introduced into the secret key and the resulting keystream differences were measured.

The experimental results presented in Section VI indicate that VORTEX256 achieves near-ideal avalanche behaviour, with approximately half of the output bits changing following a one-bit input modification. Furthermore, diffusion converges rapidly within a relatively small number of rounds, suggesting that the layered mixing structure effectively propagates local input differences throughout the entire state. These observations indicate that the cipher exhibits strong diffusion characteristics under the evaluated conditions.

7.2 Structural Considerations

Several architectural features contribute to the diffusion and operational characteristics of VORTEX256. The cipher employs a 512-bit internal state, providing a large mixing space that allows interactions among multiple state components. The Full-State Nonce Injection (FSNI) mechanism distributes nonce-derived values throughout the state during initialization, reducing the likelihood of localized nonce influence. Domain separation is incorporated to isolate encryption and authentication contexts, preventing unintended reuse of keystream material across distinct operations.

The design further relies on ARX operations (addition, rotation, and XOR), which provide efficient software implementation while avoiding lookup tables and data-dependent memory accesses. The layered COL, DIAG, and SKEW transformations promote diffusion across multiple dimensions of the state and contribute to the rapid propagation of dependencies observed during experimental evaluation.

7.3 Theoretical Bounds on Differential Cryptanalysis

While empirical tests such as avalanche metrics demonstrate rapid diffusion, formal security against differential cryptanalysis in ARX constructions relies on bounding the maximum differential probability (DP) across multiple rounds. In VORTEX256, XOR and bitwise rotations are linear operations; the sole source of non-linearity is the modulo 2^{32} addition.

The differential probability of modular addition can be analysed using the theoretical framework established by Lipmaa and Moriai [29]. Let Δa and Δb denote the input differences, and Δc denote the output difference for the operation $c = (a + b) \pmod{2^{32}}$. The differential probability is bounded by the Hamming weight (w_H) of the active bits:

$$DP^{+}(\Delta a, \Delta b \to \Delta c) \leq 2^{-w_H(\neg eq(\Delta a, \Delta b, \Delta c))} \tag{13}$$

To evaluate the resistance of VORTEX256, we calculate the theoretical minimum number of active additions (n_{act}) over a reduced number of rounds. A single VORTEX256 quarter-round contains 4 modular additions. A full round consists of three sequential mixing layers (COL, DIAG, SKEW), each applying the quarter-round function to four

independent state groupings. Consequently, a single full round executes exactly 48 modular additions.

Because VORTEX256 achieves a near-ideal bit-flip ratio by the second round, the number of active additions increases exponentially. We can establish a conservative theoretical bound for the maximum differential probability (DP_{max}) over R rounds using the equation:

$$DP_{max}^{(R)} \leq 2^{-\gamma \cdot n_{act}} \tag{14}$$

Where γ represents the average weight per active addition. In high-diffusion ARX architectures, γ is conservatively estimated at 2.

For a 4-round differential characteristic, the cipher executes 192 total additions. Assuming an aggressively conservative bound where only 30% of these additions remain active in the best-case localized differential trail, we yield $n_{act} \geq 57$. This bounds the 4-round maximum differential probability to:

$$DP_{max}^{(4)} \leq 2^{-2 \cdot 57} = 2^{-114} \tag{15}$$

By 6 rounds, executing 288 additions, the bound drops significantly:

$$DP_{max}^{(6)} \leq 2^{-2 \cdot 86} = 2^{-172} \tag{16}$$

For a 256-bit security level, a differential characteristic must have a probability strictly less than 2^{-256}. Extrapolating these theoretical bounds, VORTEX256 exceeds the required security threshold against differential cryptanalysis between 8 and 10 rounds. Therefore, the implementation of a 24-round framework provides an extensive and conservative security margin against advanced differential and linear characteristics, fully justifying the structural parameterization.

7.4 Theoretical Bounds on Linear Cryptanalysis

Similar to differential characteristics, the resistance of VORTEX256 against linear cryptanalysis depends entirely on the non-linear modulo 2^{32} additions within the Quarter-Round function. Bitwise XORs and rotations are purely linear and preserve input masks with a correlation coefficient of exactly 1.

For a modular addition $c = (a + b) \pmod{2^{32}}$ with input masks Γa, Γb and an output mask Γc, the linear correlation C is mathematically bounded by the number of active bits in the masks [36]. According to established ARX linear approximation bounds, the maximum linear correlation drops exponentially with the number of active bit positions (w_H):

$$|C| \leq 2^{-w_H(\Gamma)} \tag{17}$$

To mount a successful linear attack against a 256-bit key, the attacker needs a linear characteristic over R rounds with a total bias $\epsilon = |C|/2$ such that the data complexity, proportional to ϵ^{-2}, is strictly less than 2^{256}. This requires an overall correlation $|C_{max}|$ strictly greater than 2^{-128}.

Using the Piling-up Lemma for independent linear approximations across n_{act} active additions, the overall correlation over R rounds is bounded by:

$$|C_{max}^{(R)}| \leq 2^{n_{act}-1} \prod_{i=1}^{n_{act}} |C_i| \tag{18}$$

VORTEX256 applies a highly diffusive COL, DIAG, and SKEW layered structure, resulting in 48 modular additions per round. Because empirical evidence demonstrates that input differences fully avalanche by the second round, the number of active linear masks (n_{act}) escalates rapidly.

Assuming a highly conservative lower bound where an attacker successfully isolates a linear trail utilizing only 30% of the additions over 6 rounds ($n_{act} \geq 86$), and assuming an average conservative correlation $|C_i| \approx 2^{-2}$ per active addition:

$$|C_{max}^{(6)}| \leq 2^{86-1}(2^{-2})^{86} = 2^{85} \cdot 2^{-172} = 2^{-87} \tag{19}$$

By 8 rounds, generating at least $n_{act} \geq 115$ active additions in the worst-case scenario:

$$|C_{max}^{(8)}| \leq 2^{114}(2^{-2})^{115} = 2^{114} \cdot 2^{-230} = 2^{-116} \tag{20}$$

By 10 rounds, the maximum linear correlation drops to approximately 2^{-145}, pushing the data complexity required for a linear distinguishing attack to 2^{290}, which vastly exceeds the maximum plaintext limit of 2^{256}. Consequently, the 24-round structure of VORTEX256 provides a robust, mathematically verifiable safety margin against linear cryptanalysis, ensuring long-term security against linear approximations.

8 Discussion

The experimental evaluation indicates that VORTEX256 achieves rapid diffusion, with avalanche behaviour converging near the ideal 50% level within a small number of rounds. This suggests that the layered ARX-based transformation structure effectively propagates local input differences throughout the 512-bit state. Although diffusion converges quickly, the cipher employs 24 rounds to provide additional mixing beyond the experimentally observed convergence point. The objective of the extended round count is to introduce a conservative margin, which is now mathematically supported by the theoretical bounds established against differential and linear cryptanalysis. Performance measurements show that this additional mixing incurs computational overhead relative to ChaCha20; however, the resulting throughput remains practical for software-based deployments. The combination of a large internal state, FSNI-based nonce handling, and domain separation further contributes to the robustness of the overall design. While the cipher demonstrates favourable diffusion characteristics, successfully passes standard statistical randomness tests, and is fortified by theoretical bounds on active additions, additional cryptanalytic analysis—such as resistance against algebraic and rotational attacks—and independent third-party evaluation remain necessary to fully establish the long-term security properties of the construction.

9 Limitations and Future Work

While this study establishes solid empirical foundations and theoretical bounds for VORTEX256, it must be interpreted within its defined scope. The cipher demonstrates favourable diffusion characteristics, successful statistical randomness evaluation, practical software performance, and mathematically derived resistance bounds against classical differential and linear cryptanalysis. However, formal resistance against other advanced vectors—such as algebraic, rotational, related-key, and distinguishability attacks—requires further specialized investigation. Furthermore, the design has not yet undergone independent third-party cryptanalysis or extensive hardware-level evaluation on embedded, FPGA, or ASIC platforms, which remains a critical validation step as demonstrated by recent high-performance hardware implementations of Grain and ChaCha20 [1,37]. Future work will therefore focus on extended cryptanalytic assessment against these remaining attack models, hardware implementations, and side-channel resistance evaluation. These investigations are necessary to further validate the security assumptions and practical applicability of the proposed construction.

10 Conclusion

This paper presented VORTEX256, an ARX-based stream cipher designed with an emphasis on efficient mixing, structural simplicity, and practical security considerations. The evaluation demonstrates that the cipher achieves rapid diffusion, with near-ideal avalanche behaviour observed within a small number of rounds. This indicates that the underlying transformation is effective in propagating input changes across the internal state. In addition to diffusion efficiency, the statistical analysis shows that the output of VORTEX256 remains stable across a wide range of inputs. The observed distribution of bit changes is close to the ideal value, with low variance, suggesting consistent and unbiased behaviour. Crucially, the paper establishes theoretical bounds against differential and linear cryptanalysis. By calculating the minimum number of active additions, we prove that the maximum differential and linear probabilities drop well below the 256-bit security threshold within 10 rounds, mathematically validating the 24-round structure as a highly conservative security margin. From a performance perspective, VORTEX256 demonstrates reasonable performance despite additional mixing stages and acceptable computational overhead when compared with established stream ciphers. Although the design incorporates a higher number of rounds, this choice reflects a deliberate trade-off aimed at maintaining a conservative security margin while preserving practical efficiency. Future efforts will focus on independent third-party cryptanalysis, evaluation on hardware platforms, and analysis of side-channel resistance. Overall, VORTEX256 represents a step toward balancing rapid diffusion, stable statistical behaviour, and practical performance within a unified design. While the current results are encouraging, continued analysis and validation will be essential for establishing its suitability in real-world cryptographic applications.

Acknowledgments

The authors utilised an Al-based language model tool ‘Gemini’ to have recommendations to revise and refine the sentences for improvements in sentence structure and grammatical correctness while writing the manuscript. This research work is not funded by any institution or funding agency or organization and does not involve any human or any kind biological subjects.

References

[1] P. Aditya et al. High-performance hardware structure of ChaCha20 stream cipher. *International Journal of Circuit Theory and Applications*, 2024.

[2] S. Afzal et al. LightAgriCrypt: A novel lightweight stream cipher for smart agriculture. *Security and Privacy Journal*, 2025.

[3] S. Afzal et al. LWLCM: A novel lightweight stream cipher using logistic mapping. *IEEE Access*, 2025.

[4] S. Afzal et al. A novel lightweight stream cipher using logistic chaos function for IoT security. *PLOS ONE*, 2025.

[5] J.-P. Aumasson, S. Fischer, S. Khazaei, W. Meier, and C. Rechberger. New features of latin dances: Analysis of salsa, chacha, and rumba. In *Fast Software Encryption (FSE 2008)*, volume 5086 of *LNCS*. Springer, 2008.

[6] D. Bakyt, M. Moldamurat, et al. Modification of the lightweight stream cipher algorithm ChaCha20. *Journal of Theoretical and Applied Information Technology*, 2025.

[7] M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. *Journal of Cryptology*, 21(4):469–491, 2008.

[8] D. J. Bernstein. The Poly1305-AES message-authentication code. In *Fast Software Encryption (FSE 2005)*, volume 3557 of *LNCS*, pages 32–49. Springer, 2005.

[9] D. J. Bernstein. Salsa20 specification. eSTREAM Project, 2005.

[10] D. J. Bernstein. ChaCha, a variant of Salsa20. In *Workshop Record of SASC 2008: The State of the Art of Stream Ciphers*, 2008.

[11] D. J. Bernstein. The ChaCha family of stream ciphers. Technical report, Technical Report, 2008.

[12] D. J. Bernstein. The Salsa20 family of stream ciphers. In *New Stream Cipher Designs*, volume 4986 of *LNCS*, pages 84–97. Springer, 2008.

[13] D. J. Bernstein. Extending the Salsa20 nonce. Cryptology ePrint Archive, IACR, 2011.

[14] D. J. Bernstein, T. Lange, and P. Schwabe. The security impact of a new cryptographic library. In *Progress in Cryptology – LATINCRYPT*. Springer, 2012.

[15] A. Biryukov, I. Dinur, and D. Khovratovich. Cryptanalysis of reduced-round Salsa20. In *Selected Areas in Cryptography (SAC)*. Springer.

[16] W. J. Buchanan and L. Maglaras. Review of the NIST lightweight cryptography finalists. arXiv:2303.14785, 2023.

[17] C. De Cannière and B. Preneel. Trivium. In *New Stream Cipher Designs*, volume 4986 of *LNCS*, pages 244–266. Springer, 2008.

[18] M. de Ree, G. Mantas, and J. Rodriguez. Grain-128PLE: Generic physical-layer encryption for IoT networks. arXiv:2309.15569, 2023.

[19] M. Hell et al. An AEAD variant of the grain stream cipher. In *C2SI Conference Proceedings*, 2019.

[20] M. Hell et al. Grain-128AEADv2: A lightweight AEAD stream cipher. NIST Lightweight Cryptography Project Specification, 2021.

[21] M. Hell, T. Johansson, A. Maximov, and W. Meier. A stream cipher proposal: Grain-128. In *Proc. IEEE ISIT*, 2006.

[22] M. Hell, T. Johansson, A. Maximov, W. Meier, and H. Yoshida. Grain-128AEADv2: Strengthening the initialization against key reconstruction. In *Cryptology and Network Security (CANS)*. Springer, 2021.

[23] M. Hell, T. Johansson, and W. Meier. Grain: A stream cipher for constrained environments. *International Journal of Wireless and Mobile Computing*, 2(1):86–93, 2007.

[24] S. Josefsson and I. Liusvaara. The Poly1305-AES message-authentication code. Technical Report RFC 7539 Supporting Material, Internet Engineering Task Force (IETF).

[25] J. Kaur, A. C. Canto, M. M. Kermani, and R. Azarderakhsh. A comprehensive survey on the implementations, attacks, and countermeasures of the current NIST lightweight cryptography standard. arXiv:2304.06222, 2023.

[26] V. Kebande. Extended-ChaCha20 stream cipher with enhanced quarter round function. *IEEE Access*, 2023.

[27] V. Kebande. Stringology-based cryptanalysis for EChaCha20 stream cipher. arXiv preprint, 2026.

[28] S. Khan et al. A comprehensive review on lightweight cryptographic algorithms for resource-constrained systems. *ACM Computing Surveys*, 2025.

[29] H. Lipmaa and S. Moriai. Secure design of addition-rotation-xor based ciphers, 2001.

[30] C. Liu et al. An improved ChaCha algorithm based on quantum random number injection. arXiv, 2025.

[31] H. Madushan et al. A review of the NIST lightweight cryptography finalists. *Electronics*, 11(24), 2022.

[32] A. Maximov. Software evaluation of Grain-128AEAD. Cryptology ePrint Archive, Report 2020/659.

[33] R. Mohanapriya et al. An optimized novel lightweight ARX cipher for image encryption. *Scientific Reports*, 2025.

[34] Y. Nir and A. Langley. ChaCha20 and Poly1305 for IETF protocols. Technical Report RFC 8439, Internet Engineering Task Force (IETF), June 2018.

[35] T. G. Sorescu et al. Comparative performance analysis of lightweight cryptographic algorithms. *IEEE Access*, 2025.

[36] J. Wallén. Linear approximations of addition modulo 2^n, 2003.

[37] Y. Wang et al. Efficient (masked) hardware implementation of Grain-128AEADv2. *Security and Communication Networks*, 2023.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 210–219.

Security Analysis of a Dual-Problem Blind Signature Scheme

Nursyaurah binti Saidin and Nurul Nur Hanisah Adenan

Malaysia Cryptology Technology and Management Centre, c/o Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia {nursyaurah,hanisahadenan}@upm.edu.my

Abstract. This paper presents a formal security analysis of the dual-problem blind signature scheme of Tahat et. al [5] which combines the Integer Factorisation Problem (IFP) and the Discrete Logarithm Problem (DLP) to improve cryptographic resilience.

We demonstrated one-more unforgeability under the DLP in the Random Oracle Model (ROM), and blindness property. Furthermore, the protocol retains the low-computational-complexity structure of prior Schnorr-type blind signatures, suggesting potential suitability for applications such as electronic cash or anonymous voting.

Keywords: Blind Signatures · Blindness · One-More Unforgeability · Integer Factorisation Problem · Discrete Logarithm Problem

1 Introduction

A blind signature is a digital signature scheme in which a signer produces a valid signature on a message without learning the content of that message and without being able to link the signature to the signing session that produced it. The blind signature scheme involves three parties, including the requester, the signer, and the verifier. The requester obtains a signature on a message while keeping the message hidden from the signer. The signer generates the signature using a private key without learning the message being signed and the verifier determines whether the resulting message–signature pair is valid. Figure 1 shows the resulting workflow and the roles and message flow among the three parties [12].

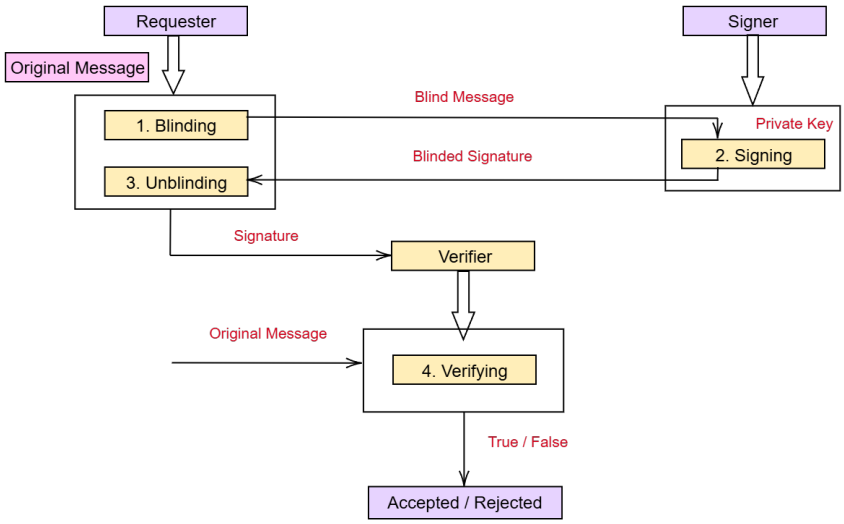


Figure 1: Blind Signature

The concept of a blind signature scheme was first developed by Chaum [2]. There were many proposals for blind signature schemes published based on a single hard problem such as integer factorisation problem (IFP) or discrete logarithm problem (DLP). A single carries an inhere hard problem have a risk: if the underlying problem is ever solved, every signature ever issued under that scheme loses its security guarantee at once. Combining two different hard problems removes this single point of failure, since an adversary would then need to solve both simultaneously to forge a signature or break anonymity. Tahat et al. [5] proposed such a dual-problem blind signature scheme, combining IFP and DLP in a three-move protocol between requester and signer that uses two blinding factors (α, β) to keep the signer's view statistically independent of the resulting message–signature pair.

In this paper, our aim is to determine the adversary's capability to break the scheme that can be transformed into an algorithm that solves these hard mathematical problems. We provide a formal security analysis of a new blind signature scheme based on IFP and DLP by Tahat et al. [5].

The paper is organised as follows. Section 2 provides an overview of single and multi-hard-problem blind signature constructions and the security literature most relevant to our analysis. Section 3 briefly describes the preliminaries of the Random Oracle Model (ROM), the relevant lemmas, and the scheme of [5] itself. Section 4 presents our security analysis, and we conclude our work in Section 5.

2 Related Works

The concept of a blind signature was introduced by Chaum [2], who illustrated the property of using the analogy of a carbon-lined envelope: a requester places a ballot inside the envelope, a trustee signs the outside, and the signature is transferred onto the ballot through the carbon lining, leaving the trustee unaware of what it has signed.

The security of blind signature schemes was originally derived from a single computational hardness assumption, generally IFP or DLP. The previous schemes such as [2] were built relying on the RSA (factoring) problem: given a modulus n, a public exponent e, and a private exponent d, a signature on a message m is computed as $\sigma = H(m)^d \pmod{n}$. A requester blinds $H(m)$ with a random value $r^e \pmod{n}$ before sending it to

the signer, and unblinds the returned value by multiplying by $r^{-1} \pmod{n}$ to recover a valid signature [9].

DLP-based constructions followed a different template, built around identification schemes such as Schnorr's and the ElGamal signature family. Shen et al. [13] presented blind signature schemes derived from variants of the Digital Signature Algorithm (DSA) [8] and the Nyberg–Rueppel signature scheme [7], both of which recover a hash of the message from the signature via the discrete-logarithm relation $y \equiv g^x \pmod{p}$. Pointcheval and Stern [9] gave a provably secure blind variant of the Schnorr signature, proving one-more unforgeability in the Random Oracle Model via the Forking Lemma [10], and later generalised the technique to blind Guillou–Quisquater and other identification-scheme-based signatures [11]. Selim [12] surveys this line of ElGamal-type blind signature schemes in the context of electronic voting.

Both the RSA-based and DLP-based families share a common vulnerability: the scheme collapses entirely if its single underlying hard problem is ever solved. To remove this single point of failure, Ismail et al. [4] proposed a digital signature scheme whose forgery is tied to both the IFP and the DLP simultaneously, so that an adversary must solve both problems to produce a forgery. Tahat et al. [5] extended this dual-problem paradigm to the blind-signature setting, producing the three-move, two-blinding-factor (α, β) protocol that is the subject of this paper: it retains the commitment–challenge–response shape of Schnorr-type schemes but binds the signer's response to an RSA-type secret key as well as a discrete-logarithm secret key, with the aim of forcing an adversary to break both IFP and DLP in order to forge a signature.

3 Preliminaries

This section presents the preliminary concepts and definitions required throughout this paper. It provides a brief overview of the random oracle model, random oracle replay attack, one-more unforgeability, blindness, and the blind signature scheme proposed by Tahat et al. [5]. These concepts establish the theoretical foundation for the security analysis and the proposed construction presented in the subsequent sections.

3.1 Random Oracle Model (ROM)

Since hash functions often present an obstacle to mathematical proofs, many security proofs utilize the ROM. ROM is a widely used idealized framework in cryptography. ROM treats a hash function as if it were a perfectly random function that everyone can query, including adversaries. This model allows designers to validate cryptographic designs and eliminate "poor" ones, proving security based on basic properties such as one-wayness [1].

3.2 The "one-more" forgery

Definition 1 ([11])**.** For some integer ℓ polynomial in the security parameter k, an adversary can obtain $\ell + 1$ valid signatures after fewer than ℓ interactions with the signer. In other words, a "one-more forgery" is an $(\ell, \ell+1)$- forgery for some polynomially bounded integer ℓ .

3.3 The Oracle Replay Attack

The following lemma uses the "oracle replay attack" that involves executing a simulated attack against a signature scheme twice using the exact same internal random variables, such as random tapes and secret keys, to ensure that the adversary's behavior remains consistent until a specific point.

Lemma 1 (The Forking Lemma [10]). *Let $\mathcal{A}$ be a probabilistic polynomial-time Turing machine, given only the public data as input. If $\mathcal{A}$ can find, with non-negligible probability, a valid signature $(m, \sigma_1, h, \sigma_2)$, then, with non-negligible probability, a replay of this machine, with the same random tape and a different oracle, outputs two valid signatures $(m, \sigma_1, h, \sigma_2)$ and $(m, \sigma_1, h', \sigma_2')$ such that $h \neq h'$.*

Remark. Probabilities are taken over random tapes, random oracles, and in some cases, over messages and keys.

3.4 One-more unforgeability (OMUF)

This definition below is essential for blind signatures to prevent inflation attacks where a requester might attempt to generate additional valid tokens from a limited number of legitimate signing sessions.

Definition 2. A signature scheme is said to be one-more unforgeable if it is computationally infeasible for an adversary $\mathcal{A}$ to output $k+1$ valid message-signature pairs after interacting with a Signing Oracle (the Signer) exactly k times.

3.4.1 The OMUF Experiment

1. Setup: The $\mathcal{C}$ generates a key pair (pk, sk) and gives the public key pk to the $\mathcal{A}$.

2. The Quota (k): The $\mathcal{A}$ is allowed to start k separate protocol sessions with the $\mathcal{C}$ (the Signer). In each session, the $\mathcal{A}$ provides a blinded message, and the $\mathcal{C}$ provides a blinded signature.

3. The Output: At the end of the experiment, the $\mathcal{A}$ must output a set of N valid signatures: $\{(m_1, \sigma_1), (m_2, \sigma_2), \ldots, (m_N, \sigma_N)\}$.

4. Winning Condition: The $\mathcal{A}$ wins if $N > k$ (i.e., they produced $k+1$ signatures).

3.5 Blindness

Definition 3 (Blindness [3]). Let $\mathcal{S}^*$ as malicious signer that running algorithm $\mathcal{A}$. An interactive signature scheme is called *blind* if for any efficient (polynomial-time) algorithm $\mathcal{A}$, the probability of $\mathcal{A}$ winning the following game is negligibly close to $\frac{1}{2}$.

1. $\mathcal{A}$, for security parameter λ, generates a public key pk and two messages m_0 and m_1, and submits them to the challenger.

2. Challenger, $\mathcal{C}$ chooses $b \in \{0, 1\}$ and returns the blinded messages $\mathcal{U}(pk, m_b)$ and $\mathcal{U}(pk, m_{1-b})$.

3. $\mathcal{A}$ queries $\sigma_b = \mathcal{S}^*(sk, \mathcal{U}(pk, m_b))$ and then $\sigma_{1-b} = \mathcal{S}^*(sk, \mathcal{U}(pk, m_{1-b}))$. If either one is $\perp$,then set $(\sigma_b, \sigma_{1-b}) = (\perp, \perp)$. Give $\mathcal{A}$ (σ_b, σ_{1-b}) .

4. $\mathcal{A}$ outputs a guess b'.

$\mathcal{A}$ wins the game if $b' = b$.

3.6 Blind Signature Scheme by Tahat et. al [5]

The following notations and parameters describe the process of this basic structure: $h(.)$ is a cryptographic hash function whose output is a t-bit length and assume $t = 128$. Let an integer P be a large prime. Let the product of two safe primes be denoted as n while it is also a factor of $(P-1)$ and $\phi(n)$ is a phi-Euler function. An integer g is a primitive element in $Z_P^* = \{1, 2, \cdots, P-1\}$ with order n satisfying $g^n \equiv 1 \pmod{P}$ and let $\gcd(a, b)$ be the greatest common divisor of a and b . Now we describe the new blind signature scheme by Tahat et. al [5].

Algorithm 1 Initialization Phase

Input : System Parameters (P, n, g)
Output : Public key pair (e, y) and private key pair (d, x)

1: Pick randomly an integer $e \in \mathbb{Z}_n^* = \{1, 2, \ldots, n-1\}$ such that $\gcd(e, n) = 1$.
2: Calculate an integer d satisfying the congruence $ed \equiv 1 \pmod{\phi(n)}$.
3: Next select at random an integer x from $\mathbb{Z}_p^*$ and compute $y \equiv g^x \pmod{P}$.
4: Return (e, y) as a pair of verification key y , public key e, and keep (d, x) as a pair of secret key of the scheme.

Algorithm 2 Blind Signature Generation

Input : System Parameters (P, n, g), hash function h$(\cdot)$ and message m
Output : Blind signature $h(\hat{m})$ public key pair $(\hat{k}, \hat{r})$

1: Compute the hash of the message, m to be verified $h = h(m)$.
2: Select an integer $\hat{r} \in (1, n)$ such that $\gcd(\hat{r}, n) = 1$.
3: Compute $\hat{k} = g^{\hat{r}} \pmod{P}$.
4: **if** $\gcd(\hat{k}, n) \neq 1$ **then**
5: Repeat **Step 1**
6: **else**
7: Send $\hat{k}$ to the signature-requester
8: **end if**
9: Signature-requester checks if $\gcd(\hat{k}, n) = 1$.
10: Select blinding factors $\alpha, \beta \in \mathbb{Z}_n^*$.
11: Compute $k \equiv \hat{k}^{\alpha} g^{\beta} \pmod{P}$.
12: **if** $\gcd(k, n) \neq 1$ **then**
13: Repeat **Step 9**
14: **else**
15: Compute $h(\hat{m}) \equiv \alpha^{-1} h \hat{k} k^{-1} \pmod{n}$
16: **end if**
17: Send blind signature $h(\hat{m})$ and public key pair $(\hat{k}, \hat{r})$ to the signer.

Algorithm 3 Signature Phase

Input : Blind signature $\hat{h(m)}$ and public key pair $(\hat{k}, \hat{r})$ to the signer
Output : Signature key pair $(k.u)$

1: Signer computes $\hat{s} \equiv h(\hat{m})x + \hat{k}\hat{r} \pmod{n}$.
2: Send $\hat{s}$ to the signature-requester.
3: Signature-requester computes $s \equiv (\alpha\hat{s}k\hat{k}^{-1} + \beta k)(\hat{s}^{-1})^e \pmod{n}$
4: Send s to the signer.
5: Signer computes $\hat{u} \equiv s^d \pmod{n}$.
6: Send $\hat{u}$ to the signature-requester.
7: Signature-requester computes $u \equiv \hat{u}\hat{s} \pmod{n}$.
8: Output signature key pair (k, u).

Algorithm 4 Signature Verification

Input : Signature key pair (k, u), verification key y, system parameter e, hash function $h(\cdot)$ and message m.
Output : Accept or reject signature.

1: Compute the hash of the message m, $h = h(m)$.
2: Compute $Y_1 = g^{u^e} \pmod{P}$
3: Compute $Y_2 = y^h k^k \pmod{P}$
4: **if** $Y_1 = Y_2$, **then**
5: Accept signature
6: **else**
7: Reject signature $\perp$
8: **end if**

Correctness. If (k, u) is a signature of the message m produced by the proposed new blind signature scheme, then $g^{u^e} \equiv y^{h(m)}k^k \pmod{P}$, and the protocol above is a blind scheme.

Proof. Note that

$$\begin{aligned} u^e &\equiv (\hat{u}\hat{s})^e \equiv (s^d\hat{s})^e \equiv s\hat{s}^e \\ &\equiv (\alpha s\hat{k}k^{-1} + \beta k)(\hat{s}^{-1})^e\hat{s}^e \\ &\equiv \alpha\big(h(\hat{m})x + \hat{k}\hat{r}\big)k\hat{k}^{-1} + \beta k \\ &\equiv \alpha k\hat{k}^{-1}\Big((\alpha^{-1}h(m)\hat{k}k^{-1})x + \hat{k}\hat{r}\Big) + \beta k \\ &= h(m)x + \alpha k\hat{r} + \beta k. \end{aligned}$$

Thus,

$$\begin{aligned} g^{u^e} &\equiv g^{h(m)x+\alpha k\hat{r}+\beta k} \\ &\equiv (g^x)^{h(m)}(g^{\alpha\hat{r}+\beta})^k \\ &\equiv y^{h(m)}(\hat{k}^\alpha g^\beta)^k \\ &\equiv y^{h(m)}k^k \pmod{P}. \end{aligned}$$

□

4 Security Proofs

In this section, we discuss some security properties of a new blind signature scheme. A secure blind signature scheme should satisfy the following requirements [6].

4.1 One-More Unforgeability

To establish the one-more unforgeability of the scheme, we perform a reduction to the DLP. Using the Forking Lemma, we show that an adversary capable of producing an extra signature can be 'forked' to reveal the signer's secret key. The success of this extraction is guaranteed by the Lemma [1], which ensures that for a non-negligible fraction of random tapes, the adversary will respond successfully to a second, distinct hash challenge.

Theorem 1. *Assume $\mathcal{A}$ be a probabilistic polynomial-time adversary that interacts* sequentially *with the signing oracle (i.e. it opens a new signing session only after the previous one has closed). Suppose $\mathcal{A}$ succeeds in a one-more forgery with non-negligible probability ε, making at most ℓ signing sessions and at most Q queries to the random oracle $h(m)$. Then there is a probabilistic polynomial-time algorithm $\mathcal{C}$ that, given discrete logarithm problem (DLP) is hard, the Tahat et al. blind signature scheme achieve one-more unforgeability property.*

Proof. Let the adversary $\mathcal{A}$ is malicious user and tries to get one-more signature. The following algorithm show the security game of one-more unforgeability for Blind Signature Scheme by Tahat et al.

Algorithm 5 Security Game of One-More Unforgeability for Tahat et al. Scheme

Setup Phase

1: The Challenger $\mathcal{C}$ generates the public key, $pk : (e, y)$ and give to the $\mathcal{A}$ and keep private key, sk.

Signing Queries

2: For any chosen session, the $\mathcal{A}$ sends a blinded message $h(\hat{m})$ and a public key pair $(\hat{k}, \hat{r})$ to the $\mathcal{C}$.

3: The $\mathcal{A}$ computes $\hat{s} \equiv h(\hat{m})x + \hat{k}\hat{r} \pmod{n}$ and returns it.

4: The $\mathcal{A}$ send a value s to the $\mathcal{A}$. The $\mathcal{A}$ computes $\hat{u} \equiv s^d \pmod{n}$, returns $\hat{u}$ to the $\mathcal{A}$, and marks this specific execution as completed.

5: The $\mathcal{A}$ adaptively decides when to halt these queries. Let ℓ denote the total number of executions that the $\mathcal{A}$ marked as completed.

Forgery Output

6: The $\mathcal{A}$ outputs a set of j message-signature pairs:

$$\{(m_1, (k_1, u_1)), (m_2, (k_2, u_2)), \ldots, (m_j, (k_j, u_j))\}$$

.

$\mathcal{A}$ wins the game if the total number of valid signatures produced (j) is strictly greater than the number of completed interactive signing sessions (ℓ). Therefore, the probability that $j > \ell$ is at most ϵ.

If the forgery is valid, the $\mathcal{C}$ applies the Forking Lemma to obtain two valid signatures on the same message with different hash values. After a polynomial replay of the $\mathcal{A}$, it obtains two valid signatures $(m, k, h(m), u)$ and $(m, k, h(m)', u')$ with $h(m) \neq h(m)'$.

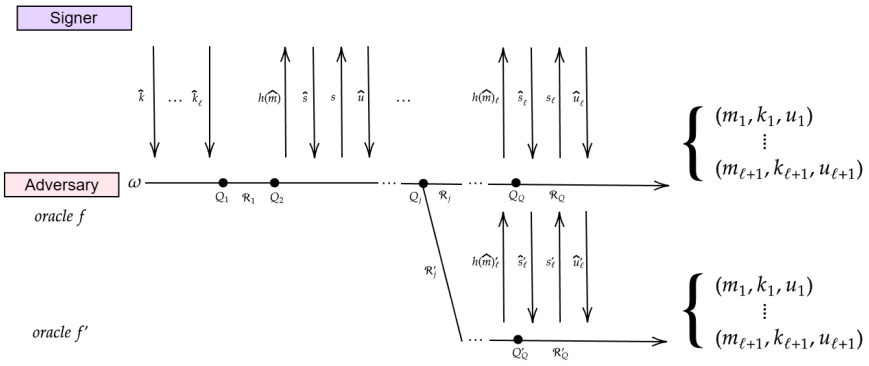


Figure 2: Forking Lemma

Then we have the following equalities,

$$g^{u^e} \equiv y^{h(m)}k^k \pmod{P}$$
$$g^{u'^e} \equiv y^{h(m)'}k^k \pmod{P}$$

from which we obtain the DLP

$$x \equiv (u^e - u'^e)[h(m) - h(m)']^{-1} \pmod{P-1}.$$

There is x such that $y \equiv g^x \pmod{P}$, then

$$g^{u^e} \equiv g^{xh(m)}k^k \pmod{P}$$
$$g^{u'^e} \equiv g^{xh(m)'}k^k \pmod{P}$$

Since $\hat{k} \equiv g^{\hat{r}} \pmod{P}$ and $k \equiv \hat{k}^\alpha g^\beta \pmod{P}$, then

$$g^{u^e} \equiv g^{xh(m)}g^{k(\alpha\hat{r}+\beta)} \pmod{P} \quad (1)$$
$$g^{u'^e} \equiv g^{xh(m)'}g^{k(\alpha\hat{r}+\beta)} \pmod{P} \quad (2)$$

Then (1) divide (2)

$$g^{u^e - u'^e} \equiv g^{x[h(m)-h(m)']} \pmod{P}$$

Since g is a generator of $\mathbb{Z}_p^*$, there exist x such that $y \equiv g^x \pmod{P}$. Therefore,

$$u^e - u'^e \equiv x[h(m) - h(m)'] \pmod{P-1}$$
$$x \equiv (u^e - u'^e)[h(m) - h(m)']^{-1} \pmod{P-1}$$

Therefore,

$$y \equiv g^x \pmod{P}$$

□

4.2 Blindness

Theorem 2. *Suppose the hash function h(m) are modeled as random oracles. Given discrete logarithm problem (DLP) is hard, the Tahat et al. blind signature scheme achieve blindness property.*

Proof. Let $b \in \{0,1\}$(i.e. b is a random bit which is kept secret from the adversary $\mathcal{A}$). The $\mathcal{A}$ executes the following experiment (where $\mathcal{A}$ act as malicious signer and tries to predict b). The system parameters (P, n, g) and a cryptographic hash function $h(\cdot)$ are generated.The key generation algorithm $Gen(1^k)$ is run to produce the signer's public key $pk = (e, y)$ and private key $sk = (d, x)$, where $ed \equiv 1 \pmod{\phi(n)}$ and $y \equiv g^x \pmod{P}$.The adversary $\mathcal{A}$ is given both pk and sk, as it completely controls the signer in this experiment.

$\mathcal{A}$ produces two target messages, m_0 and m_1 and send to the Challenger $\mathcal{C}$. The $\mathcal{C}$ selects a random secret bit $b \in_R \{0,1\}$. The $\mathcal{C}$ assigns m_b and m_{1-b} and send to the $\mathcal{A}$. At the same time, the $\mathcal{C}$ make H-query to random oracle to get hash value of m_o and m_1 which are $h(m_0)$ and $h(m_1)$.

The $\mathcal{A}$ chooses $\hat{r}_j$ where $j \in \{b, 1-b\}$, computes $\hat{k}_j \equiv g^{\hat{r}_j} \pmod{P}$, and sends $\hat{k}_j$ to the $\mathcal{C}$. The $\mathcal{C}$ chooses blinding factors $\alpha_j, \beta_j \in \mathbb{Z}_n^*$, computes $k_j \equiv \hat{k}_j^{\alpha_j} g^{\beta_j} \pmod{P}$, and generates the blinded message $h(\hat{m}_j) \equiv \alpha_j^{-1} h(m)_j \ \hat{k}_j \ k_j^{-1} \pmod{n}$. The $\mathcal{C}$ sends $h(\hat{m}_j)$ to the $\mathcal{A}$. The $\mathcal{A}$ computes $\hat{s}_j \equiv h(\hat{m}_j)x + \hat{k}_j \hat{r}_j \pmod{n}$ and sends it to the $\mathcal{C}$. The $\mathcal{C}$ computes $s_j \equiv (\alpha_j \hat{s}_j \hat{k}_j k_j^{-1} + \beta_j k_j)(\hat{s}_j^{-1})^e \pmod{n}$ and sends s_j to the $\mathcal{A}$. The $\mathcal{A}$ computes $\hat{u}_j \equiv s_j^d \pmod{n}$ and sends $\hat{u}_j$ to the $\mathcal{C}$. The $\mathcal{C}$ unblinds the signature to compute $u_j \equiv \hat{u}_j \hat{s}_j \pmod{n}$, resulting in the final signature pair (k_j, u_j).

These signatures, $\sigma(m_b) = (k_b, u_b)$ and $\sigma(m_{1-b}) = (k_{1-b}, u_{1-b})$, are provided to the $\mathcal{A}$ strictly ordered according to bit b that chosen by $\mathcal{C}$. The $\mathcal{A}$ outputs a bit $\tilde{b}$. The $\mathcal{A}$ wins the game if it successfully links the interactive sessions to their respective signatures, meaning $\tilde{b} = b$. Therefore, the probability, taken over the choice of b, that $\tilde{b} = b$ is at most $\frac{1}{2} + \epsilon$. □

5 Conclusion and Future Work

In this paper, we analyse a blind signature scheme that combines two difficult mathematical problems: IFP and DLP. By using both, the scheme stays secure even if one of these math problems is eventually solved under certain conditions. Our analysis confirms that the protocol successfully meets the key requirements for a secure system:

- Blindness: The signer cannot see the content of the message or track which requester they are signing for, ensuring total anonymity.
- OMUF: It is mathematically impossible for a requester to create extra, fake signatures beyond what the signer authorized.

References

[1] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pages 62–73, 1993.

[2] David Chaum. Blind signatures for untraceable payments. In *Advances in Cryptology: Proceedings of Crypto 82*, pages 199–203. Springer, 1983.

[3] Lucjan Hanzlik. A note on the blindness of the scheme from eprint 2025/397. *Cryptology ePrint Archive*, 2025.

[4] Eddie Shahrie Ismail, NMF Tahat, and Rokiah R Ahmad. A new digital signature scheme based on factoring and discrete logarithms. *Journal of mathematics and statistics*, 4(4):222, 2008.

[5] Eddie Shahrie Ismail, NMF Tahat, and Rokiah R Ahmad. A new blind signature scheme based on factoring and discrete logarithms. *International Journal of Cryptology Research*, 1(1):1–9, 2009.

[6] Ari Juels, Michael Luby, and Rafail Ostrovsky. Security of blind digital signatures. In *Annual International Cryptology Conference*, pages 150–164. Springer, 1997.

[7] Kaisa Nyberg and Rainer A Rueppel. Message recovery for signature schemes based on the discrete logarithm problem. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 182–193. Springer, 1994.

[8] Thomas Petri. Digital signature standard (dss). *National Institute of Standards and Technology*, 2013.

[9] David Pointcheval and Jacques Stern. Provably secure blind signature schemes. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 252–265. Springer, 1996.

[10] David Pointcheval and Jacques Stern. Security proofs for signature schemes. In *International conference on the theory and applications of cryptographic techniques*, pages 387–398. Springer, 1996.

[11] David Pointcheval and Jacques Stern. New blind signatures equivalent to factorization. In *Proceedings of the 4th ACM Conference on Computer and Communications Security*, pages 92–99, 1997.

[12] Mazen Selim. Blind signature schemes based on elgamal signature for electronic voting: a survey. *International Journal of Computer Applications*, 2018.

[13] Victor RL Shen, Yu Fang Chung, Tzer Shyong Chen, and Yu An Lin. A blind signature based on discrete logarithm problem. *International Journal of Innovative Computing, Information and Control*, 7(9):5403–5416, 2011.

Revisiting the LUCELG and Cramer–Shoup Analogous Cryptosystem: A Formal Security Analysis

Putri Zalia Megat Johar[1] and Muhammad Asyraf Asbullah[1,2]

[1] Malaysia Cryptology Technology and Management Centre, c/o Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia {putrizalia,ma_asyraf}@upm.edu.my
[2] Centre of Foundation Studies for Agricultural Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia ma_asyraf@upm.edu.my

Abstract. This paper presents a rigorous provable security analysis of a public key cryptosystem analogous to both LUCELG and the Cramer–Shoup scheme. Although this hybrid system was designed to leverage the algebraic advantages of Lucas functions alongside hash-based verification, previous security proofs lacked the formalization required by modern cryptographic standards. These analytical gaps are addressed by systematically defining the underlying Lucas-based hardness assumptions, including the Lucas-Computational Diffie–Hellman (LCDH) and Lucas-Decisional Diffie–Hellman (LDDH) problems. Through a game-based framework, formal mathematical reductions are established to prove the resilience of the scheme against passive attacks, specifically regarding total break and one-wayness (OW-CPA). Furthermore, a corrected, simulator-based reduction is provided to demonstrate indistinguishability under adaptive chosen ciphertext attacks (IND-CCA). The analysis confirms that the cryptosystem achieves strong, provable security guarantees while maintaining the computational efficiency inherent in Lucas sequences. Since the hardness of Lucas-based schemes reduces to the discrete logarithm problem in $\mathbb{F}_{p^2}$, the scheme is best understood as a structurally distinct and computationally efficient alternative within the discrete logarithm setting, rather than one based on a strictly harder assumption.

Keywords: Lucas Function · Provable Security · Lucas problem · IND-CCA

1 Introduction

The foundational security of public key cryptography relies heavily on the assumed intractability of specific mathematical hard problems. Historically, the discrete logarithm problem (DLP) has served as a primary cornerstone for such systems, most notably in the ElGamal cryptosystem [4] and the Diffie–Hellman key agreement protocol [3]. However, the cryptographic community has continually sought alternative mathematical structures to diversify security assumptions and potentially improve computational profiles. It was demonstrated that ciphers traditionally based on the discrete logarithm problem can also be robustly implemented using Lucas functions instead of standard modular exponentiation.

The exploration of Lucas functions in cryptography gained significant traction with the development of the LUC public key cryptosystem, introduced by [14]. The Lucas function operates as a special form of a second-order linear recurrence relation utilizing a large public integer as a modulus [7]. This unique mathematical framework proved highly versatile, allowing researchers to implement direct analogs of several foundational cryptographic schemes, including the ElGamal cryptosystem [4], the Diffie–Hellman key

agreement protocol [3], and the RSA cryptosystem [12] over Lucas sequences [10, 15]. In the ElGamal cryptosystem, security relies on the difficulty of computing discrete logarithms, and based on the same hard mathematical problem, LUCELG uses the calculation of Lucas functions instead of discrete logarithms.

The security of Lucas-based cryptosystems has since been examined from several directions. Bleichenbacher, Bosma and Lenstra [1] showed that the discrete logarithm problem for Lucas functions can be transferred into the multiplicative group of $\mathbb{F}_{p^2}$, so Lucas-based analogues of ElGamal and Diffie–Hellman, including LUCELG, admit the same subexponential attacks as their classical counterparts. Consequently, the parameters of any Lucas-based scheme must be chosen so that the discrete logarithm problem in $\mathbb{F}_{p^2}$ remains infeasible. Laih, Tu and Tai further investigated structural properties of the LUC system and the one-wayness of the Lucas function itself [7, 8]. Subsequent work developed concrete attacks on LUC-type systems, including a cryptanalytic attack on LUC with short secret exponents using continued fractions [5]. A more recent line of research has evaluated LUC, LUC3 and LUC4,6 and their ElGamal-type variants against specific attack strategies, including chosen message attacks [17], common modulus attacks [16], and greatest common divisor attacks on LUCELG, LUC(3)ELG and LUC(4,6)ELG [13]. These studies share a common methodology. Each selects a particular attack and tests whether a given Lucas-based scheme resists it. None of them relates the security of a complete encryption scheme to a precisely defined Lucas hardness assumption through a formal reduction.

Despite the theoretical elegance of ElGamal and its Lucas-based counterparts such as LUCELG, these fundamental schemes share a vulnerability to sophisticated active threats, specifically adaptive chosen ciphertext attacks (CCA2). To address these vulnerabilities in standard group-theoretic cryptography, Ronald Cramer and Victor Shoup developed the Cramer–Shoup cryptosystem in 1998 [2]. By extending the basic ElGamal scheme with a hash-based verification mechanism, the Cramer–Shoup scheme and its variants became highly practical and were rigorously proven secure against adaptive chosen ciphertext attacks under standard intractability assumptions. The Cramer–Shoup design remains an active subject of study. Recent work includes a faster variant with a full IND-CCA2 proof [6] and the cryptanalysis of Cramer–Shoup-like constructions instantiated over alternative algebraic structures [9]. The latter result is instructive: it demonstrates that transplanting the Cramer–Shoup design into a new algebraic setting does not automatically preserve its security guarantees, and that each instantiation requires its own formal proof.

Following the trajectory of these advancements, Muslim and Said [11] proposed a hybrid cryptosystem analogous to both LUCELG and Cramer–Shoup. This integrated system leverages group-based operations similar to those found in LUCELG while simultaneously incorporating the hash-based verification mechanisms introduced by Cramer and Shoup. The primary objective of this integration was to provide the alternative mathematical foundation of Lucas functions while securing the scheme against adaptive chosen ciphertext attacks.

1.1 Problem Statement and Research Gap

The literature surveyed above reveals a clear division. On one side, the security of Lucas-based cryptosystems has been studied almost exclusively through attack-specific analyses [1, 5, 17, 16, 13], which establish resistance or vulnerability to individual attack strategies but do not yield general security guarantees. On the other side, the provable security methodology developed for Cramer–Shoup and its successors [2, 6, 9] delivers such guarantees, but it has been applied to classical groups and other algebraic structures rather than to Lucas functions. The scheme of Muslim and Said [11] sits at the intersection of these two lines of work, yet neither tradition has been fully applied to its security analysis.

Specifically, three gaps remain open. First, the original proof asserting security against

adaptive chosen ciphertext attacks (IND-CCA) lacked the rigorous formalization required by modern cryptographic standards. The original work did not provide a complete simulator-based reduction, nor did it strictly define how the public key, the challenge ciphertext, and the decryption queries are simulated in a reduction to the underlying Lucas-Decisional Diffie–Hellman (LDDH) assumption. Second, the hardness assumptions on which the scheme rests, namely the Single Lucas, Double Lucas, Lucas-Computational Diffie–Hellman (LCDH) and LDDH problems, were never stated as precise computational problems, so no reduction could be meaningfully formulated. Third, the security properties under passive attacks were never established formally. Notions such as total break and one-way encryption were not related to the underlying hardness assumptions through explicit reductions. As the cryptanalysis of Cramer–Shoup-like schemes over alternative structures has shown [9], the absence of such proofs is not a formality: a design that is secure in one algebraic setting can fail in another. To confidently deploy such a scheme, it is imperative to revisit the cryptosystem and construct a mathematically sound, formal proof of its security bounds.

1.2 Objective

The objective of this paper is to construct a complete and formal security analysis of the LUCELG and Cramer–Shoup analogous cryptosystem proposed by Muslim and Said [11]. In particular, the paper aims to (i) formalize the Lucas-based hardness assumptions underlying the scheme, (ii) establish formal reductions relating passive adversaries to these assumptions, covering total break and one-way encryption, and (iii) construct a corrected, simulator-based reduction proving indistinguishability under adaptive chosen ciphertext attacks (IND-CCA). This closes the gaps identified in Section 1.1 and places the scheme on the same provable footing as the classical Cramer–Shoup cryptosystem.

1.3 Contributions

In this paper, the LUCELG and Cramer–Shoup analogous cryptosystem proposed in [11] is revisited, and its security analysis is corrected and formalized. The key generation, encryption, and decryption processes are systematically studied, and the theoretical security bounds are formalized against both passive and active adversaries.

Specifically, the main contributions are summarized as follows:

- **Correction of the IND-CCA Proof:** The indistinguishability under chosen ciphertext attack (IND-CCA) proof of the original scheme is corrected. A rigorous, simulator-based reduction is constructed, and the simulation of the public key, the challenge ciphertext, and the decryption oracle is explicitly specified.
- **Formalization of Lucas Hardness Assumptions:** The underlying hardness assumptions required for the scheme's security are formalized, including the Single Lucas Problem, Double Lucas Problem, Lucas-Computational Diffie–Hellman (LCDH) problem, and the Lucas-Decisional Diffie–Hellman (LDDH) problem.
- **Analysis Under Passive Attacks:** Formal mathematical reductions are established to strictly relate the capabilities of successful passive adversaries to the aforementioned Lucas-based hardness assumptions. Security notions including total break and one-way encryption are analyzed through explicit reductions to these problems.

Ultimately, the formal analysis rectifies the gaps in the original literature and confirms that the proposed cryptosystem satisfies strong, provable security guarantees against passive attacks and adaptive chosen ciphertext attacks, while effectively maintaining efficient operations.

2 Preliminaries

2.1 Lucas Function

Lucas functions are a special class of higher-order linear recurrences. A sequence of integers $\{T_n\}$ can be defined by an m-th order linear recurrence relation:

$$T_n = P_1 T_{n-1} + P_2 T_{n-2} + \cdots + P_m T_{n-m},$$

where $P_1, P_2, \ldots, P_m$ are integers. The initial terms $T_0, T_1, \ldots, T_{m-1}$ must be defined independently to generate the sequence.

For the purposes of the LUCELG cryptosystem, attention is restricted to second-order linear recurrence relations of the form:

$$T_n = PT_{n-1} - QT_{n-2},$$

where P and Q are relatively prime integers [14].

Let α and β be the roots of the characteristic equation associated with the recurrence:

$$x^2 - Px + Q = 0.$$

From this equation, the following relationships hold:

$$P = \alpha + \beta, \qquad Q = \alpha\beta.$$

2.1.1 The Lucas Functions U_n and V_n

Two particular solutions of the recurrence are of special interest in cryptography. These are the Lucas functions $U_n(P, Q)$ and $V_n(P, Q)$, defined as:

$$U_n(P, Q) = \frac{\alpha^n - \beta^n}{\alpha - \beta}, \qquad V_n(P, Q) = \alpha^n + \beta^n.$$

These definitions yield integer sequences with the following initial values:

$$U_0 = 0,\ U_1 = 1, \qquad V_0 = 2,\ V_1 = P.$$

An important property for cryptographic applications is that Lucas functions behave well under reduction modulo an integer N. For any integer N, the following congruences hold [14]:

$$U_n(P \bmod N, Q \bmod N) \equiv U_n(P, Q) \pmod{N}, \qquad V_n(P \bmod N, Q \bmod N) \equiv V_n(P, Q) \pmod{N}.$$

The most crucial property for public-key cryptography is the composition rule, which mirrors the behaviour of exponentiation:

$$V_n(V_k(P, Q), Q^k) = V_{nk}(P, Q).$$

When $Q = 1$, the composition property simplifies considerably. In this case, $\alpha\beta = 1$, so $\beta = \alpha^{-1}$, and the following relationship is obtained:

$$V_{nk}(P, 1) = V_n(V_k(P, 1), 1).$$

This special case is particularly useful in cryptographic implementations, as it reduces computational complexity while preserving the security properties of the general Lucas functions [14].

2.2 Cryptosystem Analogous to LUCELG and Cramer–Shoup

Following the work of [11], a cryptosystem analogous to LUCELG and Cramer–Shoup integrates features from both schemes. The system employs group-based operations similar to LUCELG while incorporating hash-based mechanisms inspired by the Cramer–Shoup cryptosystem, thereby achieving security against adaptive chosen ciphertext attacks.

In the key generation phase, a prime p is selected together with initial values P_1 and P_2, with $Q = 1$. Random elements $x_1, x_2, y_1, y_2, z \in \mathbb{F}_{p^2}^{\star}$ are then chosen. Using these values, the components of the public key are computed as

$$c \equiv V_{x_1}(P_1, 1) \cdot V_{x_2}(P_2, 1) \pmod{p}, \qquad d \equiv V_{y_1}(P_1, 1) \cdot V_{y_2}(P_2, 1) \pmod{p}, \qquad h \equiv V_z(P_1, 1) \pmod{p}.$$

The public key is given by $pk = (P_1, P_2, c, d, h, p)$, while the secret key is defined as $sk = (x_1, x_2, y_1, y_2, z)$.

For the encryption process, a random value $k \in [1, p-1]$ is selected together with a message $m \in [1, p-1]$. The ciphertext components are then computed as

$$u_1 \equiv V_k(P_1, 1), \quad u_2 \equiv V_k(P_2, 1) \pmod{p}, \qquad G \equiv V_k(h, 1) \pmod{p} \equiv V_k(V_z(P_1, 1), 1),$$

$$e \equiv G \cdot m \pmod{p}, \qquad \alpha \equiv H(u_1, u_2, e), \qquad v \equiv V_k(c, 1) \cdot V_{k\alpha}(d, 1) \pmod{p}.$$

The resulting ciphertext is given by (u_1, u_2, e, v).

During decryption, the plaintext is recovered by computing

$$m \equiv \frac{e}{V_z(u_1, 1)} \pmod{p}.$$

3 Provable Security

3.1 Security Model and Hard Problem

Definition 1 (Single Lucas Problem)**.** Let p be a large prime and let $P \in \mathbb{F}_p$ be such that the Lucas sequence $\{V_n(P, 1)\}_{n \geq 0}$ is well-defined modulo p. Given P and an element

$$y \equiv V_x(P, 1) \pmod{p},$$

for a randomly chosen $x \in \mathbb{Z}$, the single Lucas problem is to compute the integer x.

Definition 2 (Double Lucas Problem)**.** Let p be a large prime and let $P \in \mathbb{F}_p$ be such that the Lucas sequence $\{V_n(P, 1)\}_{n \geq 0}$ is well-defined modulo p. Given P and two elements

$$y_1 \equiv V_{x_1}(P, 1) \pmod{p}, \qquad y_2 \equiv V_{x_2}(P, 1) \pmod{p},$$

for randomly chosen integers $x_1, x_2 \in \mathbb{Z}$, the Double Lucas Problem is to compute the pair (x_1, x_2).

Definition 3 (Lucas Computational Diffie–Hellman Problem)**.** Let p be a large prime and let $P \in \mathbb{F}_p$ be such that the Lucas sequence $\{V_n(P, 1)\}_{n \geq 0}$ is well-defined modulo p. Given

$$X \equiv V_x(P, 1) \pmod{p}, \qquad Y \equiv V_y(P, 1) \pmod{p},$$

for randomly chosen integers $x, y \in \mathbb{Z}$, the Lucas Computational Diffie–Hellman (Lucas–CDH) problem is to compute

$$V_{xy}(P, 1) \pmod{p}.$$

Definition 4 (Lucas-Decisional Diffie–Hellman Problem)**.** Let p be a large prime and let $P_1, P_2 \in \mathbb{F}_p$ be Lucas parameters with $Q = 1$. Define the two distributions

$$D = \big\{(P_1, P_2, V_k(P_1, 1), V_k(P_2, 1)) : k \text{ chosen at random}\big\},$$

$$\tilde{D} = \big\{(P_1, P_2, V_{k_1}(P_1, 1), V_{k_2}(P_2, 1)) : k_1, k_2 \text{ chosen at random}, k_1 \neq k_2\big\}.$$

Given a tuple (P_1, P_2, U_1, U_2) sampled from either D or $\tilde{D}$, the Lucas-Decisional Diffie–Hellman (LDDH) problem is to decide from which distribution the tuple was drawn.

Theorem 1. *If the single Lucas problem and the double Lucas problem are hard, then the cryptosystem is secure against total break under passive attack.*

Proof. Assume that there exists an adversary $\mathcal{A}$ that can recover the secret key from the public key with non-negligible probability $\varepsilon_{\mathcal{A}}(k)$. An algorithm $\mathcal{B}$ is constructed that uses $\mathcal{A}$ to solve both the single Lucas problem and the double Lucas problem.

[H] Reduction Algorithm $\mathcal{B}(U, W)$ [1] Lucas instance (U, W) where $U = V_s(P_1, 1)$ and $W = V_{r_1}(P_1, 1) \cdot V_{r_2}(P_2, 1)$ (s, r_1, r_2) Choose parameters P_1, P_2 with $Q = 1$ Pick random $y_1, y_2 \in \mathbb{F}_{p^2}^*$ $c \leftarrow W$ $d \leftarrow V_{y_1}(P_1, 1) \cdot V_{y_2}(P_2, 1)$ $h \leftarrow U$ Form public key (P_1, P_2, c, d, h) Give public key to adversary $\mathcal{A}$ Receive $(x_1', y_1', x_2', y_2', z')$ from $\mathcal{A}$ $V_{z'}(P_1, 1) = h$ $s \leftarrow z'$ $V_{x_1'}(P_1, 1) \cdot V_{x_2'}(P_2, 1) = c$ $(r_1, r_2) \leftarrow (x_1', x_2')$ (s, r_1, r_2)

The algorithm $\mathcal{B}$ embeds the given Lucas instance (U, W) into the public key by setting $h = U$ and $c = W$, while the remaining components are generated honestly. This ensures that the distribution of the simulated public key is identical to that of a real key generated by the system. Hence, from the perspective of the adversary $\mathcal{A}$, the simulation is indistinguishable from a real execution.

If the adversary $\mathcal{A}$ outputs a valid secret key, then by correctness of the scheme, the values must satisfy the defining equations of the public key. In particular, it follows that

$$s = z', \qquad (r_1, r_2) = (x_1', x_2').$$

Therefore, the algorithm $\mathcal{B}$ successfully recovers the solutions to both the single Lucas problem and the double Lucas problem.

Since $\mathcal{A}$ succeeds with non-negligible probability $\varepsilon_{\mathcal{A}}(k)$, the algorithm $\mathcal{B}$ also succeeds with non-negligible probability. This contradicts the assumed hardness of the single Lucas problem and the double Lucas problem. Therefore, the cryptosystem is secure against total break under passive attack. □

Theorem 2. *If the Lucas Computational Diffie-Hellman (LCDH) problem is hard, then the cryptosystem is secure under one-way encryption under passive attack.*

Proof. The statement is proved via a reduction between the decryption problem and the LCDH problem. In particular, it is shown that an efficient algorithm for decryption implies an efficient solution to the LCDH problem, and vice versa.

First, assume the existence of an oracle $\mathcal{O}$ that solves the LCDH problem in $\mathbb{F}_p^*$. An algorithm is constructed that uses $\mathcal{O}$ to decrypt any ciphertext.

[H] Decryption using LCDH Oracle (Decrypt $\leqslant_T$ LCDH) [1] Ciphertext (u_1, u_2, e, v) Plaintext m Query oracle $\mathcal{O}$ on (P_1, h, u_1) to obtain G $\tilde{m} \leftarrow e \cdot G^{-1} \pmod p$ $\tilde{m}$

The correctness follows directly from the structure of the ciphertext. Observe that

$$\tilde{m} = eG^{-1} = \frac{Gm}{V_{kz}(P_1, 1)} = \frac{V_{kz}(P_1, 1) \cdot m}{V_{kz}(P_1, 1)} = m,$$

which shows that the original plaintext is recovered. Hence, if the LCDH problem can be solved efficiently, then decryption can also be performed efficiently.

Conversely, assume that there exists an oracle $\mathcal{O}$ that decrypts arbitrary ciphertexts of the scheme under a supplied public key. An algorithm is constructed that uses this oracle to solve the LCDH problem. Given an LCDH instance (P_1, X, Y), where $X = V_x(P_1, 1)$ and $Y = V_y(P_1, 1)$ for unknown x and y, the goal is to compute $V_{xy}(P_1, 1)$.

[H] Solving LCDH using Decryption Oracle (LCDH $\leqslant_T$ Decrypt) [1] LCDH instance (P_1, X, Y) where $X = V_x(P_1, 1)$ and $Y = V_y(P_1, 1)$ LCDH value $V_{xy}(P_1, 1)$ Choose P_2 and random $x_1, x_2, y_1, y_2 \in \mathbb{F}_{p^2}^\star$ $c \leftarrow V_{x_1}(P_1, 1) \cdot V_{x_2}(P_2, 1)$ $d \leftarrow V_{y_1}(P_1, 1) \cdot V_{y_2}(P_2, 1)$ $h \leftarrow X$ Form public key (P_1, P_2, c, d, h) $u_1 \leftarrow Y$; choose random $e \in \mathbb{F}_p^*$; set u_2 and v arbitrarily Query oracle $\mathcal{O}$ on (u_1, u_2, e, v) under this public key to obtain m' $G \leftarrow e \cdot (m')^{-1} \pmod p$ G

The public key embeds the first component of the LCDH instance by setting $h = X = V_x(P_1, 1)$, so the implicit secret exponent is x. Since the public key components c and d are generated honestly, the constructed public key is distributed identically to a real key. The decryption procedure of the scheme computes $m' \equiv e/V_x(u_1, 1) \pmod p$. Since $u_1 = Y = V_y(P_1, 1)$, the composition rule for $Q = 1$ gives

$$V_x(u_1, 1) = V_x\big(V_y(P_1, 1), 1\big) = V_{xy}(P_1, 1).$$

Therefore,

$$G \equiv e \cdot (m')^{-1} \equiv V_{xy}(P_1, 1) \pmod p,$$

which is exactly the solution to the LCDH instance. No component of the constructed ciphertext depends on the value $V_{xy}(P_1, 1)$: the component e is chosen freely, and the components u_2 and v are not used by the decryption procedure of Section 2.2. Thus, an efficient decryption oracle yields an efficient algorithm for solving the LCDH problem.

The above reductions establish a tight relationship between the decryption problem and the LCDH problem. In particular, any adversary capable of breaking the one-wayness of the encryption scheme under passive attack can be used to construct an algorithm that solves the LCDH problem with non-negligible probability. This contradicts the assumed hardness of the LCDH problem. Therefore, the cryptosystem is secure under one-way encryption against passive attacks. □

The following theorem is adapted from [11]. The original proof is revisited and a corrected version is presented for completeness.

Theorem 3. *If the hash function H is chosen from a universal one-way family and the Lucas-Decisional Diffie–Hellman (LDDH) problem is hard, then the cryptosystem is secure under IND-CCA.*

Proof. Let the adversary submit 1^k to the simulator. The simulator is then given an LDDH instance (P_1, P_2, U_1, U_2) drawn from either D or $\tilde{D}$ as in Definition 4. It generates the private keys by choosing random values $(x_1, x_2, y_1, y_2, z_1, z_2) \in \mathbb{F}_{p^2}^\star$ and constructs the public key components as

$$c = V_{x_1}(P_1, 1) \cdot V_{x_2}(P_2, 1), \qquad d = V_{y_1}(P_1, 1) \cdot V_{y_2}(P_2, 1), \qquad h = V_{z_1}(P_1, 1) \cdot V_{z_2}(P_2, 1).$$

The simulator then gives the public key $pk = (P_1, P_2, c, d, h)$ to the adversary.

After receiving the public key, the adversary submits two messages m_0 and m_1. The simulator selects a random bit $b \in \{0, 1\}$ and constructs the challenge ciphertext by embedding the given LDDH instance. Specifically, it sets $u_1 = U_1$ and $u_2 = U_2$, and computes

$$e = V_{z_1}(U_1, 1) \cdot V_{z_2}(U_2, 1) \cdot m_b = V_k(h, 1) \cdot m_b.$$

It then computes $\alpha = H(u_1, u_2, e)$ and

$$v = V_k(c, 1) \cdot V_{k\alpha}(d, 1),$$

and returns the challenge ciphertext (u_1, u_2, e, v) to the adversary.

The simulator answers decryption queries as follows. For any ciphertext (u_1', u_2', e', v') distinct from the challenge ciphertext, it first checks whether $\log_{P_1} u_1' = \log_{P_2} u_2'$. If this condition does not hold, the ciphertext is rejected. Otherwise, there exists a value k' such that

$$u_1' = V_{k'}(P_1, 1), \qquad u_2' = V_{k'}(P_2, 1),$$

and the simulator returns the plaintext

$$m = \frac{e'}{V_{z_1}(u_1', 1) \cdot V_{z_2}(u_2', 1)}.$$

If the given instance is drawn from the distribution D, then it holds that $\log_{P_1} u_1 = \log_{P_2} u_2$, and the constructed ciphertext is a valid encryption of m_b under the public key. On the other hand, if the instance is drawn from the random distribution $\tilde{D}$, then with overwhelming probability $\log_{P_1} u_1 \neq \log_{P_2} u_2$, and the resulting ciphertext is invalid.

The following Lemma 1 and Lemma 2 will continue to prove Theorem 3. □

Lemma 1. *When simulator input comes from D, encryption and calls to the decryption oracles are indistinguishable to the adversary from a real adaptive chosen ciphertext attack.*

Proof. The input comes from D, so it holds that

$$u_1 = V_k(P_1, 1), \qquad u_2 = V_k(P_2, 1)$$

where k is uniformly random. Since u_1, u_2 are distributed exactly as in a real encryption, then

$$e = V_k(h, 1) \cdot m_b$$

makes e indistinguishable as well. The value v is computed as

$$v = V_k(c, 1) \cdot V_{k\alpha}(d, 1)$$

where $\alpha = H(u_1, u_2, e)$. This matches the real encryption algorithm, so the challenge ciphertext is indistinguishable from a real one.

Now consider decryption oracle queries (u_1', u_2', e', v') submitted by the adversary. Following the simulator's specification:

- If $\log_{P_1} u_1' \neq \log_{P_2} u_2'$, the simulator rejects the ciphertext.
- Otherwise, there exists k' such that $u_1' = V_{k'}(P_1, 1)$ and $u_2' = V_{k'}(P_2, 1)$, and the simulator outputs

$$\frac{e}{V_{z_1}(u_1, 1) \cdot V_{z_2}(u_2, 1)} = \frac{e}{V_z(V_k(P_1, 1), 1)}.$$

Lemma 1 follows from the following claim: □

Claim. *Decryption oracle rejects all invalid ciphertext except with negligible probability.*

Proof. Consider an invalid ciphertext (u_1', u_2', e', v') submitted to the decryption oracle, where

$$\log_{P_1} u_1' = r_1, \qquad \log_{P_2} u_2' = r_2$$

with $r_1 \neq r_2$. For the ciphertext to be accepted, it must pass the verification test involving v'.

Let $w = \log_{P_1} P_2$. Then it can be written that

$$\log_{P_1}(c) = x_1 + wx_2 \tag{1}$$
$$\log_{P_1}(d) = y_1 + wy_2 \tag{2}$$

The values x_1, x_2, y_1, y_2 are chosen uniformly at random and are unknown to Adversary. The verification equation for a ciphertext (u_1', u_2', e', v') with $\alpha' = H(u_1', u_2', e')$ requires

$$v' = V_{k'}(c, 1) \cdot V_{k'\alpha'}(d, 1)$$

for the unique k' such that

$$u_1' = V_{k'}(P_1, 1), \qquad u_2' = V_{k'}(P_2, 1).$$

However, since $r_1 \neq r_2$, no such k' exists.

For an invalid ciphertext to be accepted, the adversary would need to forge a v' that satisfies the verification equation without knowing the private keys. Given that x_1, x_2, y_1, y_2 are random and the adversary's information is limited to the linear relations (1) and (2), the probability that a randomly chosen point in the 2-dimensional space of possible (x_1, x_2) or (y_1, y_2) lies on any specific line is negligible. Therefore, the decryption oracle rejects all invalid ciphertexts except with negligible probability. □

Lemma 2. *When simulator input comes from $\tilde{D}$, the encrypted message will be hidden and thus cannot be distinguished by Adversary.*

Proof. When the input comes from $\tilde{D}$, it holds that

$$u_1 = V_{k_1}(P_1, 1), \qquad u_2 = V_{k_2}(P_2, 1)$$

with $k_1 \neq k_2$ (except with negligible probability). Let $w = \log_{P_1} P_2$, so

$$u_2 = V_{k_2 w}(P_1, 1).$$

Lemma 2 follows from the following two claims. □

Claim. *If the decryption oracle rejects all invalid ciphertexts, then the encrypted message is perfectly hidden from the adversary.*

Proof. Consider the point $Q = (z_1, z_2) \in \mathbb{F}_{p^2}^2$. Before any decryption queries, the adversary only access

$$h = V_{z_1}(P_1, 1) \cdot V_{z_2}(P_2, 1).$$

Taking $\log_{P_1}$ of both sides,

$$\log_{P_1}(h) = z_1 + wz_2. \tag{3}$$

Thus, Q is a random point on the line defined by equation (3).

Now the adversary can make decryption queries (u_1', u_2', e', v') to gain information about

$$u_1'^{z_1} u_2'^{z_2} = V_{k_1' z_1}(P_1, 1) \cdot V_{k_2' z_2}(P_2, 1).$$

The decryption oracle only responds when the ciphertext is valid and Adversary can obtain information if

$$\log_{P_1} u_1' = \log_{P_2} u_2' = r_1' = r_2'.$$

In that case:

$$V_{k_1' z_1}(P_1, 1) \cdot V_{k_2' z_2}(P_2, 1) = V_k(V_z(P_1, 1), 1)$$

This yields only the linearly dependent relation:

$$r' \log_{P_1}(h) = r' z_1 + w r' z_2$$

giving no further information about Q since this is just a scalar multiple of Equation (3).

Now consider the challenge ciphertext (u_1, u_2, e, v), the information about the message m is contained only within $e = \varepsilon \cdot m_b$ where $\varepsilon = V_{z_1}(u_1, 1) \cdot V_{z_2}(u_2, 1)$.

This yields the constraint on ε:

$$\log_{P_1}(\varepsilon) = r_1 z_1 + w r_2 z_2 \tag{4}$$

Equation (3) and (4) are linearly independent. Therefore, from the adversary's point of view, e is uniformly distributed regardless of m_b. Hence the message is perfectly hidden, and e acts as a one-time pad. □

Claim. *The decryption oracle rejects all invalid ciphertexts except with negligible probability.*

Proof. Let $\mathcal{P} = (x_1, x_2, y_1, y_2) \in \mathbb{F}_{p^2}^4$ where $\mathcal{P}$ lies on the line L which is the intersection of Equation (1) and (2).

Consider an invalid ciphertext query (u_1', u_2', e', v') with

$$r_1 = \log_{P_1} u_1', \qquad r_2 = \log_{P_2} u_2',$$

and $r_1 \neq r_2$. Let $\alpha' = H(u_1', u_2', e')$.

Three cases are considered.

Case 1: $(u_1', u_2', e') = (u_1, u_2, e)$. The decryption oracle rejects if $v' \neq v$.

Case 2: $(u_1', u_2', e') \neq (u_1, u_2, e)$ and $\alpha' \neq \alpha$. For the ciphertext to be accepted, $\mathcal{P}$ must lies on Equation (1) and (2), but these equations are linearly independent, and the hash function, H intersects L at a single point. Since $\mathcal{P}$ is a random on L, the probability that it lies exactly on this intersection is $1/|\mathbb{F}_{p^2}|$, which is negligible. Therefore, the decryption oracle will reject except with negligible probability that this intersection lies on $\mathcal{P}$.

Case 3: $(u_1', u_2', e') \neq (u_1, u_2, e)$ and $\alpha' = \alpha$. Since H is chosen from a universal one-way family, the probability that the adversary finds a distinct input $(u_1', u_2', e') \neq (u_1, u_2, e)$ with the same hash value $\alpha' = \alpha$ is negligible.

In all cases, the probability that an invalid ciphertext is accepted is negligible. Therefore, the decryption oracle rejects all invalid ciphertexts except with negligible probability.

Since both claims apply, the message in the challenge ciphertext is perfectly hidden when the input comes from $\tilde{D}$. This completes the proof of Lemma 2. □

Conclusion

This work revisits the public key cryptosystem analogous to LUCELG and Cramer-Shoup in order to address and rectify formal security gaps present in its original 2009 proposal. While the initial design successfully integrates the alternative mathematical foundation of Lucas functions with the hash-based verification mechanism of the Cramer-Shoup scheme, it does not provide rigorous formal security proofs consistent with modern cryptographic standards.

To address this limitation, a comprehensive and formal security analysis is presented. The underlying Lucas-based hardness assumptions are explicitly defined, including the Single Lucas, Double Lucas, LCDH, and LDDH problems. Formal reductions are then established to evaluate the scheme under passive attack models, including total break security and one-way encryption under passive attacks. In addition, the IND-CCA security proof is revised by constructing a rigorous simulator-based reduction that aligns with the algebraic structure of the cryptosystem.

The analysis formally establishes that the hybrid cryptosystem achieves strong provable security guarantees against both passive attacks and adaptive chosen ciphertext attacks (CCA2). By strengthening its theoretical foundations, the LUCELG and Cramer-Shoup

analogous cryptosystem is demonstrated to achieve provable security guarantees comparable to those of the classical Cramer-Shoup scheme, provided that the parameters are selected so that the discrete logarithm problem in $\mathbb{F}_{p^2}$ remains infeasible [1]. The scheme therefore constitutes a structurally distinct and computationally efficient alternative within the discrete logarithm setting, rather than a scheme based on a strictly harder problem.

References

[1] D. Bleichenbacher, W. Bosma, and A. K. Lenstra. Some remarks on lucas-based cryptosystems. In *Advances in Cryptology — CRYPTO '95, Lecture Notes in Computer Science*, volume 963, pages 386–396, 1995.

[2] R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Proceedings of CRYPTO'98, LNCS 1462*, pages 13–25, 1998.

[3] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[4] T. ElGamal. A public-key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31:469–472, 1985.

[5] B. Ibrahimpašić. A cryptanalytic attack on the luc cryptosystem using continued fractions. *Mathematical Communications*, 14(1):103–118, 2009.

[6] P. Lafourcade, L. Robert, and D. Sow. Fast cramer-shoup cryptosystem. In *Proceedings of the 18th International Conference on Security and Cryptography (SECRYPT 2021)*, 2021.

[7] C. S. Laih, F. K. Tu, and W. C. Tai. Remarks on the luc public key system. *Electronics Letters*, 30:123–124, 1994.

[8] C. S. Laih, F. K. Tu, and W. C. Tai. On the security of the lucas function. *Information Processing Letters*, 53:243–247, 1995.

[9] J. Liu, Y. Yu, B. Yang, J. Jia, and Q. Lai. Cryptanalysis of cramer-shoup like cryptosystems based on index exchangeable family. *International Journal of Foundations of Computer Science*, 32(1):73–91, 2021.

[10] F. E. A. Lucas. Théorie des fonctions numériques simplement périodiques. *American Journal of Mathematics*, 1:184–240, 289–321, 1878.

[11] N. Muslim and M. R. Md. Said. A new cryptosystem analogous to lucelg and cramer–shoup. *International Journal of Cryptology Research*, 1(2):191–204, 2009.

[12] R. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[13] I. N. Sarbini, T. J. Wong, M. Othman, and L. F. Koo. The security analysis of gcd attack on lucelg, luc(3)elg and luc(4,6)elg. *AIP Conference Proceedings*, 2895(1):020006, 2024.

[14] P. Smith and M. Lennon. Luc: A new public key system. In *Proceedings of the Ninth International Conference on Information Security, IFIP/Sec*, 1993.

[15] P. J. Smith and C. Skinner. A public-key cryptosystem and a digital signature system based on the lucas function analogue to discrete logarithms. In *Advances in Cryptology — ASIACRYPT '94, Lecture Notes in Computer Science*, volume 917, pages 298–306, 1994.

[16] T. J. Wong, L. F. Koo, A. F. N. Rasedee, F. H. Naning, and M. H. Abdul Sathar. Security analysis on luc-type cryptosystems using common modulus attack. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, 29(3), 2023.

[17] T. J. Wong, L. F. Koo, I. N. Sarbini, F. H. Naning, M. Othman, and M. M. Magiman. On the security comparison of luc-type cryptosystems using chosen message attack. *Advances in Mathematics: Scientific Journal*, 9(12), 2020.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 232–244.

Hardware Implementation of CiliPadi Family on FPGA

Muhammad Hafiz Abu Hassan and Muhammad Reza Z'aba

Malaysia Cryptology Technology and Management Centre
c/o Universiti Putra Malaysia 43400, UPM, Serdang, Selangor, Malaysia
{hafizhassan,rezazaba}@upm.edu.my

Abstract. This article describes the performance analysis of the hardware implementation of the CiliPadi family of lightweight authenticated encryption with associated data (AEAD) v1.2 on FPGA platform. An earlier version, dubbed v1.0, was accepted as one of the Round 1 candidates in the US NIST Lightweight Cryptography (LWC) project. CiliPadi is designed based on the Sponge construction which is also used in the SHA-3 hash function. This paper presents an extended hardware implementation of all CiliPadi variants on AMD Xilinx Artix-7 and Kintex-7 FPGA platforms. The design follows the Competition for Authenticated Encryption: Security, Applicability, and Robustness (CAESAR) Hardware Application Programming Interface (API) specification to ensure standardized benchmarking and fair comparison with the NIST LWC standard, Ascon. Performance evaluation is conducted in terms of area utilization and implementation throughput. Experimental results demonstrate that CiliPadi-Mild uses fewer lookup tables (LUTs) and slices than the evaluated Ascon implementation on both FPGA platforms, but achieves lower throughput and throughput per LUT. These results demonstrate an area advantage for CiliPadi-Mild under the evaluated architecture and implementation conditions.

Keywords: Cilipadi · Lightweight Authenticated · FPGA

1 Introduction

The rapid growth of Internet of Things (IoT) devices, embedded systems, and edge computing platforms has increased the demand for cryptographic algorithms suitable for resource-constrained environments. These platforms operate under strict limitations in logic resources, memory, and power consumption [18]. Consequently, lightweight cryptographic algorithms must provide strong security while minimizing implementation cost.

Symmetric cryptography has a long tradition of organized competitions aimed at identifying secure and efficient cryptographic primitives. The outcomes of these initiatives have significantly influenced modern information security. For example, Rijndael, which was later standardized as the Advanced Encryption Standard (AES), has become the de facto symmetric encryption standard for protecting data in modern information and communication systems. More recently, the CAESAR competition [3] evaluated authenticated encryption (AE) schemes over a five-year period from 2014 to 2019, attracting more than 45 candidate algorithms competing for inclusion in its final portfolio of practical security solutions. This was followed by the NIST Lightweight Cryptography (LWC) Project [16], launched in 2019, which accepted 56 candidate algorithms. Among the outcomes of these initiatives, Ascon [8] emerged as the leading lightweight AE with associated data (AEAD) algorithm and was later standardised in NIST SP 800-232.

When competing designs exhibit comparable cryptographic strength and no practical security weaknesses, implementation efficiency becomes an important evaluation criterion [1].

Consequently, hardware metrics such as area, throughput, operating frequency, and the implementation cost of side-channel countermeasures are essential for distinguishing candidate algorithms. Reflecting this, most submissions to both the CAESAR competition and the NIST LWC [16] project, including Ascon and Beetle [7], reported preliminary hardware implementation results.

CiliPadi [22] is a family of lightweight AEAD algorithms. Version 1.0 was submitted to the NIST Lightweight Cryptography (LWC) competition in March 2019 and accepted as one of the 56 Round 1 candidates [20]. However, it did not advance to Round 2 because of a length-extension attack identified by Bagheri [19]. To address this vulnerability, CiliPadi v1.2 was introduced in 2021 with a revised padding scheme [22]. In this work, we present an FPGA implementation and hardware performance evaluation of CiliPadi v1.2 and benchmark it against Ascon, the NIST-standardized lightweight cryptography algorithm.

The comparison is conducted using the CAESAR Hardware API, which provides a common benchmarking framework and ensures a fair evaluation of both algorithms. Since the available CAESAR Hardware API implementation of Ascon is based on version 1.2 of the specification [8], this paper adopts that version instead of the standardized specification in NIST SP 800-232. Although the two versions differ slightly, these differences have a negligible impact on the hardware implementation and therefore do not affect the validity of the comparison. Both implementations are synthesized on AMD Xilinx Artix-7 and Kintex-7 FPGAs using identical architectural assumptions and tool settings. In addition to the baseline architecture, this work investigates several implementation optimizations, including ANF-based S-boxes and serial MixColumns realizations, to evaluate the trade-offs among area, maximum operating frequency, and throughput.

This work makes three main contributions. First, it presents CAESAR Hardware API implementations of all four CiliPadi v1.2 variants targeting Artix-7 and Kintex-7 FPGAs. Second, it investigates multiple hardware architectures, including the baseline design, an algebraic normal form (ANF)-based S-box, and a serial MixColumns realization, and evaluates their impact on area, operating frequency, and throughput. Third, it provides a fair comparison between CiliPadi and the legacy Ascon v1.2 CAESAR Hardware API implementation using identical FPGA devices, synthesis tools, and benchmarking methodology.

The remainder of this paper is organized as follows. Section 2 introduces the CiliPadi algorithm. Section 3 describes the CAESAR Hardware API framework used for benchmarking. Section 4 presents the proposed hardware architecture and the implementation optimizations. Finally, Section 5 presents the FPGA implementation results and compares the performance of CiliPadi with Ascon.

2 CiliPadi

CiliPadi is a family of lightweight AE algorithms. CiliPadi is built upon the MonkeyDuplex [4] construction. The Sponge [12] construction is a versatile cryptographic design that has undergone extensive analysis and has been widely adopted in various hash functions and authenticated encryption schemes, including Keccak [5], PHOTON [13], Fides [6], and Ascon [8].

The CiliPadi family consists of four variants: CiliPadi-Mild, CiliPadi-Medium, CiliPadi-Hot, and CiliPadi-ExtraHot. These variants differ in terms of key length, tag length, message block size, and the number of rounds used in the internal and external permutation. The underlying permutation function incorporates an unkeyed two-round instance of the lightweight block cipher LED [14] as the F-function within a Type-II generalized Feistel network (GFN) structure [21]. This design approach is conceptually similar to the Simpira v2 permutation framework proposed by Gueron and Mouha [11].

2.1 Mode of Operation

The CiliPadi $[n; r; a; b]$ mode of operation is based on the MonkeyDuplex construction and depicted in Figure 1. It consists of four phases: initialization, associated data authentication, message encryption/decryption, and finalization. The state length is n bits initialized with the value of the secret key K and nonce N. The bitrate is r bits and the capacity is $c = n - r$ bits. The permutation for the initialization and finalization phases has a rounds while the permutation for the associated data and message encryption/decryption phases has b rounds where $a >$.

All the parameter of CiliPadi Family are listed in Table 1 according to increasing level of security. They are CiliPadi-Mild, CiliPadi-Medium, CiliPadi- Hot and CiliPadi-ExtraHot. The lengths stated in the table are all in bits.

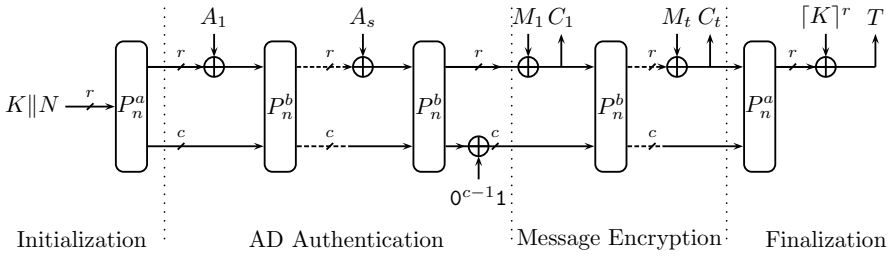


Figure 1: CiliPadi mode of operation [22]

Table 1: CiliPadi parameters where the primary member is CiliPadi-Mild [22].

CiliPadi	Algorithm $[n, r, a, b]$	Key	Nonce	Tag	Block	P_n^a	P_n^b
Mild	$[256, 64, 18, 16]$	128	128	64	64	18	16
Medium	$[256, 96, 20, 18]$	128	128	96	96	20	18
Hot	$[384, 96, 18, 16]$	256	128	96	96	18	16
ExtraHot	$[384, 128, 20, 18]$	256	128	128	128	20	18

Formally, the CiliPadi family of AEAD accepts a k-bit secret key K and a 128-bit nonce N. These values become the initial value of the n-bit internal state $S = K \parallel N$. The state is then updated by the permutation P_n^a. If the sr-bit associated data $A = A_1 \parallel \cdots \parallel A_s$ is non-empty it will be subsequently processed, along with the internal state, by the associated data authentication phase. Message Encryption phase takes the padded message $M = M_1 \parallel \cdots \parallel M_t$ and outputs the ciphertext $C = C_1 \parallel \cdots \parallel C_t$ and tag T where $M_i = C_i = T = r$ bits. Decryption takes the ciphertext C and tag T and outputs the original message M if and only if C is authentic, else it outputs $\perp$.

The *Initialization* phase takes the secret key K (128 bits or 256 bits) and the public nonce N (128 bits). This nonce has to be fresh for every encryption and must not be used twice. If the nonce is used twice or multiple times, then the confidentiality can be compromised. After the *Initialization* phase the optional associated data A_i is processed. Associated data is information, which does not need to be confidential, but must not be altered by an attacker. Each block A_i is XORed to the secret state. If there is no associated data to process, the whole step can be excluded.

In the *Encryption* phase, each plaintext block M_i is XORed with the secret state to produce one ciphertext block C_i.For each message block , CiliPadi-Mild and CiliPadi-Hot

apply 16 permutation rounds to 64-bit and 96-bit blocks, respectively. CiliPadi-Medium and CiliPadi-ExtraHot apply 18 permutation rounds to 96-bit and 128-bit blocks.

After ciphertext generation, the *Finalization* phase produces an r-bit authentication tag T, corresponding to 64, 96, 96, and 128 bits for CiliPadi-Mild, CiliPadi-Medium, CiliPadi-Hot, and CiliPadi-ExtraHot, respectively. With the help of this tag, modifications of the ciphertext and the associated data can be detected during decryption (validation). Decryption is very similar to encryption. Just the part, where the ciphertext is processed instead of the plaintext differs slightly. Thus, no inverse of the permutation is needed for decryption. So, both encryption and decryption can be implemented with just a slight overhead compared to encryption only.

2.2 Permutation

The permutation function P makes use of an unkeyed two-round[1] of the lightweight block cipher LED [14] as the round and line-dependent F-function in a Type-II generalized Feistel network (GFN) [21]. We refer to a Type-II GFN that accepts d input sub-blocks as a d-line Type-II GFN. For CiliPadi, d is an even number and each line is of length n/d bits. There are $d/2$ F-functions in each round that accept input from odd-numbered lines. Let $X_1 \parallel \cdots \parallel X_d$ denote the input lines. They are updated by the F-function in the i-th round as follows:

$$\begin{aligned} X_j &\leftarrow X_j, && \text{for } j = 1, 3, \ldots, d-1, \\ X_j &\leftarrow X_j \oplus F^i_{j/2}(X_{j-1}), && \text{for } j = 2, 4, \ldots, d. \end{aligned} \tag{1}$$

As previously described, the round and line-dependent F function corresponds to an unkeyed two-round version of the LED block cipher. In this construction, all operations are preserved in the final (i.e., second) LED round. The input to the LED round function is a 64-bit value divided into sixteen 4-bit cells. Let $x = x_1 \| x_2 \| \ldots \| x_{16}$ denote the input state. This state can be represented as a 4×4 matrix arranged in row-major order. Each LED round consists of four sequential transformations applied to a 64-bit state: *AddConstants* (`AC`), *SubCells* (`SC`), *ShiftRows* (`SR`), and *MixColumnsSerial* (`MCS`), as illustrated in Figure 2.

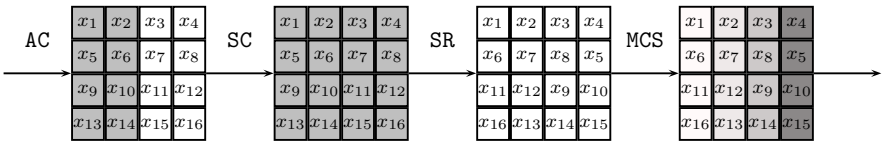


Figure 2: A single round of LED.

The transformation `AC` XORs the state with a round-dependent constant, while `SC` applies the 4-bit S-box independently to each state cell. Let $x_{i,j}$ and $y_{i,j}$ denote the j-th input and output bits of the S-box, respectively. The algebraic normal form (ANF) of the

[1]The 2-round LED is used as the F-function.

S-box is given by

$$
\begin{aligned}
y_{i,0} &= x_{i,0} \oplus x_{i,2} \oplus (x_{i,1} \cdot x_{i,2}) \oplus x_{i,3} \\
y_{i,1} &= x_{i,1} \oplus (x_{i,0} \cdot x_{i,1} \cdot x_{i,2}) \oplus x_{i,3} \oplus (x_{i,1} \cdot x_{i,3}) \oplus (x_{i,0} \cdot x_{i,1} \cdot x_{i,3}) \oplus (x_{i,2} \cdot x_{i,3}) \oplus \\
&\quad (x_{i,0} \cdot x_{i,2} \cdot x_{i,3}) \\
y_{i,2} &= 1 \oplus (x_{i,0} \cdot x_{i,1}) \oplus x_{i,2} \oplus x_{i,3} \oplus (x_{i,0} \cdot x_{i,3}) \oplus (x_{i,1} \cdot x_{i,3}) \oplus (x_{i,0} \cdot x_{i,1} \cdot x_{i,3}) \oplus \\
&\quad (x_{i,0} \cdot x_{i,2} \cdot x_{i,3}) \\
y_{i,3} &= 1 \oplus x_{i,0} \oplus x_{i,1} \oplus (x_{i,1} \cdot x_{i,2}) \oplus (x_{i,0} \cdot x_{i,1} \cdot x_{i,2}) \oplus x_{i,3} \oplus (x_{i,0} \cdot x_{i,1} \cdot x_{i,3}) \oplus \\
&\quad (x_{i,0} \cdot x_{i,2} \cdot x_{i,3})
\end{aligned}
$$

The transformation `SR` cyclically rotates the rows of the state to the left, as illustrated in Figure 2. The `MCS` transformation performs a 4×4 matrix multiplication on each state column. The corresponding MDS matrix is

$$
M = (A)^4 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 4 & 1 & 2 & 2 \end{pmatrix}^4 = \begin{pmatrix} 4 & 1 & 2 & 2 \\ 8 & 6 & 5 & 6 \\ B & E & A & 9 \\ 2 & 2 & F & B \end{pmatrix}.
$$

3 CAESAR Hardware API

The proposed implementation follows the CAESAR Hardware API, which defines a standardized framework for implementing authenticated encryption algorithms in hardware. The API specifies several key components, including the interface structure, communication protocol, minimum compliance requirements, and timing characteristics of the cryptographic core. These elements are designed with two primary objectives that is ensuring compatibility among implementations of the same algorithm developed by different designers and enabling fair and consistent benchmarking of authenticated encryption algorithms in hardware platforms.

The interface separates data streams into three main channels: Public Data Input (PDI), Secret Data Input (SDI), and Data Output (DO). The PDI interface is used for transmitting public information such as plaintext, ciphertext, nonce, and associated data, while the SDI interface is dedicated to secret inputs such as the encryption key. The DO interface provides the resulting ciphertext, plaintext, or authentication tag. Communication between the host and the cryptographic core follows a word-oriented streaming protocol with handshake signals to ensure reliable data transfer. This standardized interface enables fair benchmarking and compatibility among hardware implementations of authenticated encryption algorithms.

The proposed hardware implementation targets a high-speed architecture by utilizing the maximum bus width permitted by the CAESAR hardware API parameters, allowing higher data throughput during encryption and decryption operations. The proposed architecture follows the high-speed, non-pipelined implementation model defined by the CAESAR hardware API specification. The top-level module, referred to as the AEAD unit, consists of four main components: the PreProcessor, PostProcessor, Command FIFO (CMD FIFO), and CipherCore, as illustrated in Figure 3.

The PreProcessor, PostProcessor, and CMD FIFO modules provide generic functionality for data formatting, command handling, and communication with the external interface. These modules are designed to be algorithm-independent and are provided as part of the CAESAR development package. As a result, designers implementing a specific authenticated encryption algorithm do not need to modify their internal structures.

The CipherCore module contains the algorithm-specific cryptographic operations. Therefore, the primary focus of the implementation lies in the design of the CipherCore,

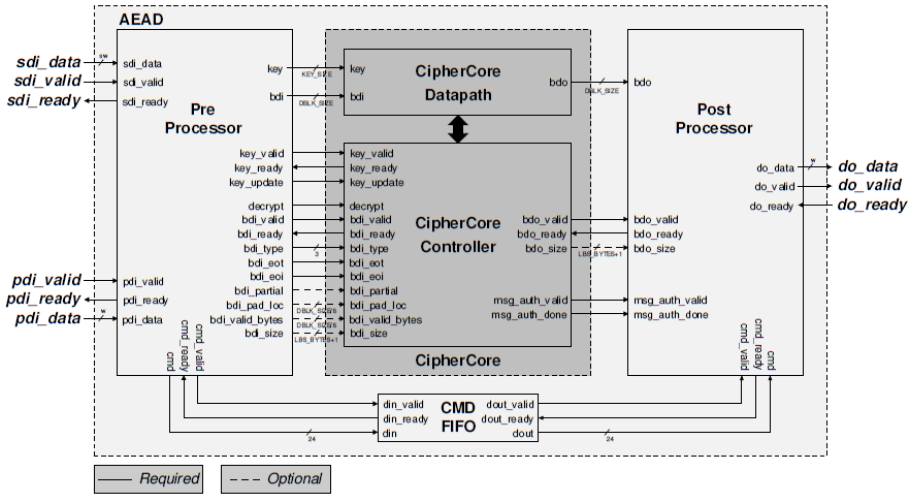


Figure 3: Top-level block diagram of a high-speed architecture of an authenticated cipher core, AEAD. [15]

which can be further divided into a datapath and a control unit depending on the architecture of the implemented cipher.

4 CiliPadi Architecture

Although the CiliPadi round function can support unrolled or multi-round-per-cycle implementations, this work adopts an iterative, non-pipelined architecture. A single instance of the permutation datapath is reused across rounds to limit hardware area and provide a consistent baseline for comparison.

The datapath architecture adopts a basic iterative design, where the round function is executed repeatedly using a single hardware instance of the permutation components. This approach minimizes hardware resources by reusing the same functional units across multiple rounds. The use of an iterative architecture also aligns with the objective of fair benchmarking, as it provides a balanced trade-off between area, throughput, and implementation complexity. By avoiding aggressive optimizations such as full loop unrolling or deep pipelining, the proposed design enables a more consistent comparison with other authenticated encryption implementations following the same benchmarking methodology. The datapath of CiliPadi-Mild, as illustrated in Figure 4, mainly consists of the Iterative (Rolled) Architecture (64-bit state registers, 16 parallel S-boxes, and the MixColumnsSerial module).

For CiliPadi-Medium, the datapath architecture remains similar to that of CiliPadi-Mild, since both variants utilize the same state size. In contrast, the CiliPadi-Hot and CiliPadi-ExtraHot variants require modifications to the datapath architecture to support a larger state size. Consequently, the larger state size introduces an additional F-function in the datapath. This is because the number of F-functions is determined by both the state size and the number of lines in the GFN structure.

The operation of the proposed architecture is controlled by a finite state machine

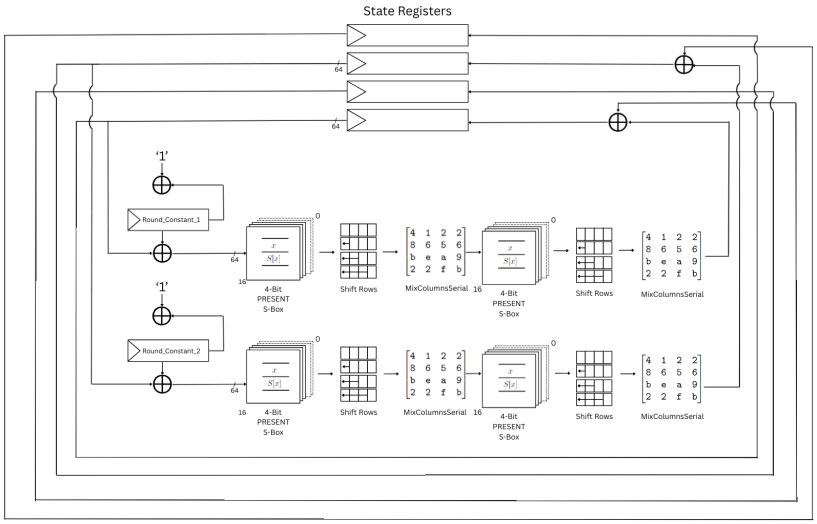


Figure 4: Datapath architecture of CiliPadi-Mild.

(FSM) implemented in the CipherCore. The FSM coordinates the different phases of the authenticated encryption process, including key update, nonce loading, initialization, data processing, permutation rounds, finalization, and tag processing.

The system starts in the idle state, where it waits for either a key update request or a valid nonce input. When a key update signal is asserted, the FSM transitions to the key update state to load the secret key. Otherwise, upon receiving a valid nonce input, the FSM moves to the nonce loading state, followed by the initialization phase, where the internal permutation is executed for a predefined number of rounds. After initialization, the FSM enters the input processing state, where associated data or plaintext/ciphertext blocks are processed.

Depending on the type of input, the FSM either triggers the permutation state for intermediate processing or proceeds directly to the finalization phase when the last data block or authentication tag is detected. During finalization, additional permutation rounds are executed before entering the tag processing state, where the authentication tag is generated or verified. Finally, the FSM returns to the idle state and waits for the next encryption or decryption request. Figure 5 illustrates the FSM controlling the CipherCore operation.

There are also the ANF-based S-box and the serial MixColumns architecture were implemented and evaluated to investigate the trade-off between hardware area and processing performance. Each design was synthesized using the same target technology, synthesis tool, and optimization settings to ensure a fair comparison. The collected results were then compared to determine the suitability of each approach for high-throughput and resource-constrained applications.

5 Performance Analysis

The performance of the proposed CiliPadi implementation is evaluated in terms of hardware area, maximum operating frequency, and throughput. The design is synthesized and implemented using AMD Xilinx Vivado targeting the xc7a35tcpg236-3 Artix-7 FPGA and xc7k70tfbg484-3 Kintex-7 FPGA. The obtained results are used to assess the efficiency of the proposed architecture.

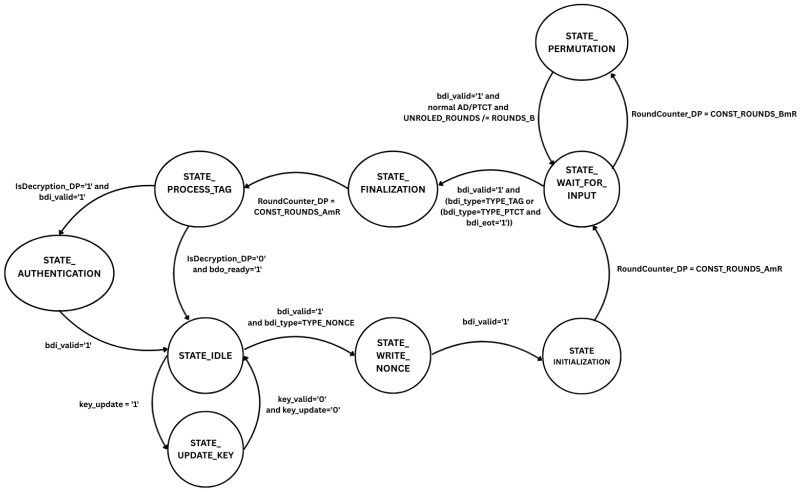


Figure 5: Finite State Machine controlling CiliPadi.

The FPGA implementation results are considered representative of the hardware performance of the proposed architecture. This is because FPGA synthesis maps the HDL design directly onto physical hardware resources such as lookup tables (LUT), registers, and routing networks. Therefore, the reported area utilization and timing characteristics provide a realistic estimation of the implementation cost and achievable performance. All designs were synthesized and implemented using AMD Xilinx Vivado 2024.1. A clock constraint of 10 ns was applied.

Table 2 presents a comparison with previous implementations of CiliPadi and Ascon, where the designs were implemented on a different FPGA platform (Virtex-6) and synthesized using Xilinx ISE tools. This comparison is provided as a reference; however, differences in FPGA platforms and synthesis tools can affect performance metrics such as frequency, area, and throughput. In addition, the implementations in Table 2 do not follow the CAESAR Hardware API, which may lead to differences in design structure and evaluation methodology. Therefore, the results should be interpreted with caution.

A more consistent and fair comparison is presented in Tables 3 and 4, where both CiliPadi and Ascon are evaluated using the same CAESAR Hardware API on identical FPGA platforms. The Ascon hardware results were obtained by synthesizing and implementing the VHDL reference design for the CAESAR Hardware API developed by Hannes Groß [10]. Comparing CiliPadi with Ascon is particularly meaningful because both algorithms adopt the sponge construction while employing different permutation designs. This enables the evaluation to highlight the impact of the underlying permutation architecture on hardware performance under a common implementation framework.

Furthermore, different hardware implementation styles were evaluated and compare for selected modules, such as table-based S-boxes, ANF-based S-boxes, a direct matrix realization of the LED MixColumns operation, and a compact serial implementation. The description of these implementation are describe in [14]. Tables 5 and 6 respectively shows the comparison of the different implementation styles for CiliPadi-Mild that is ANF-based S-box and Serial MixColumns.

Table 2: FPGA implementation results from previous work

Scheme	Platform	LUT	Slices	Freq. (MHz)	Mbps	Mbps/ LUT
CiliPadi-Mild [22]	Virtex 6	1052	303	639.95	1138	1.08
CiliPadi-Medium [22]	Virtex 6	747	363	620.00	1353	1.81
CiliPadi-Hot [22]	Virtex 6	1268	373	631.00	1685	1.32
CiliPadi-ExtraHot [22]	Virtex 6	1332	392	605.00	1760	1.32
Ascon-128 [9]	Virtex 6	1274	451	341.10	3118	2.45
Ascon-128a [9]	Virtex 6	1587	547	358.60	5099	3.21

Table 3: FPGA implementation results for Artix-7 FPGA

Scheme	LUT	Slices	Freq. (MHz)	Mbps	Mbps/ LUT
CiliPadi-Mild	1274	391	139.14	556.56	0.44
CiliPadi-Medium	1453	429	134.17	715.57	0.49
CiliPadi-Hot	1800	514	122.53	735.18	0.41
CiliPadi-ExtraHot	1733	534	137.85	980.26	0.57
Ascon-128	1532	444	161.57	1723.41	1.35
Ascon-128a	1587	492	133.38	2134.08	1.34

Table 4: FPGA implementation results for Kintex-7

Scheme	LUT	Slices	Freq. (MHz)	Mbps	Mbps/ LUT
CiliPadi-Mild	1274	399	197.91	791.60	0.62
CiliPadi-Medium	1453	439	153.65	819.46	0.56
CiliPadi-Hot	1800	538	177.77	1066.62	0.59
CiliPadi-ExtraHot	1733	527	156.98	1116.30	0.64
Ascon-128	1530	449	210.83	2248.85	1.47
Ascon-128a	1726	505	196.03	3136.48	1.82

Table 5: ANF-based S-box implementation for Artix-7 FPGA

Scheme	LUT	Slices	Freq. (MHz)	Mbps	Mbps/ LUT
CiliPadi-Mild	1275	389	131.34	525.32	0.41
CiliPadi-Medium	1280	384	141.58	755.09	0.59
CiliPadi-Hot	1810	555	132.81	796.86	0.44
CiliPadi-ExtraHot	1710	500	132.29	940.74	0.55

Table 6: Serial MixColumns implementation for Artix-7

Scheme	LUT	Slices	Freq. (MHz)	Mbps	Mbps/ LUT
CiliPadi-Mild	1249	389	131.03	524.11	0.42
CiliPadi-Medium	1299	387	125.31	668.32	0.51
CiliPadi-Hot	1675	517	118.20	709.22	0.42
CiliPadi-ExtraHot	1699	519	125.11	889.66	0.52

From Table 3, on Artix-7 slice usage increases from 391 slices for CiliPadi-Mild to 534 slices for CiliPadi-ExtraHot. LUT usage generally increases for the larger-state variants but is not monotonic, since CiliPadi-Hot uses 1800 LUTs compared with 1733 LUTs for CiliPadi-ExtraHot. The LUT usage increases from 1274 LUTs for CiliPadi-Mild to 1733 LUTs for CiliPadi-ExtraHot, while the number of slices increases from 391 to 534 slices. This trend is expected since the more advanced variants incorporate additional cryptographic operations that enhance security strength but require additional hardware resources.

Despite the increase in area, the throughput also improves significantly. The throughput rises from 556.56 Mbps for CiliPadi-Mild to 980.26 Mbps for CiliPadi-ExtraHot, demonstrating that the architecture scales well with increased complexity. However, the frequency does not strictly increase with the variant level; for instance, CiliPadi-Hot operates at a slightly lower frequency compared to other variants. This behavior can be attributed to the longer combinational paths. The reason of the throughput improvement is due to the increase in the block size, which allows more data to be processed per round.

The efficiency metric Mbps/LUT indicates that the CiliPadi-ExtraHot variant achieves the highest hardware efficiency (0.57 Mbps/LUT) among the proposed variants on Artix-7, highlighting the favorable balance between throughput and area.

A comparison between Table 3 and Table 4 shows that implementations on the Kintex-7 FPGA consistently achieve higher operating frequencies and throughput compared to the Artix-7 platform. For example, the frequency of CiliPadi-Mild increases from 139.14 MHz on Artix-7 to 197.91 MHz on Kintex-7, resulting in a throughput improvement from 556.56 Mbps to 791.60 Mbps. This improvement can be attributed to the more advanced architecture and higher performance characteristics of the Kintex-7 FPGA, which offers better timing performance and routing resources.

Table 3 and Table 4 also compare the proposed implementations with Ascon-128 and Ascon-128a. Note that the NIST-standardized Ascon-AEAD128 is based on Ascon-128a with some minor differences. The results show that Ascon128 achieves higher throughput values, reaching 1732.41 Mbps on Artix-7 and 2248.85 Mbps on Kintex-7. However, this higher throughput is accompanied by a larger hardware footprint and a different architectural design philosophy.

While Ascon128 exhibits higher raw throughput and efficiency values (up to 1.12 Mbps/LUT on Artix-7 and 1.47 Mbps/LUT on Kintex-7), the proposed CiliPadi implementations remain competitive considering their design goals and architectural simplicity. In particular, the CiliPadi designs maintain moderate area requirements while achieving throughput values close to 1 Gbps on Artix-7 devices.

Two alternative module implementations were evaluated independently against the baseline architecture: an ANF-based S-box replacing the baseline S-box, and a serial MixColumns implementation replacing the baseline direct MixColumns realization. Tables 5 and 6 report these two experiments separately.

From Table 5, the ANF-based S-box implementation demonstrates a relatively balanced trade-off between area and performance across all variants. The LUT utilization remains comparable to the original design, with only minor variations, while maintaining stable

operating frequencies in the range of approximately 131–142 MHz. Notably, CiliPadi-Medium achieves the highest efficiency of 0.59 Mbps/LUT, indicating that the ANF representation provides a compact and hardware-friendly realization of the S-box. This is mainly due to the Boolean logic structure of ANF, which maps efficiently onto FPGA LUT resources without requiring memory-based lookups.

In contrast, Table 6 shows the results for the serial MixColumns implementation, where a slight reduction in LUT usage can be observed for most variants, particularly for CiliPadi-Mild and CiliPadi-Hot. The serial MixColumns implementation produces variant-dependent area changes. LUT usage decreases for CiliPadi-Mild, CiliPadi-Hot, and CiliPadi-ExtraHot, but increases for CiliPadi-Medium. Throughput decreases for all four variants. Therefore, the serial implementation is beneficial only when the measured area saving for the selected variant justifies the corresponding throughput reduction.

Comparing both approaches, it can be observed that the ANF-based S-box implementation generally provides better throughput and overall efficiency, while the serial MixColumns design offers modest area savings at the expense of performance. This highlights the classical trade-off between area and speed in hardware design. For applications where higher throughput is required, the ANF-based S-box is more suitable. Conversely, for resource-constrained environments where minimizing area is critical, the serial MixColumns approach may be preferred.

These results demonstrate that the proposed architecture is flexible and allows different implementation strategies without significantly affecting the overall hardware efficiency. In particular, the Serial MixColumns approach slightly reduces the LUT count while maintaining comparable throughput, making it a best option for area-constrained FPGA implementations.

Overall, the FPGA implementation results demonstrate that the proposed CiliPadi variants achieve a compact hardware footprint with moderate throughput performance, making them suitable for resource-constrained hardware environments. In particular, the designs maintain relatively low LUT and slice utilization while still achieving throughput close to the gigabit-per-second range on modern FPGA platforms. This indicates that the CiliPadi architecture is well suited for lightweight hardware implementations.

However, recent trends in hardware-oriented cryptographic designs increasingly emphasize high-throughput performance, largely due to the continuous advancement of FPGA and ASIC technologies [17] that enable deeper pipelining, parallelism, and aggressive loop unrolling [2]. As demonstrated by the comparison with Ascon implementations, modern lightweight ciphers often prioritize throughput efficiency to fully exploit available hardware resources.

In this context, future improvements to CiliPadi may focus on revisiting the permutation structure and round configuration. By optimizing the permutation design or reducing the number of rounds while maintaining the required security margin, it may be possible to further improve throughput and hardware efficiency. Such architectural refinements could allow CiliPadi to achieve more competitive performance with state-of-the-art lightweight cryptographic primitives such as Ascon, while preserving its lightweight hardware characteristics.

6 Conclusion

The FPGA implementation results demonstrate that the proposed CiliPadi variants achieve a compact hardware footprint while maintaining moderate throughput performance across different FPGA platforms. The implementations show that CiliPadi can be efficiently realized using a relatively small number of LUTs and slices, confirming the lightweight nature of the design. Furthermore, the evaluation of alternative implementation strategies, such as ANF-based S-boxes and serial MixColumns architectures, indicates that the

proposed architecture provides flexibility in exploring different hardware trade-offs without significantly affecting the overall performance.

Nevertheless, advances in FPGA and ASIC technologies have significantly increased the available hardware resources and achievable operating frequencies, allowing modern cryptographic designs to emphasize throughput-oriented architectures through techniques such as pipelining and parallelism. In light of these developments, future work could investigate enhancements to the CiliPadi permutation structure or optimize the round configuration to further improve throughput while maintaining the required security margin. In addition, integrating side-channel countermeasures, such as masking or threshold implementations, would strengthen resistance against power analysis attacks and improve the security of hardware implementations. Collectively, these research directions would enhance the practical deployment of CiliPadi, enabling it to provide stronger implementation security while remaining competitive with modern lightweight cryptographic primitives.

References

[1] Farzaneh abed, Christian Forler, and Stefan Lucks. General classification of the authenticated encryption schemes for the CAESAR competition. Cryptology ePrint Archive, Paper 2014/792, 2014.

[2] Fares Alharbi, Muhammad Khurram Hameed, Anusha Chowdhury, Ayesha Khalid, Anupam Chattopadhyay, and Ibrahim Tariq Javed. Analysis of area-efficiency vs. unrolling for estream hardware portfolio stream ciphers. *Electronics*, 9(11):1935, 2020.

[3] Daniel J. Bernstein. Caesar: Competition for authenticated encryption: Security, applicability, and robustness. https://competitions.cr.yp.to/caesar.html, 2014. Accessed: 2026-03-02.

[4] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Permutation-Based Encryption, Authentication and Authenticated Encryption. DIAC – Directions in Authenticated Ciphers, July 2012.

[5] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak sponge function family main document, 2009.

[6] Begül Bilgin, Andrey Bogdanov, Miroslav Knežević, Florian Mendel, and Qinju Wang. Fides: Lightweight Authenticated Cipher with Side-Channel Resistance for Constrained Hardware. In Guido Bertoni and Jean-Sébastien Coron, editors, *Cryptographic Hardware and Embedded Systems – CHES 2013*, volume 8086 of *Lecture Notes in Computer Science*, pages 142–158. Springer-Verlag, 2013.

[7] Avik Chakraborti, Nilanjan Datta, Mridul Nandi, and Kan Yasuda. Beetle family of lightweight and secure authenticated encryption ciphers. *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, 2018(2):218–241, 2018.

[8] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2: Lightweight authenticated encryption and hashing. *Journal of Cryptology*, 34(3):33, 2021.

[9] George Mason University. Authenticated Encryption FPGA Ranking. ATHENa: Automated Tools for Hardware EvaluatioN, 2014. https://cryptography.gmu.edu/athenadb/fpga_auth_cipher/table_view.

[10] Hannes Groß. Ascon hardware implementations (caesar hardware api). https://github.com/isec-tugraz/ascon_hardware, 2016. Accessed: 2026.

[11] Shay Gueron and Nicky Mouha. Simpira v2: A Family of Efficient Permutations Using the AES Round Function. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology – ASIACRYPT 2016, Part I*, volume 10031 of *Lecture Notes in Computer Science*, pages 95–125. Springer-Verlag, 2016.

[12] Bertoni Guido, Daemen Joan, P Michaël, and V Gilles. Cryptographic sponge functions, 2011.

[13] Jian Guo, Thomas Peyrin, and Axel Poschmann. The PHOTON Family of Lightweight Hash Functions. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 222–239. Springer, 2011.

[14] Jian Guo, Thomas Peyrin, Axel Poschmann, and Matt Robshaw. The LED Block Cipher. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems – CHES 2011*, volume 6917 of *Lecture Notes in Computer Science*, pages 326–341. Springer-Verlag, 2011.

[15] Ekawat Homsirikamol, William Diehl, Ahmed Ferozpuri, Farnoud Farahmand, and Kris Gaj. Implementer's guide to the caesar hardware api, 2016.

[16] National Institute of Standards and Technology. Lightweight cryptography project. https://csrc.nist.gov/projects/lightweight-cryptography, 2023. Accessed: 2026.

[17] Lesley Shannon, Veronica Cojocaru, Cong Nguyen Dao, and Philip HW Leong. Technology scaling in fpgas: Trends in applications and architectures. In *2015 IEEE 23rd Annual International Symposium on Field-Programmable Custom Computing Machines*, pages 1–8. IEEE, 2015.

[18] Vishal A. Thakor, Mohammad Abdur Razzaque, and Muhammad R. A. Khandaker. Lightweight cryptography algorithms for resource-constrained iot devices: A review, comparison and research opportunities. *IEEE Access*, 9:28177–28193, 2021.

[19] Meltem Sönmez Turan, Kerry A. McKay, Çağdaş Çalık, and Larry Bassham. Status Report on the First Round of the NIST Lightweight Cryptography Standardization Process. NIST IR 8268, October 2019. National Institute of Standards and Technology.

[20] Muhammad Reza Z'aba, Norziana Jamil, Mohd Saufy Rohmad, Hazlin Abdul Rani, and Solahuddin Shamsuddin. The CiliPadi Family of Lightweight Authenticated Encryption. NIST Lightweight Cryptography Project, 2019.

[21] Yuliang Zheng, Tsutomu Matsumoto, and Hideki Imai. On the Construction of Block Ciphers Provably Secure and Not Relying on Any Unproved Hypotheses. In Gilles Brassard, editor, *Advances in Cryptology – CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*, pages 461–480. Springer-Verlag, 1990.

[22] Muhammad Reza Z'aba, Norziana Jamil, Mohd Saufy Rohmad, Hafnidar A Rani, Solahuddin Shamsuddin, and C Malaysia. The cilipadi family of lightweight authenticated encryption, v1. 2. *Malays. J. Math. Sci*, 15:1–23, 2021.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 245–252.

Security Proof for a Blind Signature Scheme under the Strong RSA Assumption

Muhammad Aiman Hakimi Rosdi[1], Tea Boon-Chian*[1,3] and Amir Hamzah Abd Ghafar[1,2]

[1] Malaysia Cryptology Technology and Management Centre, c/o Universiti Putra Malaysia, UPM Serdang, 43400, Selangor, Malaysia
{aimanhakimi,amir_hamzah}@upm.edu.my,teaboonchian@uitm.edu.my

[2] Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, Serdang, 43400, Selangor, Malaysia amir_hamzah@upm.edu.my

[3] Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA (UiTM), 40450 Shah Alam, Selangor, Malaysia teaboonchian@uitm.edu.my

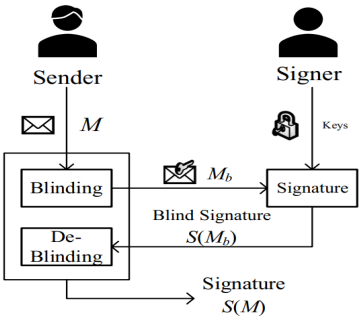
Abstract. Blind signatures combine the authenticity of digital signatures with user privacy. Naji and Abu Hasan introduced a compact signature scheme based on the Strong RSA assumption, along with an interactive blind-signature transformation. However, its security originally relied on informal arguments. In this work, we address this gap by establishing a formal security proof. Specifically, we demonstrate the existential unforgeability under chosen-message attack (EUF-CMA) of the underlying non-interactive signature scheme, grounded in the hardness of the Strong RSA problem within the random oracle model.

Keywords: Blind Signature, Strong RSA, Existential Unforgeability under Chosen Message Attack

1 Introduction

Digital signatures represent a foundational pillar of modern cryptography, providing authenticity and integrity guarantees based on formal unforgeability definitions[8], and are widely recognized as ensuring data integrity, authentication, and non-repudiation in practical systems[15]. In a standard digital signature protocol, a signer utilizes a mathematically generated private key to compute a signature on a specific message[14]. Any verifier possessing the corresponding public key can subsequently confirm that the signature was undeniably generated by the key owner and that the message has not been altered in transit. In modern privacy-preserving applications such as untraceable electronic cash systems or anonymous secure voting protocols this direct binding creates a privacy violation. In these scenarios, a user explicitly requires the signer, a bank or an election authority to authorize a transaction without revealing the underlying content of that transaction to the signer.

To resolve this inherent conflict between authorization and privacy, Chaum[5] introduced the revolutionary concept of the blind signature. A blind signature protocol is an interactive mathematical exchange involving two distinct parties, a sender and a signer. The protocol is conceptually analogous to placing a paper document and a sheet of carbon paper inside a sealed envelope. The signer signs the outside of the envelope and the carbon paper imprints the signature onto the hidden document inside.



[17]

Figure 1: Blind signature process

A complete blind-signature analysis must therefore address two separate adversarial goals, blindness against a malicious signer and unforgeability against a malicious user. For the latter goal, the usual definition is one-more unforgeability. After completing at most q issuing sessions, a user must not be able to output more than q valid signatures on distinct messages[9, 12]. Consequently, EUF-CMA and one-more unforgeability are not interchangeable. EUF-CMA asks for one fresh forgery relative to a message-signing oracle, whereas one-more unforgeability counts successful outputs against completed interactive issuing sessions.

The present work deliberately studies the former, narrower property. More precisely, it considers the non-interactive signature relation induced by the Naji–Abu Hasan construction, for which a purported signature $\sigma = (y, e)$ on a message m is accepted when e has the prescribed form and

$$y^e \equiv xh^{H(m)} \pmod{n}. \tag{1}$$

The EUF-CMA experiment gives the adversary access to a random oracle $\mathcal{O}_H$ and to a signing oracle for this relation. The adversary wins only if it outputs a valid pair (m^*, σ^*) for a message m^* that was not submitted to the signing oracle. The algebraic foundation is the Strong RSA assumption, introduced in work on collision-free accumulators [2]. Given an RSA modulus n of unknown factorization and a random element $z \in \mathbb{Z}_n^*$, the Strong RSA problem asks for any pair (w, E) such that $E > 1$ and $w^E \equiv z \pmod{n}$. In contrast with the ordinary RSA problem, the solver may choose the exponent E after seeing the challenge. Cramer and Shoup used this assumption to construct signatures secure against adaptive chosen-message attacks in the standard model[6]. Their result also illustrates a central requirement of reduction proofs, a simulator must embed the given hard-problem instance into the public key, answer every oracle query without the secret factorization, and extract a solution from the final forgery.

Naji and Hasan proposed a simplified Strong-RSA-based signature relation and a blind transformation of that relation[11]. The original paper presents the algorithms and gives informal discussions of correctness, blindness, and unforgeability, but it does not provide a game-based reduction for the blind transformation. In particular, unforgeability of an underlying ordinary signature scheme does not automatically imply security of an interactive blind version. This motivates a precise separation between the property analyzed here and the stronger security requirements of a complete blind-signature scheme.

2 Related Works

2.1 Unforgeability of ordinary and blind signatures

Goldwasser et al[8] formalized adaptive chosen-message security for ordinary digital signatures . Chaum's RSA-based blind signature introduced interactive signature issuance for privacy-preserving payment systems[5]. The interaction changes the relevant forgery objective, the malicious user does not merely request a signature on a visible message, but controls blinded protocol sessions. Pointcheval and Stern[12] therefore introduced the one-more forgery notion and showed that security of an ordinary signature scheme does not automatically carry over to its blind transformation . Juels developed simulation-based foundations for blind-signature security[9], while Pointcheval and Stern refined the associated reduction techniques and forking arguments[13] .

2.2 Random-oracle proofs for blind signatures

The random oracle model treats the hash function as an ideal, consistently sampled random function that can be queried by all parties[4]. It has been used extensively to obtain security arguments for interactive blind-signature constructions. Bellare and Semanko proved the security of Chaum's RSA blind signature in this model under one-more RSA inversion assumptions [3]. Abe and Okamoto developed provably secure partially blind signatures and supplied formal random-oracle proofs for both blindness and unforgeability [1]. These results reinforce two methodological points relevant here, the oracle simulation must be explicit, and the security notion must match the interactive capability given to the adversary.

2.3 Signatures from the Strong RSA assumption

Barić and Pfitzmann introduced the Strong RSA assumption in their work on collision-free accumulators [2]. Cramer and Shoup then gave an efficient signature scheme with a standard-model proof of adaptive chosen-message security under Strong RSA and collision resistance [6]. Their proof classifies possible forgeries and embeds the challenge so that signing queries can be answered without knowing the factorization of the RSA modulus. This is materially stronger than simply observing that a forged verification equation resembles a Strong RSA equation, the reduction must demonstrate how the supplied challenge element z appears in the simulation and how the forgery yields a non-trivial root of that same element.

2.4 The Naji–Abu Hasan construction and the present scope

Naji and Abu Hasan proposed a compact signature construction derived from the Cramer–Shoup line of work and then described a blind-signature transformation [11]. For the blind variant, their paper verifies the intended algebraic relation and argues informally that the blinding factor hides the message, its unforgeability discussion relies on the claim that security is inherited from the underlying signature scheme. No explicit security game, oracle simulation, extraction algorithm, or probability bound is supplied for that transformation. Later work has emphasized that black-box reductions for short, low-round blind signatures are subtle, particularly outside the random oracle model [7]. This analysis does not replace the standard one-more unforgeability requirement for blind signatures. It only examines whether the scheme's verification equation satisfies EUF-CMA security.

3 Preliminaries

3.1 Safe Primes

Definition 1 (**Safe Primes** [10])**.** A prime number p is called a safe prime if it can be written in the form

$$p = 2p' + 1,$$

where p' is also a prime number and a Sophie Germain prime.

3.2 Strong RSA (SRSA) Assumption

Definition 2 (**Strong RSA Assumption** [2])**.** Let $n = pq$ be an RSA modulus where p and q are large secret primes. Given n and a randomly chosen element $z \in \mathbb{Z}_n^*$, it is computationally infeasible for any probabilistic polynomial-time adversary to find an integer $y \in \mathbb{Z}_n^*$ and an exponent $e > 1$ such that:

$$y^e \equiv z \pmod{n}$$

3.3 Quadratic Residue

Definition 3 (Quadratic Residue [16])**.** Let $n \in \mathbb{N}$ and $a \in \mathbb{Z}_n$. The integer a is called a quadratic residue modulo n if there exists an integer $x \in \mathbb{Z}_n$ such that

$$x^2 \equiv a \pmod{n}.$$

3.4 Naji–Abu Hasan's Scheme

The proposed blind signature scheme [11] consists of three interactive phases between the signer and the sender, which are the setup phase, signing phase, and verification phase.

3.4.1 Setup Phase

This phase generates the system parameters and the keys for the signer. It is parameterized by two security parameters l and l' such that $l + l \leq l'$ (a reasonable choice might be $l = 256$ and $l' = 512$).

1. Select two random l-bit primes p and q such that $p = 2p' + 1$ and $q = 2q' + 1$, where both p' and q' are also primes .
2. Compute the RSA modulus $n = p \cdot q$.
3. Choose two random elements $h, x \in QR_n$, where QR_n denotes the set of quadratic residues modulo n.
4. The public key is published as $\mathsf{pk} = (n, h, x)$.
5. The corresponding secret key is kept private as $\mathsf{sk} = (p, q)$.

3.4.2 Signing Phase

To blindly sign a message $m \in \{0,1\}^*$, the signer and the sender execute the following interactive protocol:

1. The sender selects a random number $r \in \mathbb{Z}_n^*$ as a blinding factor and computes blinded value:
$$R \equiv r \cdot x \cdot h^{H(m)} \pmod{n}$$
where H is a collision-resistant hash function. The sender sends R to the signer.

2. **Signing**: Upon receiving R, the signer randomly picks an $(l+1)$-bit prime e. Using the secret key, the signer calculates $\overline{y}$ such that:

$$\overline{y}^e \equiv R \pmod{n}.$$

The signer then returns the pair $(\overline{y}, e)$ back to the sender.

3. **Unblinding**: Upon receiving $(\overline{y}, e)$, the sender extracts the actual signature y by removing the blinding factor r. The sender computes:

$$y \equiv r^{-(1/e)} \cdot \overline{y} \pmod{n}.$$

The resulting valid blind signature on the message m is (y, e).

3.4.3 Verification Phase

To verify that a putative signature (y, e) on a message m is valid with respect to the signer's public key $\mathsf{pk} = (n, h, x)$, any verifier can perform the following checks:

1. Check that e is an $(l+1)$-bit prime.
2. Verify that the following equation holds correctly:

$$x \equiv y^e \cdot h^{-H(m)} \pmod{n}$$

If both conditions are met, the signature is accepted as valid; otherwise, it is rejected.

Correctness: The signature of a message blindly signed can be checked using public key

$$\begin{aligned}
y^e &\equiv \left(r^{(-e^{-1})} \cdot \bar{y}\right)^e \pmod{n} \\
y^e &\equiv r^{-1} \cdot \bar{y}^e \pmod{n} \\
y^e &\equiv r^{-1} \cdot R \pmod{n} \\
y^e &\equiv r^{-1} \cdot r \cdot x \cdot h^{H(m)} \pmod{n} \\
y^e &\equiv x \cdot h^{H(m)} \pmod{n}
\end{aligned}$$

4 Security Proof of Naji-Abu Hasan Blind Signature Scheme

This section describes the security model and its proof of security to the Naji-Abu Hasan blind signature scheme in a random oracle model.

4.1 Security Model

Setup: The simulator $\mathcal{B}$ takes a security parameter 1^λ as input and runs the key generation algorithm to output the public parameter. It sends the parameters to the adversary. Note that the simulator neither knows nor keeps the private key generated in the setup phase.

Hash query: The adversary $\mathcal{A}$ sends hash queries on messages m_i of its choice. Upon receiving a query m_i, the oracle $\mathcal{O}_H$ checks whether m_i has previously been queried and responds accordingly.

Signing query: The adversary $\mathcal{A}$ sends signing queries of the message m_i of his choice. The signing oracle $\mathcal{O}_{\mathsf{Sign}}$ then responds by running the algorithm $\mathsf{Sign}(\cdot)$ and sends the resulting signature to the adversary $\mathcal{A}$. These queries may be asked adaptively so that the current query may depend on the previous responses.

Forgery: Upon terminating the signing phase, the adversary $\mathcal{A}$ outputs a message-signature pair (m^*, σ^*) that it wishes to forge. The forged signature σ^* must be a message m^* that was not previously submitted to the signature query.

4.2 Security Proof

We now formally provide the security proof via the following theorem.

Theorem 1. *Suppose that the hash function H is modeled as a random oracle. Then the Naji–Abu Hasan blind signature scheme is existentially unforgeable under chosen-message attacks (EUF-CMA) in the Random Oracle Model via the hardness assumption of the intractability of Strong RSA problem.*

Proof. Suppose there exists an adversary $\mathcal{A}$ who can break the Naji-Abu Hasan blind signature scheme in the EUF-CMA model, then we can construct a simulator $\mathcal{B}$ to solve the Strong RSA problem. $\mathcal{B}$ is given the Strong RSA instance of (n, z), and models H as random . The goal of $\mathcal{B}$ is to find an integer w and an exponent $E > 1$ such that

$$w^E \equiv z \pmod{n}.$$

The simulator $\mathcal{B}$ interacts with the adversary $\mathcal{A}$ and uses any successful forgery produced by $\mathcal{A}$ to solve the Strong RSA problem.

1. **Setup:** Simulator $\mathcal{B}$ initially takes on security parameter 1^λ as input and runs **KeyGen** to output the public keys $\{n, h, x\}$. These public keys are sent to $\mathcal{A}$.
2. **H-query:** Before receiving queries from the adversary, $\mathcal{B}$ randomly chooses an integer $i^* \in [1, q_H]$, where q_H denotes the number hash queries to the random oracle. The simulator $\mathcal{B}$ prepares a hash list $\mathcal{L}_H$ to answer all hash queries. The list is initially empty.

   For any message m_i queried by the adversary $\mathcal{A}$, the simulator $\mathcal{B}$ first checks if the query already exists in $\mathcal{L}_H$. If it does, $\mathcal{B}$ returns the corresponding hash value. Otherwise, it sends to random oracle who outputs H_i, stores the tuple (i, m_i, H_i) in $\mathcal{L}_H$, and returns $H_i = H(m_i)$ to $\mathcal{A}$.
3. **Signing Query:** For any signing query on a message m_i, the adversary $\mathcal{A}$ randomly samples r_i and computes blinded value $R_i \equiv r_i x h^{H_i} \pmod{n}$, sends the tuple (m_i, R_i) to the simulator $\mathcal{B}$. The simulator $\mathcal{B}$ first checks if m_i has been queried:
   (a) If $i = i^*$, that is if the message m_i is the i^*-th queried message set by the simulator $\mathcal{B}$ in $\mathcal{L}_H$, then abort.
   (b) Otherwise if $i \neq i^*$, $\mathcal{B}$ obtains the corresponding tuple (i, m_i, H_i) from $\mathcal{L}_H$ and forward the tuple (i, m_i, H_i, R_i) to the signing oracle. The signing oracle returns a signature (y_i, e_i) satisfying the signing equation of the proposed scheme. Finally, $\mathcal{B}$ returns the signature (y_i, e_i) to the adversary $\mathcal{A}$ and updates the tuples $(i, m_i, H_i, R_i, y_i, e_i)$ into $\mathcal{L}_H$.

   One can observe that the correctness of the output signature (y_i, e_i)

$$y_i^{e_i} \equiv x h^{H_i} \pmod{n} \tag{2}$$

   validates the blind signature scheme.

4. **Forgery**: After completing the random-oracle and signing queries, the adversary $\mathcal{A}$ stops and outputs a message-signature pair $(m^*, (y^*, e^*))$ that it wishes to forge:
   (a) If $i \neq i^*$, that is if the message m_i is not the i^*-th queried message set by the simulator $\mathcal{B}$ in $\mathcal{L}_H$, then abort.
   (b) Otherwise if $i = i^*$, $\mathcal{B}$ randomly selects (y, e) from the $\mathcal{L}_H$ list and outputs the solution to the Strong RSA problem as

$$y^e \equiv z \pmod{n},$$

   ending the EUF-CMA game.

It remains now to evaluate the advantage of the simulated game described above. As $\mathcal{A}$ has the advantage of ε in forging the correct signature following the hardness assumption of Strong RSA problem. Hence,

$$\Pr\left[\mathsf{Verify}\left(\mathsf{pk}, m^*, (y^*, e^*)\right) = 1 \ \wedge \ (m_i, (y_i, e_i)) \notin (m^*, (y^*, e^*))\right] \geq \varepsilon.$$

with ε is non-negligible.

If the simulator $\mathcal{B}$ successfully guesses i^*, then all the queried signature are simulate-able, this implies that the signature forgery are reducible since message m^* cannot be chosen during the signature query. Therefore the probability of successful simulation is given by $\frac{1}{q_H}$. Combining all the above scenarios, the success probability of $\mathcal{B}$ is hence bounded by

$$\mathrm{Adv}_{\mathcal{B}}^{\mathrm{SRSA}}(\lambda) \leq_{\mathrm{poly}} \mathrm{Adv}_{\Pi,\mathcal{A}}^{\mathrm{EUF-CMA}}(\lambda) \leq \frac{\varepsilon}{q_H},$$

where q_H denote the numbers of H-queries. This completes the proof of security of the proposed Naji-Abu Hasan blind signature scheme. ■

5 Conclusion

In this paper, we successfully formalized the security of the Naji-Abu Hasan blind signature scheme by providing a rigorous mathematical reduction. While the foundational scheme was highly efficient, its interactive blind variant previously lacked a formal security proof. We bridged this gap by demonstrating that the scheme achieves existentially unforgeable under chosen-message attacks (EUF-CMA) .

By framing our reduction around the presumed intractability of the Strong RSA problem, we proved that any successful polynomial-time adversary capable of outputting a forged blind signature can be efficiently leveraged to extract the Strong RSA root. Because the interactive query phase relies on a cryptographic hash function to decouple the message from the signer's view, our formal reduction was strictly conducted within the Random Oracle Model (ROM). Ultimately, this proof successfully establishes the protocol's formal security, validating its viability for modern privacy-preserving applications such as digital cash and secure voting.

■

References

[1] Masayuki Abe and Tatsuaki Okamoto. Provably secure partially blind signatures. In *Annual international cryptology conference*, pages 271–286. Springer, 2000.

[2] Niko Barić and Birgit Pfitzmann. Collision-free accumulators and fail-stop signature schemes without trees. In *International conference on the theory and applications of cryptographic techniques*, pages 480–494. Springer, 1997.

[3] Bellare, Namprempre, Pointcheval, and Semanko. The one-more-rsa-inversion problems and the security of chaum's blind signature scheme. *Journal of Cryptology*, 16(3):185–215, 2003.

[4] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pages 62–73, 1993.

[5] David Chaum. Blind signatures for untraceable payments. In *Advances in Cryptology: Proceedings of Crypto 82*, pages 199–203. Springer, 1983.

[6] Ronald Cramer and Victor Shoup. Signature schemes based on the strong rsa assumption. *ACM Transactions on Information and System Security (TISSEC)*, 3(3):161–185, 2000.

[7] Marc Fischlin and Dominique Schröder. On the impossibility of three-move blind signature schemes. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 197–215. Springer, 2010.

[8] Shafi Goldwasser, Silvio Micali, and Ronald L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.

[9] Ari Juels, Michael Luby, and Rafail Ostrovsky. Security of blind digital signatures. In *Annual International Cryptology Conference*, pages 150–164. Springer, 1997.

[10] Lee McCulloch-James. Safe primes in the eisenstein integers. *structure*, 5:2, 2026.

[11] Akram Naji and Yahya Abu Hasan. A simple provably secure signature scheme based on the strong RSA and its blind signature scheme. In *Proceedings of International Cryptology Workshop and Conference*, 2008.

[12] David Pointcheval and Jacques Stern. Provably secure blind signature schemes. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 252–265. Springer, 1996.

[13] David Pointcheval and Jacques Stern. Security arguments for digital signatures and blind signatures. *Journal of cryptology*, 13(3):361–396, 2000.

[14] Ronald L Rivest, Adi Shamir, and Len Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[15] Victor RL Shen, Yu Fang Chung, Tzer Shyong Chen, Yu An Lin, et al. A blind signature based on discrete logarithm problem. *International Journal of Innovative Computing, Information and Control*, 7(9):5403–5416, 2011.

[16] Ferucio Laurenţiu Ţiplea. The jacobi symbol problem for quadratic congruences and applications to cryptography. *Mathematics*, 14(3):465, 2026.

[17] Jiazheng Zhang, Shouwei Li, and Minghui Chen. Multifactor sm9 blind signature large models data privacy preservation approach. *Peer-to-Peer Networking and Applications*, 18(4):184, 2025.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 253–264.

Provable Security Analysis of a Dual-Hardness Digital Signature Scheme

Ernie Noorsyafiqah Nor Hisham[1] and Wan Nur Aqlili Ruzai[1,2]

[1] Malaysia Cryptology Technology and Management Centre, c/o Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia ernie@upm.edu.my
[2] School of Distance Education, Universiti Sains Malaysia, Penang 11800, Malaysia aqliliruzai@usm.my

Abstract. This paper presents a formal security analysis of the digital signature scheme proposed by Ismail et al. [4], whose security is based on the combined hardness of the Integer Factorisation Problem (IFP) and the Discrete Logarithm Problem (DLP). Although the scheme was introduced as an improvement over earlier signature schemes relying on either a single assumption or dual-hardness assumptions, its security has previously been supported only by heuristic arguments. In particular, the original analysis informally claims that recovering the private key requires solving both the IFP and the DLP, but it provides neither a reduction-based security proof against a formally defined adversary nor a proof of existential unforgeability under chosen-message attacks. Consequently, it remains unclear whether a valid signature could be forged without directly solving either underlying problem. To address this gap, we provide a comprehensive theoretical analysis under standard cryptographic attack models. We first revisit the construction of the scheme and establish the mathematical foundations required for the analysis. We then show that recovering its secret parameters is computationally related to solving the underlying hard problems. Using reduction-based techniques and the Forking Lemma in the Random Oracle Model, we further prove that an adversary capable of producing a valid forgery can be used to solve the DLP. Our analysis establishes the existential unforgeability of the scheme under both no-message and chosen-message attacks. These results provide a stronger theoretical foundation for the signature scheme of Ismail et al. and increase confidence in the security of cryptographic constructions based on dual-hardness assumptions.

Keywords: Digital signature · Provable security · Integer factorisation problem · Discrete logarithm problem · Dual hardness assumption

1 Introduction

Digital signature techniques are a crucial cryptographic application, introduced conceptually alongside public-key cryptography by Diffie and Hellman [1], with the first concrete construction given by Rivest et al. [10]. They provide the electronic counterpart to the traditional concept of document signing. The digital signature process is a cryptographic exchange where a signer generates a unique message digest via a hash function and encrypts it with their private key to ensure authenticity and integrity. Upon receipt, the verifier uses the signer's public key to decrypt the signature and reveal the original hash, while simultaneously re-hashing the received message to compare the two results. A successful match confirms the message is untampered and originated from the legitimate sender, a process secured in this scheme by the computational difficulty of solving the IFP and DLP.

Among the most prominent foundations for public-key cryptography are the IFP and the DLP. Factorisation-based signature schemes, typified by constructions in the spirit of RSA, derive security from the presumed difficulty of factoring a large composite integer into its prime constituents [10]. In such schemes, the secret key is usually related to the factorisation of a modulus ($n = pq$), while the public key is a function over the ring ($\mathbb{Z}_n$). The hardness of computing multiplicative inverses or extracting certain roots without knowledge of the factors ensures the infeasibility of forging signatures.

Discrete-logarithm-based signatures, on the other hand, rely on the assumed intractability of computing discrete logarithms in suitably chosen groups, such as multiplicative subgroups of finite fields [2] or groups of points on elliptic curves [5]. Given a generator (g) of a large cyclic group and an element ($y = g^x$), recovering the exponent (x) from (y) is believed to be computationally infeasible. Signature schemes built on this paradigm typically use exponentiations in groups of large prime order, with verification equations that trace back to algebraic relations involving discrete logarithms. Both IFP-based and DLP-based designs have been extensively deployed and analyzed, forming the backbone of many standardized cryptographic protocols.

Despite their success, an important concern arises from the reliance of most conventional signature schemes on a single hardness assumption. Advances in algorithms, computational power, or quantum computing may eventually render either the IFP or DLP tractable in certain parameter regimes. If a scheme is based solely on one of these problems, the breaking of that problem would directly compromise the security of all systems instantiated under that design. This observation has motivated a line of research on dual-hardness or multi-hardness cryptographic schemes, which seek to enhance robustness by simultaneously leveraging more than one intractable problem ([3]; [7]; [6]; [11, 12]; [13]).

Dual-hardness signature schemes based on both IFP–DLP schemes aim to mitigate the risk that a single algorithmic breakthrough destroys the security of the entire system. This is because of the common perception that it is highly improbable that several cryptographic presumptions would simultaneously become easy to solve.

An early approach linking the IFP to key distribution security was proposed by Kevin S. McCurley in 1988 [8]. His scheme modified the Diffie–Hellman key exchange so that breaking the system requires both factoring a large composite number and solving the DLP. Later, Lei Harn [3] improved this design by addressing the imbalance in security parameters between the two problems. By introducing a special prime structure ($p = 2p'q' + 1; \quad p', q' =$ prime number), his scheme balanced the security level of the IFP-DLP assumptions and improved efficiency compared to the earlier model.

By 2009, the focus shifted toward optimizing signature schemes. Ismail et al. observed that while many schemes had been designed based on both IFP and DLP, several had been proven insecure [4]. Their scheme was designed to provide a higher level of security compared to single-assumption signature schemes. Ismail et al. [4] argue that solving both problems simultaneously is significantly more difficult than solving either one individually, thereby increasing resistance against cryptanalytic attacks. Security analysis confirms that even if an adversary can break one problem (e.g., if IFP becomes solvable), the signature (K, u) remains protected because the attacker still lacks information related to the other problem (i.e., DLP).

The primary contribution of this research is the establishment of a rigorous provable security analysis for the digital signature scheme proposed by Ismail et al. [4], which is

constructed from the combined hardness assumptions of the IFP and the DLP. Although the original scheme was designed to achieve stronger cryptographic security through a dual-hardness framework, its formal security properties had not previously been analysed in a rigorous theoretical setting. This work addresses that gap by providing reduction-based security proofs under standard cryptographic attack models. In particular, we demonstrate that recovering the secret parameters or forging a valid signature is computationally related to solving the underlying hard mathematical problems. By employing the Forking Lemma within the Random Oracle Model, the analysis further establishes existential unforgeability against both no-message and chosen-message attacks. These results strengthen the theoretical foundation of the scheme and provide greater confidence in the security and reliability of dual-hardness cryptographic constructions.

In Section 2, we present the preliminaries required for this paper, including the theorems and definitions used for the provable security analysis. In Section 3, we review the Ismail et al. [4] signature scheme, describing its key generation, signature generation, and verification processes. In Section 4, we present the main result of this paper, which is the proof of security of the scheme. Finally Section 5, concludes the paper by contributions and outlines strategic directions for future research in dual-hardness cryptographic frameworks.

2 Preliminaries

This section provides the necessary preliminaries, including the definitions and the lemma used throughout the paper.

2.1 The Oracle Replay Attack

In cryptographic security models, adversaries are typically modeled as probabilistic polynomial-time (PPT) algorithms interacting with oracles. A common approach is to use such an attacker to solve a hard problem via a reduction. This is achieved through an *oracle replay attack* (see Figure 1), where the attack is repeated polynomially with different oracle responses (the $\mathcal{Q}_i$'s are the queries and the ρ's are the answers). By doing so, the attacker is forced to produce two related forgeries on the same message (m), namely (σ_1, h, σ_2) and $(\sigma_1', h', \sigma_2')$, such that $\sigma_1 = \sigma_1'$ but $h \neq h'$. This allows the extraction of a solution to a hard problem. A key challenge in this reduction is to simulate oracle and signer interactions in a way that remains indistinguishable from real executions, even without access to the secret key. This technique, and the terminology adopted here, follows the treatment of Pointcheval and Stern [9], on which the Forking Lemma of Section **??** is built.

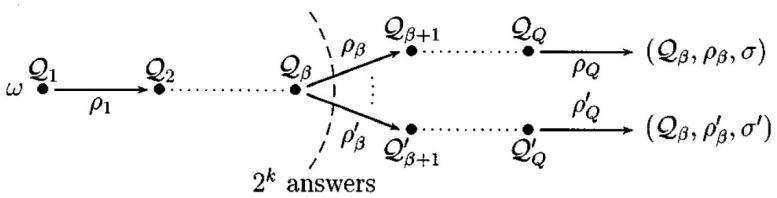


Figure 1: The Oracle Replay Attack

2.2 The Forking Lemma

The forking lemma was introduced by Pointcheval et al. [9] in their proof of security for Schnorr signatures. This lemma uses the "*oracle replay attack*" by a polynomial replay of the attack with the same random tape and a different oracle, we obtain two signatures of a specific form, which open a way to solve the underlying hard problem. This lemma will be used repeatedly in the following sections.

Lemma 1 (Forking Lemma, Pointcheval et al. [9])**.** *Let $\mathcal{A}$ be a probabilistic polynomial time Turing computer whose input consists only public data as input. If $\mathcal{A}$ can find a valid signature σ with respect to a message m and a random oracle output h with non-negligible probability, then if $\mathcal{A}$ is executed again with the same source of randomness but a different random oracle, then $\mathcal{A}$ will produce a second valid forgery σ' for the same message m with respect to a random oracle output h', such that $h \neq h'$.*

3 Review of Ismail et. al [4] signature scheme

This section provides an overview of the Ismail et. al [4] signature scheme, describing the key generation, signing, and verification algorithms.

Algorithm 1 Generate parameters

Input: Security parameters
Output: Public keys, pk: (n, e) and (p, y, g).
Private keys, sk: d and x.

1: Choose large p such that n is a factor of $p-1$ that is the product of two safe prime p' and q', i.e., $n = p'q'$.
2: Compute $\phi(n) = (p'-1)(q'-1)$.
3: Choose $e \in \{1, 2, ..., n-1\}$ such that $\gcd(e, \phi(n)) = 1$.
4: Calculate d such that $ed \equiv 1 \mod \phi(n)$.
5: Select at random an integer x from $\{1, 2, ..., p-1\}$.
6: Choose $g \in \{0, 1, 2, ..., p-1\}$ with order n satisfying $g^n \equiv 1 \mod p$.
7: Calculate $y \equiv g^x \mod p$.

Algorithm 2 Signing message

Input: $pk = n, p, g$, $sk = x, d$ and message, M.
Output: Signature pair, (K, u).

1: Select an integer $0 < r < n$ such that $\gcd(r, n) = 1$.
2: Compute $K \equiv g^r \mod p$.
3: Calculate $s \equiv h(M)x + Kr \mod n$.
4: Calculate $u \equiv s^d \mod n$.

Algorithm 3 Verification

Input: g, e, y, hash value of message, $h(M)$ and signature pair, (K, u).
Output: "Accept" or "Reject".

1: $A \equiv g^{u^e} \mod p$.
2: $B \equiv y^{h(M)} K^K \mod p$.
3: if $A == B$, then "Accept"; else "Reject".

3.1 Numerical example

To illustrate the proposed signature scheme, we present the following numerical example that illustrates Ismail et al. [4] scheme.

Example 1. Consider two 64-bit safe primes and its corresponding keys.

$$\begin{aligned}
p' &= 9340084625692080119\\
q' &= 17722287166531688843\\
n &= 165527661896222684148953762906734412317\\
p &= 427061367692254525104300708299374783777787\\
g &= 483\\
e &= 755218443643942072306572939475625934 37\\
d &= 840662713099047777739352089538427979 85\\
x &= 165170042541319230361117500263505001121 44\\
y &= 146039764379060737033370073202930194263338
\end{aligned}$$

Next, we set the signing message as Algorithm 2.
For a message, $M = 492032246618711190251485398146069603 68$, assume a hash value $h(M) = 113012515454705852756111335590313811816318301901 91$. Choose a random integer $r = 130459566982345089076827303788460678579$ such that $\gcd(r, n) = 1$, and compute $K \equiv g^r \mod p \equiv 116246936930246367812728377005941129845 20$. Next, compute

$$\begin{aligned}
s &\equiv h(M)x + Kr \mod n\\
&\equiv 118461829192524998025540908016479925
\end{aligned}$$

and

$$\begin{aligned}
u &\equiv s^d \mod n\\
&\equiv 118302533866363795045472650576220023 67.
\end{aligned}$$

The resulting signature is

$$(K, u) = (116246936930246367812728377005941129845 20,\ 118302533866363795045472650576220023 67)\,.$$

To verify the validity of the signature (K, u), we evaluate both sides of the verification equation

$$g^{u^e} \equiv y^{h(M)} K^K \pmod p,$$

as specified in Algorithm 3. For the left-hand side, we obtain

$$g^{u^e} \equiv 424456023901639014040888812402111922923 33 \pmod p,$$

whereas the right-hand side gives

$$y^{h(M)} K^K \equiv 424456023901639014040888812402111922923 33 \pmod p.$$

Since both sides are congruent modulo p, the signature (K, u) is valid.

4 Provable Security of Ismail et. al [4] Scheme

4.1 Total Break under Passive Attack

Theorem 1. *This scheme is strong against total break under passive attack if the IFP and the DLP hold.*

Proof. The Ismail et. al [4] scheme, the public key consists of the pair (y, e) while the private key consists of (x, d). Recovering the private value d from $ed \equiv 1 \mod \phi(n)$ is equivalent to the RSA private-key recovery problem which is equivalent to the IFP. However, recovering private value x from $y = g^x \mod p$ is precisely the DLP.

Importantly, the two secret components x and d are derived from independent number-theory structures. One is defined modulo p, and the other is modulo $\phi(n)$. There is no parameter shared between the two computations. Consequently, breaking the scheme requires solving DLP and IFP separately, and success on one does not provide any advantage toward solving the other.

To justify security under a total-break passive attack, we therefore, analyse the hardness of each problem individually. For each problem, we construct a reduction showing that breaking the corresponding component of the scheme is equivalent to solving the underlying hard problem—that is, the reduction holds in both directions.

First, we will show that RSA private-key recovery $\leq_T$ IFP. Suppose there exists a polynomial-time machine $\mathcal{A}$ that solves IFP. Consider an machine $\mathcal{B}$ that uses the IFP output from machine $\mathcal{A}$ to find private key as follows:

Algorithm 4 RSA private-key recovery $\leq_T$ IFP

Input: n, e
Output: d

1: $\mathcal{B}$ receives n, e, submit N to $\mathcal{A}$.
2: $\mathcal{B}$ receives (p', q') from $\mathcal{A}$, compute $\phi(n) = (p' - 1)(q' - 1)$.
3: $\mathcal{B}$ runs Extended Euclidean Algorithm on $(\phi(n), e)$ and obtain (d, u) such that $e \cdot d + u \cdot \phi(n) = 1$.
4: Output d.

Next, we want to show that if an attacker can get the RSA private key d (given n and e), then the attacker can factor n efficiently. Let θ be an oracle that produces the RSA private key when given an RSA public key.

An algorithm to factor n is:-

Algorithm 5 IFP $\leq_T$ RSA private-key recovery

Input: n, e
Output: factorised prime number, p'

1: Choose a large random prime e.
2: Run $\theta(n, e)$ to obtain d.
3: Compute $\hat{M} = ed - 1$.
4: Write $\hat{M} = 2^r m$ where m is odd.
5: Choose a random $a \in Z_n^*$.
6: Compute $\gcd(a, n)$. If $\neq 1$, repeat (5).
7: $a_0 = a^m$.
8: For $i = 1, ..., r$,
9: $\quad a_i = a_{i-1}^2 \pmod n$.
10: $\quad$ If $a_i = 1$, stop iteration. If $a_r \neq 1$, Repeat from (5).
11: Compute $p'_k = \gcd(a_{i-1}, n)$.
12: Repeat (5) to (11) until $\prod p'_k = n$.

The reply d from θ is such that $ed \equiv 1 \bmod n$. So, $M = ed - 1$ is such that $M|\phi(n)$. So for any $g \in Z_n^*$, $g^M \equiv g^{k\phi(n)} \equiv 1 \bmod n$.Thus, $a_r = 1$, for a '*good guess*' of a.

For any prime p' which divides n, if a is a quadratic non-residue modulo p', then $a_0 \equiv a^m \bmod n \not\equiv 1 \bmod p'$ because for any t with $a^t \equiv 1 \bmod p'$, t must divide M and so t must be even.

Since $a_0 \equiv 1$ and $a_r \neq 1$, it is highly likely there is some i such that $a_i \not\equiv 1$ but $a_{i+1} \equiv a_i^2 \equiv 1$. Then such a_i is a non-trivial square root of 1 modulo n and $P(a_i \not\equiv -1) \geq \frac{1}{2}$. That implies $a_i \equiv -1 \mod p_1'$ and $a_i \equiv 1 \mod p_2'$ for some p_1', p_2' prime factor of n. Since this gives $a_{i+1} \equiv 1 \mod p_1' \Rightarrow (a_i + 1)|p_1'$, $\gcd(a_i + 1, n) = p_1'$. Thus, step (12) produces a prime factor of n, and the full factorisation of n can be recovered by repeating the procedure on the resulting cofactor.

This establishes that RSA private-key recovery and IFP are computationally equivalent, i.e IFP $\equiv_T$ RSA private-key recovery. This equivalence concerns on the factoring-based component (d) of the Ismail et al. [4] private key.

Next, we show that the private key extraction problem is reducible to solving the DLP. Suppose there exists a polynomial-time machine $\mathcal{C}$ that solves the DLP. Consider an machine $\mathcal{B}$ as follows:

Algorithm 6 Solving secret key $\leq_T DLP$

Input: p, g, y.
Output: x.

1: $\mathcal{B}$ receives public key $pk = (p, g, y)$.
2: $\mathcal{B}$ submits pk to $\mathcal{C}$.
3: $\mathcal{C}$ outputs the $sk = x$.
4: Output x.

Because both the DLP and IFP are assumed to be computationally intractable for any probabilistic polynomial-time (PPT) algorithm, the success of the Simulator creates a fundamental logical contradiction. Consequently, the scheme is proven to be secure under the specified attack model.

□

4.2 Existential Unforgeability under No-Message and Chosen-Message Attacks (EUF-NMA/EUF-CMA)

Existential unforgeability (EUF) is the standard security notion for digital signature schemes. Informally, a signature scheme is said to be existentially unforgeable if no probabilistic polynomial-time (PPT) adversary $\mathcal{A}$, given the public key pk and some interaction with a signing oracle and/or a random oracle, outputs a valid signature (K, u) on any message m, including a message of the adversary's own choosing, that was not directly obtained from the legitimate signer. The adversary is not required to forge a signature on a specific, pre-chosen message, nor is it restricted in how it selects the message it ultimately forges. It is this general definition that distinguishes EUF from weaker notions such as selective forgery.

Since the Ismail et al. [4] scheme constructs its signature (K, u) around the DLP relationship $s = h(M)x + Kr$, where x is the secret exponent satisfying $y \equiv g^x \mod p$, the existential unforgeability of the scheme is naturally tied to the hardness of the DLP

rather than the IFP. That is, if an adversary,$\mathcal{A}$ can produce a valid existential forgery, this forgery can be transformed into a solution of the DLP instance embedded in the public key. This reduction is achieved through the Forking Lemma (Section 2.2). By replaying $\mathcal{A}$ under the same random tape but different oracle responses $h \neq h'$ yield two forgeries on the same commitment K, from which x can be isolated algebraically.

We consider this reduction under two attack models, distinguished by the capabilities granted to $\mathcal{A}$ during the Query Phase. The first is existential unforgeability against a no-message attack (EUF-NMA), in which $\mathcal{A}$ has no access to a signing oracle and must forge using only public information and random oracle queries. The second is existential unforgeability against a chosen-message attack (EUF-CMA), in which $\mathcal{A}$ is additionally granted adaptive access to a signing oracle before producing its forgery.

Because the Setup Phase and the Forgery Phase of the reduction are identical regardless of which attack model is considered, and the two models differ only in what $\mathcal{A}$ is permitted to query during the Query Phase, we present both cases as a single security game with a branching Query Phase, rather than as two independent proofs. This avoids repeating the Setup and Extraction arguments twice and makes explicit that the additional signing-oracle access granted under EUF-CMA does not require a fundamentally different reduction technique. The structure of this game is described below.

1. **Setup Phase:** The Challenger $\mathcal{C}$ initiates the security game by presenting the Simulator $\mathcal{S}$ with an instance of the DLP, (p, g, Y) and n, for which $\mathcal{S}$ must find x such that $Y \equiv g^x \pmod{p}$. Since $\mathcal{S}$ does not know x, it sets the public key of the signature scheme as $y = Y$ and forwards $pk = (p, g, y, n)$ to $\mathcal{A}$.

2. **Query Phase:** The two cases are illustrated in Figures 2 and 3 respectively.

   - **Case 1, EUF-NMA:** As shown in Figure 2, $\mathcal{A}$ is strictly forbidden from requesting any valid signatures, and no signing oracle is simulated. The interaction is limited solely to the Random Oracle, where $\mathcal{S}$ manages hash queries by maintaining a consistent table of random hash assignments, while waiting for $\mathcal{A}$ to construct a forgery using only the provided public parameters.
   - **Case 2, EUF-CMA:** As shown in Figure 3, $\mathcal{A}$ is additionally permitted to make signature queries alongside hash queries. Since $\mathcal{S}$ does not possess the secret key, it answers each signature query for a message m_i by performing a backwards simulation, in which it selects random signature components picks $(K^{(i)}, u^{(i)})$ at random and programs $H^{(i)} = f(m_i, K^{(i)})$. Because the probability of a hash collision during these queries is negligible, the simulated responses are statistically indistinguishable from those of a legitimate signer, so $\mathcal{A}$ cannot distinguish $\mathcal{S}$ from a real signer throughout the query phase.

3. **Forgery Phase:** $\mathcal{A}$ outputs a forgery (m^*, K, H, u). $\mathcal{S}$ identifies the corresponding hash query and rewinds the entire simulation to the state immediately preceding it. By supplying a different hash value H', $\mathcal{S}$ "forks" the execution path. By the Forking Lemma, $\mathcal{A}$ outputs a second valid signature u' for the same message and commitment K, adapted to the new hash challenge.

$$(m^*, K, H, u) \text{ and } (m^*, K, H, u), \qquad H \neq H.$$

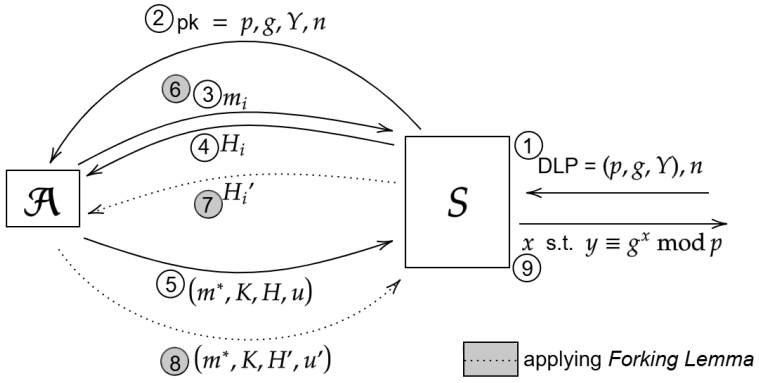


Figure 2: Illustration of EUF-NMA

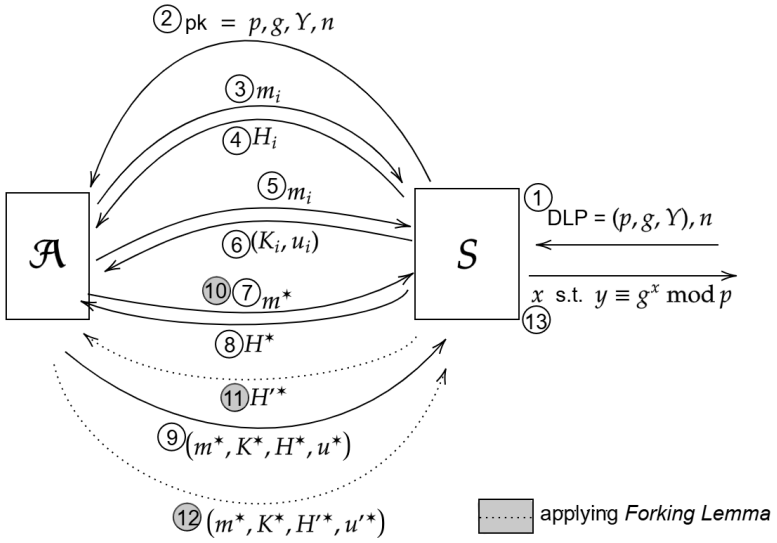


Figure 3: Illustration of EUF-CMA

$\mathcal{S}$ then uses these two related forgeries to solve the DLP, exactly as follows.

Theorem 2. *Assume that, within a time bound T, an attacker $\mathcal{A}$ performs either an EUF-NMA or an EUF-CMA against the Ismail et al. [4] signature, making at most Q queries to the random oracle (and, in the EUF-CMA case, at most R queries to the signing oracle). Then the discrete logarithm in the subgroup of order n generated by g can be solved.*

Proof. By the assumption $\mathcal{A}$ exists, we will build an algorithm to solve the DLP. By the use of the Forking Lemma, after a polynomial replay of the attacker $\mathcal{A}$, we obtain two valid signatures (m, K, h, u) and (m, K, h', u') with $h \neq h'$.

In the EUF-CMA case, this replay additionally requires $\mathcal{S}$'s R simulated signature responses to remain indistinguishable from a real signer's, as established in the Query Phase above. In the EUF-NMA case, no such simulation is needed since $\mathcal{A}$ never queries the signing oracle. From this point on, the extraction proceeds identically in both cases.

Since, (m, K, h, u) and (m, K, h', u') are the valid signatures for m, we have that:

$$g^{u^e} \equiv y^h K^K \mod p \tag{1}$$
$$g^{u'^e} \equiv y^{h'} K^K \mod p \tag{2}$$

Then, divide 1 by 2:

$$g^{u^e - u'^e} \equiv y^{h-h'} \mod p.$$

Since $y = g^x \mod p$,

$$g^{u^e - u'^e} \equiv g^{x(h-h')} \mod p$$
$$u^e - u'^e \equiv x(h - h') \bmod p - 1$$

To extract x, the reduction must compute:

$$x \equiv (u^e - u'^e)(h - h')^{-1} \bmod p - 1.$$

Verifying this against g^x, then:

$$\begin{aligned} g^x &\equiv g^{(u^e - u'^e)(h-h')^{-1}} \mod p \\ &\equiv g^{(u^{de} - u'^{de})(h-h')^{-1}} \mod p \\ &\equiv g^{(s-s')(h-h')^{-1}} \mod p \\ &\equiv g^{[(hx+Kr)-(h'x+Kr)]\,(h-h')^{-1}} \mod p \\ &\equiv g^{(hx-h'x)(h-h')^{-1}} \mod p \\ &\equiv g^{x(h-h')(h-h')^{-1}} \mod p \\ &\equiv y \mod p. \end{aligned}$$

Thus, whether $\mathcal{A}$ operates under the EUF-NMA restriction (no signing oracle) or the EUF-CMA capability (signing oracle available), the successful extraction of x shows that forging a signature, in either attack model, is equivalent to solving the DLP. Given the widely accepted assumption that the DLP is computationally intractable within a subgroup of prime order, no such polynomial-time adversary $\mathcal{A}$ can exist in either case. Consequently, the scheme is proven to be both EUF-NMA and EUF-CMA secure, contradicting the initial assumption of the attacker's existence.

□

Remark 1. Unforgeability rests entirely on the hardness of the DLP. The signing oracle in EUF-CMA does not weaken this bound, since $\mathcal{S}$ simulates signatures without knowledge of x, and extraction depends only on the fork $h \neq h'$, not on how m^* was chosen.

5 Conclusion

The security analysis presented in this paper strengthens the theoretical foundation of the Ismail et. al [4] signature scheme by formally demonstrating its resistance against total break under passive attacks as well as existential forgery under both no-message and chosen-message attacks. By employing reduction-based arguments and the Forking Lemma within the Random Oracle Model, the analysis shows that a successful forgery against the scheme would imply an efficient solution to the underlying DLP, while the recovery of the RSA-related secret component remains computationally equivalent to solving the

IFP. The independence between these two mathematical assumptions further enhances the robustness of the scheme, since compromising one component does not directly weaken the other. These results provide stronger confidence in the security of dual-hardness signature constructions and reinforce their relevance as an alternative to conventional single-assumption schemes.

Future research may focus on extending the proof framework to stronger adversarial models and exploring security reductions in the Standard Model without relying on random oracles. It would also be valuable to investigate the practical efficiency of the scheme in real-world deployment scenarios, including constrained and distributed environments. In light of ongoing developments in quantum computing, another important direction is the study of hybrid constructions integrating classical dual-hardness assumptions with post-quantum cryptographic primitives to achieve stronger long-term security guarantees. Further comparative analysis with existing dual-hardness and post-quantum signature schemes may also provide deeper insight into the scalability, efficiency, and cryptographic resilience of the proposed framework.

References

[1] Whitfield Diffie and Martin E Hellman. New directions in cryptography. In *Democratizing cryptography: the work of Whitfield Diffie and Martin Hellman*, pages 365–390. 2022.

[2] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469–472, 1985.

[3] L Harn. Public-key cryptosystem design based on factoring and discrete logarithms. 1993.

[4] ES Ismail, NMF Tahat, and RR Ahmad. A new signature scheme based on factoring and discrete logarithms. *Journal of Discrete Mathematical Sciences and Cryptography*, 12(3):313–318, 2009.

[5] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177):203–209, 1987.

[6] Chi-Sung Laih and Wen-Chung Kuo. New signature schemes based on factoring and discrete logarithms. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 80(1):46–53, 1997.

[7] N-Y Lee and T Hwang. Modified harn signature scheme based on factorising and discrete logarithms. *IEE Proceedings-Computers and Digital Techniques*, 143(3):196–198, 1996.

[8] Kevin S McCurley. A key distribution system equivalent to factoring. *Journal of Cryptology*, 1(2):95–105, 1988.

[9] David Pointcheval and Jacques Stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 13(3):361–396, 2000.

[10] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[11] Zuhua Shao. Signature schemes based on factoring and discrete logarithms. *IEE Proceedings-Computers and Digital Techniques*, 145(1):33–36, 1998.

[12] Zuhua Shao. Digital signature schemes based on factoring and discrete logarithms. *Electronics Letters*, 38(24):1518–1519, 2002.

[13] Ching-Te Wang, Chu-Hsing Lin, and Chin-Chen Chang. Signature schemes based on two hard problems simultaneously. In *17th International Conference on Advanced Information Networking and Applications, 2003. AINA 2003.*, pages 557–560. IEEE, 2003.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 265–272.

State-Dependent Vulnerability Detection via Enriched Semantic Graph Representations

Yassin Ben Abdallah[1], Mohammed Nasser Al-Andoli[2], Mustafa Alobaedy[1], Ibrahim M Alwayle[3], Nabil Hasan Al-kumaim[4] and Mohammed Ahmed Salem[5]

[1] Faculty of Computing & Informatics, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia
{ben.abdallah.yassin@student,mustafa.alobaedy}@mmu.edu.my
[2] Cybersecurity and Quantum Computing, CoE for Advanced Cloud, Faculty of Computing & Informatics, Multimedia University, Cyberjaya, Malaysia nasser.alandoli@mmu.edu.my
[3] Department of Computer Science, College of Science and Arts in Sharurah, Najran University, Najran, Saudi Arabia imgebah@nu.edu.sa
[4] Department of Information Systems, Faculty of Management, Universiti Teknologi Malaysia, Johor Bahru, Malaysia nabilhasansaleh@utm.my
[5] Faculty of Electronics and Computer Technology and Engineering, Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia binghooth@utem.edu.my

Abstract. State-dependent vulnerabilities are considered hard to detect using static or dynamic analysis, although the current shift to GNN (Graph Neural Network) still faces several limitations to properly detect these kinds of flaws, like CWE-416 use-after-free and CWE-362 race conditions. We have proposed an approach to detect this kind of vulnerability, which is called SDVD (State-Dependent Vulnerability Detection). We added three types of relational edges to representation graphs: Object Lifecycle (OL), Value Constraint (VC), and Synchronization (SYNC). We designed a multi-component node representation that contains 176 dimensions composed of four separate components: syntactic component, value-state component, object-state component, and synchronization component. To process these, we created a three-stage architecture starting with the spatial dimension (relation-typed GNN), then the temporal dimension (BiLSTM on execution paths), and in the end, the reasoning dimension (component attention and fusion). It has been evaluated on a related vulnerability that exists in the Big-Vul dataset, where it outperforms the dedicated baselines.

Keywords: Vulnerability Detection · Graph Neural Networks · Semantic State Modeling · Program Representation Learning · Software Security · Object Lifecycle Analysis · Use-After-Free Detection · Race Condition Detection · BigVul Dataset

1 Introduction

Existing static, dynamic, and GNN-based approaches are not able to capture state-dependent vulnerability, while they mainly rely on the structural program representations that don't have the semantic capabilities [4, 19, 5, 3, 8, 6]. State-dependent vulnerabilities manifest under specific conditions; they occur as a result of an execution sequence that is a part of the changes that exist during the execution time [18, 15]. CWE-416 true positive rates didn't exceed 20% in the latest state-of-the-art systems [9, 17], as well as the classes on which they are based are state-dependent and hard to detect [10, 16]. The gaps that this work is built on are that the current vulnerability detection approach still struggles with identifying complex state-dependent vulnerabilities that require deep semantic understanding[14, 12]. We are presenting a semantic representation for the

code with a design principle separating features to prevent the known 'representational bottleneck'. Through SDVD, we presented an improvement compared to the chosen baseline with macro-F1=0.914 on BigVul (9,632 balanced samples), and the largest gain has been on CWE-362 (+24.8 pp) and CWE-416 (+14.2 pp).

2 Background and Related Work

Static analysis tools are able to identify patterns that can be described in a syntactic way. However, they produce systematic false negatives on those state-dependent classes [5, 11, 1]. Dynamic methods have been facing limitations because they don't have the ability to cover all the possible paths, which could sometimes lead to path explosion [8], while the rule-based analyzers are unable to scale across function vulnerability patterns [13]. Graph-based detection represents programs as representation graphs instead of tokens and thus avoids token-based linearity. Devign [19] integrates AST (Abstract Syntax Tree), CFG (Control Flow Graph), DFG (Data Flow Graph), and code sequence edges in a GGNN. DeepWukong [5] builds sliced PDG subgraphs for scalability reasons. In [3], the authors have added an interprocedural call graph to MVD, aiming to detect memory safety using edges. Further contributions include BGNN4VD [2], LineVD [7], andDeepVD [16] has been shown in Table 1. As has been mentioned in MVD [3], a freed pointer has been representationally indistinguishable from a valid one in standard DFG encodings, since DFG edges encode value propagation, but they do not encode the runtime validity of the objects that those values refer to. It has been established from empirical review [9] that classes with temporal and concurrency-dependent properties have the lowest true positive rate. SDVD introduces the three relational edges alongside a multi-component node feature design that is dedicated, as will be detailed in the next chapter.
Table 1 has been comparing the state-aware feature coverage across representative methods, covering object lifetime (OL), value-flow context (VC), synchronization (SYNC), multi-component features (Multi-comp.), and state-aware representations (State-aware). Most of the prior methods have been lacking the support across these dimensions, while SDVD addresses all five.

Table 1: State-aware feature coverage across representative detection methods.

Method	OL	VC	SYNC	Multi-comp.	State-aware
Devign [19]	×	×	×	×	×
DeepWukong [5]	×	×	×	×	×
MVD [3]	Partial	×	×	×	Partial
DeepVD [16]	×	×	×	Partial	×
Zhang et al. [17]	×	×	×	×	Partial
SDVD (Proposed)	✓	✓	✓	✓	✓

3 Methodology

We build a heterogeneous program graph $G=(V, E, X)$, which contains both structural edges (AST, CFG, DFG, PDG) and the semantic edges: Object Lifecycle (OL), Value Constraint (VC), and Synchronization (SYNC). As summarized in Table 2, these 3 relation types represent categories of state-dependent program behavior not present in standard structural program graphs. OL edges link allocation, usage, and deallocation sites; VC edges connect branch conditions to guarded operations; and SYNC edges relate concurrent accesses to shared resources. These semantic relations are inspired by static analysis and provide approximate program state reconstruction without full alias-sensitive or path-sensitive analysis.

Table 2: Edge type definitions and primary target CWE categories.

Type	Semantic Meaning	Target CWEs
AST	Syntactic nesting	All
CFG	Control flow sequencing	All
DFG	Value propagation	All
PDG	Control + data dependency	All
OL	Memory lifecycle relations	416, 476, 399
VC	Guard condition relations	119, 125, 787, 190
SYNC	Concurrency dependency relations	362

Figure 3 presents the 6-stage pipeline of the overall work.

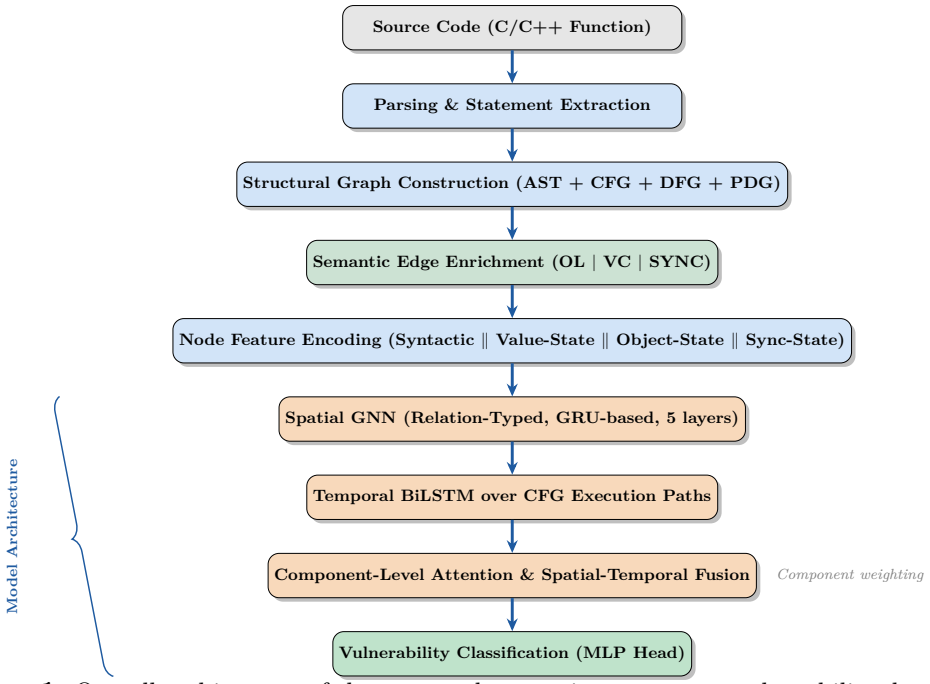


Figure 1: Overall architecture of the proposed semantic state-aware vulnerability detection approach. Shaded blocks indicate the primary representational contributions.

3.1 Semantic Edge Construction

OL edges target CWE-416, CWE-476, and CWE-399. The analysis module searches for allocated memory or resource objects' names through the allocated patterns (`malloc`, `calloc`, and `new`) and builds four directed sub-types, such as allocation-to-use, use-to-release, release-to-use, and null-to-use. A release-to-use edge encodes the `free()` site as a following dereference, which directly is an indicator of the post-free access pattern characteristic of CWE-416. This edge does not exist in the structural DFG, which is a characteristic of CWE-416 called the post-free access pattern.**VC edges** that target CWE-119, CWE-125, CWE-787, and CWE-190. For each array access, the module detects the index relational predicates that have the same module within the dominator and control-flow reachable region leading to the access using the CFG dominance analysis to find guards; in case there are 'no VC edges,' it means a potential unguarded access flaw.**SYNC** targets CWE-362; a SYNC edge joins any two accesses of the same shared variable such that at least one of the accesses is a write and both lie within a `pthread_mutex_lock/unlock` critical section.

3.2 Node Features and Learning Architecture

Each node contains a 176-dimensional feature vector that encodes four components: syntactic component, value-state component, object-state component, and synchronization component. Combining these heterogeneous semantics into just a single node vector leads to representational bottlenecks. Through this four-component design, we are going to address this limitation, as well as enable the component-level attention module to weight each sub-vector's contribution. The model leverages a relation-aware GNN for spatial encoding, a BiLSTM for temporal path encoding, and a fully-connected bidirectional graph network for the text model, along with the component-level attention mechanism to combine semantic representations.

3.3 Model Equations

Message Passing For relation r at layer l:

$$\mathbf{m}_{j\to i}^{(l,r)} = \mathbf{W}_r^{(l)}\mathbf{h}_j^{(l-1)}, \quad \mathbf{h}_i^{(l)} = \mathrm{GRU}\Big(\mathbf{h}_i^{(l-1)}, \sum_r \sum_{j\in\mathcal{N}_i^r} \mathbf{m}_{j\to i}^{(l,r)}\Big) \tag{1}$$

where $\mathbf{h}_i^{(l)}$ is hidden state, $\mathbf{W}_r^{(l)}$ is the relation-specific ($r \in \{\text{AST,CFG,DFG,PDG,OL,VC,SYNC}\}$), and $\mathcal{N}_i^r$ are the neighbors under r. Messages are aggregated, then GRU makes the updates.
Component Attention Weight for component c:

$$\alpha_{i,c} = \frac{\exp(\mathbf{q}_c^T \tanh(\mathbf{W}_c\mathbf{s}_i))}{\sum_{c'}\exp(\mathbf{q}_{c'}^T \tanh(\mathbf{W}_{c'}\mathbf{s}_i))} \tag{2}$$

with $\mathbf{s}_i$ as node i's spatial embedding, $c \in \{\text{syn,val,obj,sync}\}$. $\mathbf{W}_c$ projects, $\mathbf{q}_c$ scores, softmax normalizes.
Fusion Gate

$$\gamma = \sigma(\mathbf{W}_g[\mathbf{h}_{\text{att}};\mathbf{t}]), \quad \mathbf{h}_{\text{fused}} = \gamma \odot \mathbf{h}_{\text{att}} + (1-\gamma)\odot\mathbf{t} \tag{3}$$

where $\mathbf{h}_{\text{att}}$ is attention output and $\mathbf{t}$ the BiLSTM temporal path. Concatenation is gated via the sigmoid to blend elementwise into $\mathbf{h}_{\text{fused}}$.

3.4 Implementation

The GNN has been using 5 layers with d_h=128, BiLSTM using d_{lstm}=64, node features are 176-dim (syntactic 128, value-state 16, object-state 16, sync 16). Training: weighted cross-entropy loss, Adam ($\text{lr} = 10^{-3}$, weight decay $= 10^{-4}$), size of batch 64, gradient clipping 1.0, early stopping patience is 10, max 100 epochs for 5 run with different random seeds and reported the mean performance. All experiments run on an NVIDIA RTX 5060 GPU (8GB VRAM) with an average of 1 hour per run, the model overall had a total of **[1.9]M trainable parameters**.

3.5 Complexity Analysis

Graph construction $O(N^2)$ has been capped at 500, GNN $O(L(Ed_h + Nd_h^2))$, BiLSTM $O(KPd_{in}d_{lstm})$, fusion $O(Nd_h^2)$ with $E{\le}1000$, $K{\le}15$, $P{\le}50$. Forward pass avg 8.65ms over a 30 BigVul samples (N=4–300). Empirical log-log scaling ≈ 0.08, runtime has been fixed-overhead dominated, not N.

4 Experimental Evaluation

We test SDVD on the BigVul dataset [9] for eight CWEs; we had a 70/15/15 split across projects, where the percentages represent the amount of data that is either project-level or categories preventing data leakage. The main baselines that we compare against are Devign, MVD, and WAVES with the F1-score metric. The ablation variants disable one of the following semantic edges: NO_OL, NO_VC, and NO_SYNC. All configs execute 5 runs; results have been presented as mean and standard deviation (STD) with deterministic seeds (times). Selected CWEs and dataset statistics are shown in Table 3.

Table 3: CWE selection and balanced dataset statistics.

Dimension	CWE	Vul.	Total Balanced
Spatial	CWE-119 (Buffer Overflow)	2,127	4,254
	CWE-125 (OOB Read)	625	1,250
	CWE-787 (OOB Write)	198	396
	CWE-190 (Integer Overflow)	307	614
Temporal	CWE-416 (Use After Free)	330	660
	CWE-476 (NULL Ptr Deref.)	215	430
Shared-State	CWE-362 (Race Condition)	278	556
	CWE-399 (Resource Mgmt.)	736	1,472
Total	8 CWEs	4,816	9,632

4.1 Overall Performance

Table 4 presents overall results by the SDVD, where it achieves F1 of 0.914 (±0.003) and AUC 0.955 (±0.003), outperforming all baselines under the same conditions, showing a significant margin. The increase is not even; it's across all. Showing that SDVD has improved overall compared to the known approach Devign with 17.2%, on the other side, compared to MVD with an enhancement of 24.9%, and with the newest baseline WAVES more than 10%.

Table 4: Overall performance, mean ± std (five runs) in all eight CWE categories.

Method	Accuracy	Recall	F1	AUC
Devign [19]	0.784 ± 0.010	0.768 ± 0.028	0.780 ± 0.007	0.855 ± 0.006
MVD [3]	0.706 ± 0.006	0.806 ± 0.044	0.732 ± 0.008	0.780 ± 0.007
WAVES [10]	0.828 ± 0.004	0.836 ± 0.014	0.830 ± 0.003	0.896 ± 0.002
SDVD (Proposed)	$\mathbf{0.909 \pm 0.003}$	$\mathbf{0.967 \pm 0.008}$	$\mathbf{0.914 \pm 0.003}$	$\mathbf{0.955 \pm 0.003}$

4.2 Per-CWE Performance

The SDVD solution improves on the highest number of CWE findings, 37.9%, over the best baseline in CWE-362, and for CWE-416, an improvement of 19.1% over WAVES. Grouping by vulnerability dimension, SDVD has achieved the largest improvement on shared-state vulnerabilities (CWE-362 and CWE-399), which increased the average F1-score from 0.707 by the best baselines to 0.893, which corresponds to a 26.2% improvement. Regarding the temporal vulnerabilities (CWE-416 and CWE-476), SDVD made an improvement for the average F1-score from 0.756 to 0.853 (+12.9%), while for the spatial vulnerabilities (CWE-119, CWE-125, CWE-787, and CWE-190), the average F1-score has been increased from 0.740 to 0.835 (+13.0%).To rule out the possibility that a performance gain is a result of the increased model capacity or the graph density rather than the semantic enrichment. We compare the alignment for each of these edge types. and vulnerability class of the target. The improvements that have been observed are consistently concentrated in CWE

categories that correspond to the relational meaning of the introduced edges (OL → CWE-416/476, VC → CWE-119/125/787, SYNC → CWE-362).Table 5 and Figure 2 report full per-CWE results.The per-CWE evaluation is critical since aggregate measures that add up all CWEs do not make sense; categories might hide class-specific representational differences, and cases might hide. Figure 1 shows the proposed representation and the advantage that it offers, a one-dimensional improvement over another.

Table 5: Per-CWE F1 (mean, 5 runs) The best values are highlighted in bold.

Method	119	125	787	190	416	476	362	399
Devign [19]	0.716	0.693	0.679	0.700	0.685	0.751	0.625	0.759
MVD [3]	0.716	0.701	0.710	0.707	0.683	0.739	0.655	0.752
WAVES [10]	0.787	0.744	0.708	0.717	0.743	0.768	0.628	0.751
SDVD (Proposed)	**0.929**	**0.877**	**0.796**	0.739	**0.885**	**0.821**	**0.903**	**0.882**

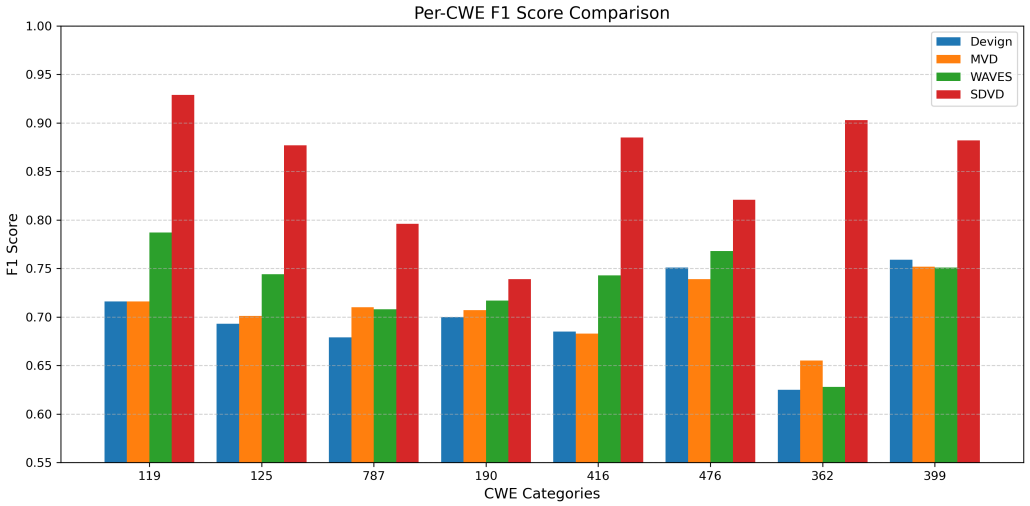


Figure 2: Comparison of per-CWE F1 scores for all methods. SDVD (red) consistently outperforms baselines, with the largest margin on CWE-362 (race conditions) and CWE-416 (use-after-free).

4.3 Ablation Study

SYNC edges are most important for CWE-362, while the OL edges improve CWE-416 detection; regarding VC edges, they improve detection of CWE-125. The node sub-vector partially covers up for the missing OL edges, meaning that a redundant and complementary signal is given to the edge by the four-component node design. Importantly, the performance degradation under edge removal is not proportional across all classes, but selectively impacts the vulnerability type associated with the removed relational channel. This behavior contradicts the hypothesis that SDVD benefits from generic feature augmentation; instead, it supports the interpretation that each typed relation encodes a distinct and non-redundant inductive bias aligned with a specific class of program-state behavior. SDVD significantly has outperformed all baselines ($p < 0.01$, Cohen's $d > 1.6$); ablation variants also showed no significant difference from FULL ($p > 0.25$). NO_OL has been showing +0.004 on CWE-119 ($\Delta{<}1\sigma$), from false-positive OL edges in stack-buffer functions using dynamic allocation, while the NO_SYNC gained +0.030 on CWE-190 because integer overflows rarely involve synchronization; removing SYNC edges eliminates noise.

Table 6: Ablation per-CWE F1 (mean, 5 runs). The top item in each column should be in bold print, indicating the best value.

Config.	119	125	787	190	416	476	362	399
FULL	**0.929**	**0.877**	**0.796**	0.739	**0.885**	**0.821**	**0.903**	**0.882**
NO_OL	0.933	0.864	0.789	0.744	0.883	0.822	0.889	0.882
NO_VC	0.928	0.856	0.795	0.747	0.892	0.825	0.906	0.892
NO_SYNC	0.928	0.858	0.799	**0.769**	0.879	0.824	0.877	0.884

4.4 Threats to Validity

Pattern-based edge extraction could introduce some noise, so gains are a lower bound—this also that's explains the ablation cases where removing an edge type helps via denoising. Evaluation has been limited to BigVul and C/C++; generalization to the other datasets and languages is a future work. We have compared against Devign, MVD, and WAVES (a recent transformer), but broader comparisons with other transformers/LLMs are deferred.

5 Conclusion

SDVD enriched the program graphs with three semantic edge types OL, VC, and SYNC encoding memory lifecycle, guard conditions, and concurrency dependencies. On 8 state-dependent CWEs from BigVul, SDVD achieves F1 of 0.914, significantly outperforming Devign (+17.2%), MVD (+24.9%), and WAVES (+10.1%) with $p < 0.01$ and Cohen's $d > 1.6$. The Largest per-CWE gains have been occurred on CWE-362 (+24.8 pp over WAVES) and CWE-416 (+14.2 pp over WAVES). Ablation confirmed SYNC edges drive the CWE-362 gain and VC edges improve CWE-125, while the four-component node design provides complementary signals when individual edges are removed ($p > 0.25$ vs. FULL). These results demonstrate that explicit semantic state modeling substantially improves detection of vulnerability classes that existing graph-based methods struggle to identify. Future work will be extending the semantic edges to inter-procedural scope and expand modeling to tackle more CWE categories across different datasets.

References

[1] Răzvan-Mihai Bolundut, Adrian-Viorel Coleşa, and Radu-Marian Portase. A path sensitive and scalable approach for mapping the capabilities required for linux kernel defined system calls using static analysis techniques. In *2024 IEEE 20th International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 1–8. IEEE, 2024.

[2] Sicong Cao, Xiaobing Sun, Lili Bo, Ying Wei, and Bin Li. Bgnn4vd: Constructing bidirectional graph neural-network for vulnerability detection. *Information and Software Technology*, 136:106576, 2021.

[3] Sicong Cao, Xiaobing Sun, Lili Bo, Rongxin Wu, Bin Li, and Chuanqi Tao. MVD: Memory-related vulnerability detection based on flow-sensitive graph neural networks. In *Proceedings of the 44th International Conference on Software Engineering*, pages 1456–1468, 2022.

[4] Xiao Cheng, Jiawei Ren, and Yulei Sui. Fast graph simplification for path-sensitive typestate analysis through tempo-spatial multi-point slicing. *Proceedings of the ACM on Software Engineering*, 1(FSE):494–516, 2024.

[5] Xiao Cheng, Haoyu Wang, Jiayi Hua, Guoai Xu, and Yulei Sui. Deepwukong: Statically detecting software vulnerabilities using deep graph neural network. *ACM Transactions on Software Engineering and Methodology*, 30(3):1–33, 2021.

[6] Jinyao Guo, Chengpeng Wang, Xiangzhe Xu, Zian Su, and Xiangyu Zhang. Repoaudit: An autonomous llm-agent for repository-level code auditing. *arXiv preprint arXiv:2501.18160*, 2025.

[7] David Hin, Andrey Kan, Huaming Chen, and M Ali Babar. LineVD: Statement-level vulnerability detection using graph neural networks. In *Proceedings of the 19th International Conference on Mining Software Repositories*, pages 596–607, 2022.

[8] Martin Kellogg, Narges Shadab, Manu Sridharan, and Michael D Ernst. Lightweight and modular resource leak verification. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 181–192, 2021.

[9] Chao Ni, Xin Yin, Xinrui Li, Xiaodan Xu, and Zhi Yu. Abundant modalities offer more nutrients: Multi-modal-based function-level vulnerability detection. *ACM Transactions on Software Engineering and Methodology*, 2025.

[10] Fangcheng Qiu, Zhongxin Liu, Xing Hu, Xin Xia, Gang Chen, and Xinyu Wang. Vulnerability detection via multiple-graph-based code representation. *IEEE Transactions on Software Engineering*, 2024.

[11] Qingkai Shi, Peisen Yao, Rongxin Wu, and Charles Zhang. Path-sensitive sparse analysis without path conditions. In *Proceedings of the 42nd ACM SIGPLAN International Conference on Programming Language Design and Implementation*, pages 930–943, 2021.

[12] Nima Shiri Harzevili, Alvine Boaye Belle, Junjie Wang, Song Wang, Zhen Ming Jiang, and Nachiappan Nagappan. A systematic literature review on automated software vulnerability detection using machine learning. *ACM Computing Surveys*, 57(3):1–36, 2024.

[13] Mingwei Tang, Wei Tang, Qingchi Gui, Jie Hu, and Mingfeng Zhao. A vulnerability detection algorithm based on residual graph attention networks for source code imbalance (RGAN). *Expert Systems with Applications*, 238:122216, 2024.

[14] Asma Touqir, Faisal Iradat, Waseem Iqbal, Abdur Rakib, Nazim Taskin, Hesam Jadidbonab, and Olivier Haas. Systematic exploration of fuzzing in IoT: techniques, vulnerabilities, and open challenges. *The Journal of Supercomputing*, 81(8):877, 2025.

[15] Huanting Wang, Guixin Ye, Zhanyong Tang, Shin Hwei Tan, Songfang Huang, Dingyi Fang, Yansong Feng, Lizhong Bian, and Zheng Wang. Combining graph-based learning with automated data collection for code vulnerability detection. *IEEE Transactions on Information Forensics and Security*, 16:1943–1958, 2020.

[16] Wenbo Wang, Tien N Nguyen, Shaohua Wang, Yi Li, Jiyuan Zhang, and Aashish Yadavally. DeepVD: Toward class-separation features for neural network vulnerability detection. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, pages 2249–2261. IEEE, 2023.

[17] Junwei Zhang, Zhongxin Liu, Xing Hu, Xin Xia, and Shanping Li. Vulnerability detection by learning from syntax-based execution paths of code. *IEEE Transactions on Software Engineering*, 49(8):4196–4212, 2023.

[18] Shunfan Zhou, Zhemin Yang, Dan Qiao, Peng Liu, Min Yang, Zhe Wang, and Chenggang Wu. Ferry: State-aware symbolic execution for exploring state-dependent program paths. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 4365–4382, 2022.

[19] Yaqin Zhou, Shangqing Liu, Jingkai Siow, Xiaoning Du, and Yang Liu. Devign: Effective vulnerability identification by learning comprehensive program semantics via graph neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 273–284.

Performance and Efficiency Analysis for Hybrid PKI Trust Model in Multi-Cloud Environments

R.Radzali[1,3], A.H. Azni[1,2], Farida Ridzuan[1,2] and Nur Hafiza Zakaria[1,2]

[1] Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai 71800, Negeri Sembilan, Malaysia {rozlinda.radzali@raudah,ahazni,farida,mzhafiza}@usim.edu.my
[2] CyberSecurity and System Research Unit, Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai 71800, Negeri Sembilan, Malaysia
{ahazni,farida,mzhafiza}@usim.edu.my
[3] Faculty of Computing and Innovative Technology, New Era University College, Blok B and C, Lot 5, Seksyen 10, Jalan Bukit, 43000 Kajang, Selangor, Malaysia rozlinda.radzali@gmail.com

Abstract. Traditional Public Key Infrastructure (PKI) architectures fall short on crypto-agility and are more susceptible to single points of failure (SPOF) within federated clouds ecosystem. In this paper, we propose a hybrid trust model that combines threshold cryptography and key exchange of short-lived keys to support interoperable communication in a secure environment between disparate Root Certificate Authorities (RCAs). The hybrid trust model, unlike other centralized models, applies (t, n) Shamir's Secret Sharing (SSS) to disperse the signing powers, strategically ensuring the complete master private key is never constructed at one single point in the system. The hybrid trust model also adopts a metadata-binding mechanism (SHA3-256) of 128 bits and combines the use of ephemeral keys of the Diffie-Hellman exchange to provide Perfect Forward Secrecy (PFS). The main strength of this model lies in its algorithmic contribution, which is reinforced by a formal correctness theorem demonstrating the RSA equivalence of the aggregated threshold signature. The model is further supported by an informal threat model, a comprehensive security analysis, and simulation-based verification of its communication complexity, including fixed-threshold stress testing. These enhancements improve the solution's fault tolerance and overall robustness while ensuring compliance with X.509 standards for the centralized distribution of public points. In addition, the proposed approach reduces communication overhead, making it well suited for low-latency cloud environments.
.

Keywords: Threshold Cryptography · Crypto Agility · Hybrid PKI · Communication Complex · Federated Cloud Security · X.509 Interoperability

1 Introduction

While PKI has long been considered the hallmark of digital trust, it has become apparent that conventional, centralized Certificate Authority (CA) architectures are ill-suited for the decentralized and multi-cloud infrastructures that are becoming increasingly common place [2], [16].Centralized certificate management systems are susceptible to malicious CA compromise, and single points of failure further result in system scalability constraints, thereby prompting a need for decentralized trust management systems [2],[16]. In the world of rapidly evolving digital threats, agility of cryptographic systems so that new and lighter cryptographic systems are put in place to replace older, heavier cryptographic systems, has become a necessity for system operators in a federated environment [11].

By combining centralized administrative control and decentralized distributed cryptographic management, this paper outlines an architecture that attempts to resolve the

identified challenges. The Model attempts the defense of Root Certificate Authority (RCA) operations based on threshold cryptography while attempting to balance the trade-off of decentralized trust management and defense in depth opportunities [10]. Combined with the defense in depth opportunities, the framework attempts to balance the management of federated cloud systems, where integrated hybrid flexible trust systems and models are becoming common [15]. This system attempts to balance the encapsulation of ephemeral key and metadata-based defense in depth opportunity, and the trade-off of decentralized trust management based on defense in depth opportunities [7], [5].

2 Related Work

PKI serves as the backbone of trust for secure communication systems. Nevertheless, PKI systems suffer from design issues with scalability and risk of trust segregation, as they rely on centralization of Certificate Authorities (CAs). Recent studies have identified weaknesses of the centralized models of PKI with regards to the distributed and multi-cloud environments and CA liability, as well as the systemic trust bottleneck [2],[16]. To address these limitation, decentralized trust management approaches have been widely explored. Some of these options include the use of Blockchains, or other distributed systems, managing user certificates and managing the trust. Other options have been shown to be resilient to CA attacks, and to be transparent in their certificate validation [16],[15]. Most, however, have been designed as 'lightweight' options, and this has created a use-case in environments that require higher throughput and lower latency, and CA will be the bottleneck in the architecture.

Concurrently, there has been work towards a threshold method of cryptography that seeks to distribute trust among numerous parties. In these schemes, cryptographic operations, such as signing or reconstructing cryptographic keys, are performed in a collaborative manner without any of the participants seeing the complete private key. In the most recent studies, threshold cryptography shows an improvement in the ability to manage distributed keys in a secure manner, and an increase in the tolerance for key-breach related loss scenarios in larger systems [10] [1]. Many of the existing threshold schemes exhibit inefficiencies in environments related to a larger time authentication and key reconstruction loops.

Another major focus area is crypto-agility, which is the flexibility of a cryptographic system to respond to new threats with minimal time and changes to the system. Today's cryptographic systems need to support algorithm-hop agility to maintain security integrity as new threats are introduced. This area of research focuses on the need for extensible cryptographic systems that integrate new components and evolve systems agility without a complete overhaul [11]. Federated cloud environments have introduced a host of new, troublesome scenarios for secure authentication and for the establishment of trust. Proposed to solve the problems of security and trust, these models integrate the distributed pillars of trust to PKI systems. Although in multi-domain cloud systems, assurance of trust and security is in the hands of practitioners, models are proposed to help them delegate and balance trust and control, and to scale security further. The need for more strategies for the support of the shift to new modern secure communication, based on better techniques, cryptographic systems are advocating more forward-secret key approaches and a wider opening of key temporary key shifts [5].

In order to provide a clearer context for this research, we divide prior work into three complementary groups. The first, blockchain-based PKI [2],[16], [14] [15], eliminates the single CA by creating a distributed ledger for certificate transactions. Because the single point of failure is eliminated, the classical CA's consensus latency, and in permission-less frameworks, unpredictable finality, which are unsuitable for the multi-cloud signing operation, replace the classical CA's latency. The second, threshold cryptography and

(t, n) secret-sharing schemes for key management [10][8], ignore the trust of the custodians because the schemes split the private key, but the schemes that we surveyed generally consider a fully peer-to-peer key reconstruction process, which we show quantitatively in Section 5.4, creates, for larger signing groups, an $\mathcal{O}(n^2)$ messaging overhead. The third stream, crypto-agility frameworks, targets migrating cryptographic primitives without a full system rebuild. Recent work here proposes an assessment framework and API design principles for application-level agility, evaluated against libraries such as OpenSSL and cloud key-management services [11]. This targets substitutability within application-facing APIs, not coordination within a federated Root CA hierarchy. No surveyed work, including [11], combines session-level algorithm identifier binding with threshold signature coordination the specific gap the Hybrid PKI Trust Model targets.

3 Research Gaps and Contributions to Design

PKI system lies at the heart of online trust systems, however, with decentralized and multi-cloud infrastructures more prevalent, the traditional centrally placed CA systems tend to fall short of the requirements [2],[16]. The challenges of traditional centrally placed CA systems involve scalability constraints, the centralization challenge of trust and vulnerability to a CA compromise, thereby raising the challenge of trust in conventional PKI systems in modern-day distributed systems [2],[16]. Additionally, in modern federated systems, crypto agility, where the concept of a cryptographic system that can modify itself to adapt to new threats has also become a core requirement [11].

3.1 Research Gaps

In the realm of decentralized trust systems and hybrid cryptographic systems, the following prevalent challenges remain:

- **Required Communication**: Several threshold-based decentralized systems, in the process of secret reconstruction, tend to communicate a lot and in turn, add a considerable amount of overhead to and make distributed systems less efficient [10].
- **Metadata**: The existence of dynamic metadata such as time, context, and an algorithm has not yet been amalgamated into a cryptographic framework and a session, thus, there is limited support for session-level crypto-agility in existing schemes [16].
- **Latency**: A considerable challenge is posed by threshold-based authentication in that the process of verification and trust establishment of the key is inefficient, and especially so in dynamic distributed systems [7].
- **Insufficient Use of Positional Exchange**: The phenomenon of using ephemeral exchanges to provide forward secrecy, becomes less prevalent with the use of threshold-based Root CA and technical exchanges in minimizing the trust gap related to long-term key exposure [5].

3.2 Design Contributions

The aim of this research is to introduce the Hybrid PKI Trust Model, a system that interrelates the centralized administrative coordination system with distributed cryptographic systems.

The framework presents a centralized threshold distribution method built on a (t, n) threshold scheme. In this method, the Root Certificate Authority handles the distribution of signing authorities and keeps the master private key secured from all bound nodes.

The model additionally employs a temporal key derivation system that generates dynamic and forward-secure communication session keys. This system works by combining ephemeral key exchange and SHA3-256 based metadata to secure communications.

The new system architecture improves the distribution of public parameters by decentrally cryptographic signing and reduces communication resources consumed during trust reconstruction. This system will maintain the integrity of authentication, scalability, and security.

A comparative analysis (Theorem 1 in SECTION 4.3) proved that threshold-generated collaborative signatures were indistinguishably equivalent to standard RSA verification and supported legacy X.509-based systems. An efficiency analysis (simulation results in section 5.4) measures the trust restructuring communication resources consumed and demonstrates a significant reduction to the inter-participant communications cryptographic strength of the proposed architecture [10][8].

4 Framework: The Hybrid PKI Trust Model

This model solves the SPOF challenge in conventional PKI by offering a Redundant RCA signing power spread-out architecture. The Hybrid PKI Trust Model builds on a central administrative setup where the RCA oversees the management and split of ownership, while the distribution of signature execution is shared across many trusted nodes. This unified algorithm integrated four critical phases: Metadata Contextualization, Threshold Key Distribution, Collaborative Signature Generation, and Ephemeral Session Establishment.

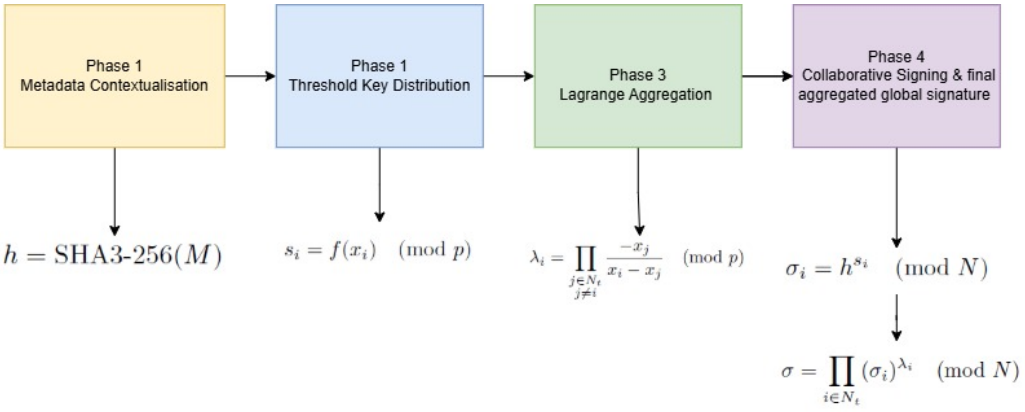


Figure 1: The Hybrid PKI Trust Model — four-phase threshold signing architecture.

4.1 Metadata Contextualization

The proposed process begins by integrating the trust metadata M into a cryptographic hash function to produce a fixed-length digest:

$$h = \text{SHA3-256}(M) \tag{1}$$

This transformation ensures that the output h is of constant length regardless of input size. It also binds the cryptographic operations to a specific context, making the system resistant to replay attacks and message substitution attacks, since any modification in M results in a completely different digest due to the avalanche property of SHA-3.

This mechanism supports a narrower but well-defined form of crypto-agility: algorithm-identifier negotiation. Because *AlgID* is embedded in the metadata $M = AlgID \parallel KeyL \parallel Ver \parallel TS \parallel TP$, and therefore in every session digest h, the RCA and its participating nodes can renegotiate which hash or signature primitive is used on a per-session basis, without touching the master threshold key. Achieving full cryptographic agility for instance, migrating the (t, n) sharing and signing scheme itself to a post-quantum construction is a larger undertaking that requires re-sharing the master secret under the new scheme entirely. That step falls outside the scope of the metadata-binding mechanism described here and is left for future work.

4.2 Threshold Secret Sharing Scheme

Next, a (t, n) threshold secret sharing scheme is applied to distribute signing authority among participating nodes. A random polynomial is constructed over a finite field $\mathbb{F}_p$:

$$f(x) = S + \sum_{j=1}^{t-1} a_j x^j \pmod{p} \tag{2}$$

where S is the master secret, a_j are randomly chosen coefficients, and p is a large prime modulus.

Each node i receives a secret share computed as:

$$s_i = f(x_i) \pmod{p} \tag{3}$$

This ensures that no single node can reconstruct the master secret S. Only a coalition of at least t nodes can participate in valid signature generation, thereby eliminating single points of trust.

4.3 Threshold Signature Generation

During the signing phase, each participating node $i \in N_t$ generates a partial signature using its secret share:

$$\sigma_i = h^{s_i} \pmod{N} \tag{4}$$

where N is the RSA modulus.

To combine these partial signatures, Lagrange interpolation in the exponent domain is applied. The Lagrange coefficient for each node is computed as:

$$\lambda_i = \prod_{\substack{j \in N_t \\ j \neq i}} \frac{-x_j}{x_i - x_j} \pmod{p} \tag{5}$$

The final aggregated global signature is obtained as:

$$\sigma = \prod_{i \in N_t} (\sigma_i)^{\lambda_i} \pmod{N} \tag{6}$$

This construction guarantees that at least t nodes must cooperate, no single node can forge a valid signature, and the final signature is publicly verifiable.

Correctness of Eq. (4)–(6) as written requires that each Lagrange coefficient λ_i be an integer exponent modulo N; this does not hold in general for coefficients computed over $\mathbb{F}_p$ as in Eq. (5). Following the scaled-exponent construction of Shoup [12], we make the equivalence claim precise:

Theorem 1 (Signature Correctness)**.** *Let $\Delta = n!$, and let each node $i \in N_t$ compute the scaled partial signature $\sigma_i = h^{2\Delta s_i} \bmod N$ in place of Eq. (4). Let*

$$\lambda_i = \Delta \cdot \prod_{\substack{j \in N_t \\ j \neq i}} \frac{-j}{i-j},$$

computed as an integer (guaranteed by the Δ scaling). Then

$$w = \prod_{i \in N_t} \sigma_i^{2\lambda_i} \bmod N$$

satisfies $w \equiv h^{4\Delta^2 d} \pmod N$, where d is the RSA private exponent implicitly shared via $f(0) = d$. Since $\gcd(4\Delta^2, e) = 1$ for a suitably chosen public exponent e, the actual RSA signature $\sigma = h^d \bmod N$ is recovered via the extended Euclidean algorithm: find integers a, b with $a \cdot (4\Delta^2) + b \cdot e = 1$, then $\sigma = w^a \cdot h^b \bmod N$.

Proof. By Lagrange interpolation, $\sum_{i \in N_t} \lambda_i \cdot s_i = \Delta \cdot d$ over the integers whenever t nodes agree on distinct evaluation points. Substituting,

$$w = \prod \sigma_i^{2\lambda_i} = \prod h^{4\Delta s_i \lambda_i} = h^{4\Delta \sum s_i \lambda_i} = h^{4\Delta^2 d} \bmod N.$$

The recovery step is the standard RSA-exponent-inversion argument [12], and correctness of σ follows because $(h^d)^e \equiv h \pmod N$ is RSA's defining property, so $\sigma^e \equiv h \pmod N$ for any h coprime to N. This shows the aggregated signature is verifiable as a standard RSA signature by any unmodified X.509/PKCS#1 v1.5 verifier, substantiating the equivalence claim in Sections 1 and 3.2. We validated Theorem 1 computationally for representative (n, t) pairs; a full reduction-based security proof (beyond functional correctness) is left as future work. □

4.4 Forward-Secure Session Key Derivation

Finally, the protocol establishes a forward-secure communication channel using an ephemeral Diffie–Hellman key exchange:

$$K = g^{ab} \pmod p \tag{7}$$

The session key is then derived by binding the shared secret with the previously computed context digest:

$$K_{\text{session}} = \text{SHA3-256}(K \parallel h) \tag{8}$$

This ensures that the session key is both ephemeral and context-aware, thereby providing resistance against replay attacks, key compromise impersonation, and session hijacking.

Algorithm 1 Trust-Aware Threshold Signature and Secure Key Derivation

Input: Metadata M, Threshold (t, n), RSA Modulus N
Output: Global Signature σ, Session Key $K_{session}$

Phase 1: Metadata Hashing
Construct $M = AlgID \parallel KeyL \parallel Ver \parallel TS \parallel TP$
Compute $h \leftarrow SHA3\text{-}256(M)$

Phase 2: Centralized Threshold Distribution
Construct polynomial $f(x) = S + \sum_{j=1}^{t-1} a_j x^j \pmod p$
Distribute shares $s_i = f(x_i) \pmod p$ to n nodes

Phase 3: Collaborative Threshold Signing
for each node $i \in N_t$ **do**
    Compute partial signature $\sigma_i = h^{s_i} \pmod N$
end for
Compute Lagrange coefficient λ_i
Aggregate $\sigma = \prod_{i \in N_t} (\sigma_i)^{\lambda_i} \pmod N$

Phase 4: Session Key Derivation
Compute $K = g^{ab} \pmod p$
Derive $K_{session} = SHA3\text{-}256(K \parallel h)$
return $\sigma, K_{session}$

5 Analysis of Performance and Efficiency

The purpose of this section is analyze the Hybrid PKI Trust Model by examining various curiosities involving the security, latency, and communication issues using the Big O notation. We provide formal and conclusive assurances that our model has addressed the research issues raised in Section 3.

5.1 Formal Communication Complexity Analysis

Reconstruction Phase remains the biggest bottleneck in distributed PKI. We compare our central coordination model to the fully decentralized models proposed by Singh and Chatterjee [14, 3].

1. Decentralized Coordination $\mathcal{O}(n^2)$: In a peer-to-peer threshold scheme, each of the n participants needs to disseminate their individual signature to all the participants to allow for full independent signature reconstruction. In total, this necessitates $n(n-1)$ communications, and communication grows quadratically with n. Hence, the network is at full saturation and the latency grows exponentially.

2. Proposed Hybrid Coordination $\mathcal{O}(t)$: Our approach models the RCA as the primary consolidator. Each node participant, $i \in N_t$, sends only one message (the partial signature σ_i) to the RCA. Thus, a total of t messages are sent. This complexity is adjusted to t, meaning the overall network load is significantly reduced as $t \leq n$.

5.2 Computational Overhead and Verification

We keep track of the number of mathematically expensive operations for each session to evaluate the impact on the processing capacity of our approach.

- **Signature Aggregation ($O(t)$)**: The aggregation scheme specified in Algorithm 1 comprises of t modular exponentiations. Our RSA counterpart model improves the efficiency of processing modular signatures as opposed to pairing schemes [4] with $O(n \log n)$ grouping complexity, which are mostly in the HSMs.

1. **Verification Latency ($O(1)$)**: Output σ is a modular RSA signature. Thus, the end entity can validate the certificate using modular classic PKCS#1 v1.5 logical operations which leads to a constant signature delay. This helps to avoid the complexity of supporting threshold capabilities on systems of an end-user, thus eliminating the "verification barrier."

Big-O complexity analysis indicates that switching from a decentralized peer-to-peer architecture to a centralized star topology will reduce communication overhead from $O(n^2)$ to $O(t)$, which is a significant improvement from a quadratic to a near-linear complexity. This also supports our primary contribution in Section 3.2 , maintaining (t, n) threshold distribution. This is especially beneficial for crypto agility as threshold distribution allows for rapid re-keying and policy changes without network staff. Moreover, with $O(1)$ verification complexity, the system can be deployed without changing the existing client-side verification engines. This also, to a small degree, to show the crypto agility upon changes to the client-side engines.

5.3 Threat Model and Security Analysis

The network channel is modeled under a Dolev–Yao active-adversary assumption: the adversary may intercept, modify, replay, or inject messages at will. A static, threshold-bounded corruption model is also assumed, in which the adversary may fully compromise up to $t - 1$ signing nodes but not the RCA's distribution channel during share issuance, matching the trust assumption underlying the (t, n) scheme in Section 4.2.

Under this model, an informal, assumption-grounded argument can be given, based on the hardness of the RSA problem and the Computational Diffie–Hellman (CDH) assumption, for the following properties.

(i) **Replay resistance.** The timestamp and nonce fields TS, TP inside M (Algorithm 1, line 4) bind h to a specific session. A verifier rejects any h whose TS falls outside an accepted validity window, so a replayed metadata hash cannot be reused to derive a valid session key or signature.

(ii) **Key-compromise impersonation (KCI) resistance.** Session keys are derived from freshly sampled ephemeral DH exponents a, b (Eq. 7). An adversary holding a party's long-term threshold share cannot compute $K = g^{ab}$ without also knowing the ephemeral secret, so compromise of long-term key material alone is not sufficient to impersonate a party in a fresh session.

(iii) **Session-hijacking resistance.** K_{session} is bound to $h = \text{SHA3-256}(M)$ via Eq. (8), so an adversary who did not participate in the metadata-hashing phase for a given session cannot derive a session key consistent with that session's context, even when the DH transcript is observable.

This argument is informal rather than a reduction-based proof. A full game-based security proof, for example in an extended Canetti–Krawczyk or UC-style model, is identified as future work.

5.4 Simulation-Based Validation

Beyond the asymptotic analysis, the proposed hybrid star-topology coordination model was implemented in Python alongside a fully decentralized peer-to-peer baseline, and the number of point-to-point messages required to complete one signature-reconstruction round was measured under each topology. The decentralized baseline requires every node to broadcast its partial signature to every other node ($n(n-1)$ messages); the hybrid model requires only the t coalition members to send their partial signature to the RCA (t messages).

As part of a fixed-threshold complexity test suite, t (the signing-coalition size) was fixed independently of n, sweeping n from 3 to 100 for $t \in \{2, 3, 5\}$. Because the coordination model bounds messaging strictly by t rather than by any function of n, message counts under the hybrid topology remain flat while the decentralized $O(n^2)$ baseline grows unbounded, so the percentage reduction increases monotonically with n and asymptotically approaches 100%. Table 1 reports this for $t = 3$; Figure 2 plots both the absolute message counts and the reduction curves for all three t values.

Table 1: Message counts under the hybrid topology when t is held fixed at 3, independent of n.

n	Decentralized (P2P) msgs	Hybrid msgs ($t = 3$)	Reduction
5	20	3	85.0%
10	90	3	96.7%
25	600	3	99.5%
50	2450	3	99.9%
100	9900	3	100.0%

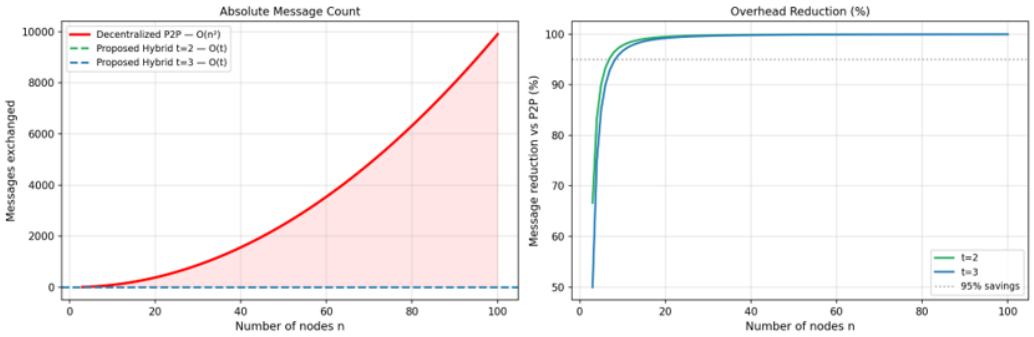


Figure 2: Fixed-t communication complexity test suite. (a) Absolute message counts, decentralized $O(n^2)$ vs. hybrid $O(t)$ for $t \in \{2, 3, 5\}$; (b) percentage message reduction vs. n for each t, approaching 100% as n grows with t fixed.

These measurements corroborate the asymptotic result of Section 5.1: message overhead under the hybrid topology is bounded by t regardless of n, so the relative savings over the decentralized baseline increase without bound as n grows, reaching effectively 100% by $n = 100$ for all tested t. These are simulation-based estimates intended to substantiate the relative scaling behavior described in Sections 5–6, rather than production benchmark numbers from a deployed system; a live multi-cloud testbed deployment is proposed as future work.

6 Discussion

Relative to the two comparison classes discussed in Sections 2 and section 3, the proposed model differs along several axes. On messaging traffic, traditional peer-to-peer secret-sharing schemes incur $O(n^2)$ overhead and blockchain-based PKI incurs consensus-dependent overhead, whereas the hybrid model is bounded by $O(t)$, a result derived analytically in Section 5.1 and corroborated empirically by the fixed-t complexity test suite in Table 1 (Section 5.4). On legacy compatibility, neither traditional secret-sharing nor blockchain-based PKI produces a signature verifiable by unmodified X.509/RSA tooling, whereas the hybrid model does, as established formally by Theorem 1 (Section 4.3). On forward secrecy, only the hybrid model provides it, via the ephemeral Diffie–Hellman exchange of Section 4.4. Computational cost and network latency are comparatively lower for the hybrid model as well, though these two properties are qualitative characterizations, based on the type of operation each class of scheme requires and on the messaging-complexity result respectively, rather than independently benchmarked figures.

The hybrid approach presents a relatively balanced relationship between security and complexity, with a reliance on new technologies and embedding. Big-O notation describes how an algorithm's processing time, or resource consumption, grows with respect to its input [4]. The hybrid model balances system complexity and security through new technology integration. Instead of $O(n^2)$, it achieves $O(t)$ complexity, providing a scalable solution to commonly used flooding protocols, which can cause issues from peer to peer. The hybrid star architecture, unlike other star-based architectures, is near-linear, increasing saturation density of the nodes without degradation of bandwidth, which is important for multi-tenant system validation.

This model also achieves session-level algorithm-identifier agility (see Section 4.1). Incorporating SHA3-256 metadata hashing, per the high-security sponge function FIPS 202 standard [9], trust policies are made immutable with regard to the session-joining process. Due to the constant-level verification complexity of PKCS#1 v1.5 logic [6], legacy systems can accommodate policy updates with no hardware modifications. This means that the threshold-generated signatures are acceptable per standard X.509 verification engines with no additional measures taken on the client (formally, by Theorem 1).

The trade-off between centralization and trust is handled to provide long-lasting resilience. The real signing authority is distributed and implemented through (t, n) shares, while the RCA distributes the coordinates to keep messaging overhead to a minimum. This design addresses the concerns regarding central authority and the Single Point of Failure (SPOF) [13]. For the legacy infrastructures of 2026, this design offers robust, forward-looking agility and performance, demonstrating that distributed cryptographic trust can be combined with administrative efficiency.

7 Conclusion and Future Work

Through the Hybrid PKI Trust Model, this paper demonstrates that communication overhead inherent to decentralized systems can be substantially reduced, while single points of failure in centralized systems can be removed. By separating centralized coordination from the underlying (t, n) threshold signing, this work presents a framework that facilitates distributed cryptographic trust combined with administrative efficiency.

The most important result of this research, the justification of the hybrid architecture, is supported by Theorem 1, showing formally that collaboratively produced threshold signatures are functionally equivalent to standard RSA signatures. Simulation results (Section 5.4) corroborate that shifting from a decentralized to a centralized star topology reduces communication complexity from $O(n^2)$ to $O(t)$ in practice. It was also demonstrated that compliance with PKCS#1 v1.5 logic means the proposed system is compatible with

existing X.509 frameworks, allowing constant $O(1)$-time verification by end entities.

The combination of SHA3-256 metadata binding and ephemeral Diffie–Hellman key exchange preserves the model's forward secrecy and session-context integrity. These characteristics are essential to protecting dynamic, federated cloud environments from existing and future threats. Future work focuses on the following directions:

1. **Resilience to Post-Quantum Computing Threats.** Extending the threshold mechanism to support lattice-based cryptographic systems, including full re-sharing of the master secret under a post-quantum scheme to achieve full (not only algorithm-identifier) crypto-agility.
2. **Mega-Scale Complexity Testing.** Deploying the model in a live global multi-cloud testbed to evaluate latency and throughput under high-concurrency scenarios, extending the simulation of Section 5.4 to production conditions.
3. **Formal, reduction-based security proof.** Extending the informal threat-model argument of Section 5.3 into a full game-based or UC-style security proof.

Acknowledgments

This research was funded by the Ministry of Higher Education (MOHE) of Malaysia under the Fundamental Research Grants Scheme (FRGS/1/2024/ICT07/USIM/02/1).

References

[1] N. A. N. Abdullah, N. H. Zakaria, A. H. Ab Halim, F. H. M. Ridzuan, A. Ahmad, K. Seman, and S. Ariffin. A theoretical comparative analysis of DNA techniques used in DNA-based cryptography. *Journal of Sustainability Science and Management*, 17(5):165–178, 2022.

[2] M. H. Au, A. Kapadia, and W. Susilo. Blockchain-based pki: A survey on decentralized trust and certificate management. *IEEE Communications Surveys & Tutorials*, 25(3), 2023.

[3] S. Chatterjee and P. Roy. Distributed pki frameworks for secure cloud environments. *Journal of Network and Computer Applications*, 178:102956, 2021.

[4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, Cambridge, MA, USA, 4th edition, 2022.

[5] D. Derler, K. Gellert, T. Jager, D. Slamanig, and C. Striecks. Forward-secret key exchange mechanisms for secure distributed systems. *Journal of Cryptology*, 2021.

[6] B. Kaliski and J. Staddon. Pkcs #1: Rsa cryptography specifications version 1.5. Technical Report RFC 2313, RFC Editor, 1998.

[7] S. Khan and M. Al-Rodhaan. Secure federated cloud architectures using hybrid cryptographic trust models. *IEEE Access*, 2024.

[8] S. E. Mohammed, M. O. Al-Dhaifallah, and M. R. Hassan. Performance analysis of secure key management and communication overhead in distributed cryptographic systems. *IEEE Access*, 2022.

[9] National Institute of Standards and Technology (NIST). Sha-3 standard: Permutation-based hash and extendable-output functions. Technical Report FIPS PUB 202, NIST, 2015.

[10] Y. Peng, Q. Feng, D.-B. He, and M. Luo. Survey on threshold cryptography and distributed key management. *Frontiers of Computer Science*, 2026.

[11] Navaneeth Rameshan and Gregoire Messmer. Cryptographic agility for applications: An assessment framework and principled API design. In Daniel Loebenberger and Marc Stöttinger, editors, *Machine Learning for Cyber Security (MAgiCS) 2026*, volume 3002 of *Communications in Computer and Information Science*, pages 165–185. Springer, 2026.

[12] Victor Shoup. Practical threshold signatures. In *Advances in Cryptology — EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 207–220, 2000.

[13] M. S. Siddiqui and C. S. Hong. Security analysis of centralized vs. decentralized pki. *IEEE Communications Surveys & Tutorials*, 26(2):450–472, 2025.

[14] A. Singh and R. Kumar. Blockchain-based distributed public key infrastructure. *IEEE Access*, 8:139939–139949, 2020.

[15] Y. Wang, J. Chen, and H. Li. Decentralized trust management in multi-cloud environments. *IEEE Access*, 2023.

[16] S. Wijethilaka, A. K. Yadav, A. Braeken, and M. Liyanage. Blockchain-based decentralized multi-party certificate management framework. In *IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2023.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 285–305.

A Scoring System for Transport Layer Security

Nur Azrina Abdul Rahman, Muhammad Aiman Hakimi Rosdi, Siti Nurfarahani Ahmad Yusri, Muhammad Hafiz Abu Hassan, Muhammad Saufy Rohmad and Muhammad Reza Z'aba

Malaysia Cryptology and Technology Management Centre (PTPKM)
c/o Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia
{nurazrina,aimanhakimi,nurfarahani,hafizhassan,saufy,rezazaba}@upm.edu.my

Abstract. Transport Layer Security (TLS) is widely used to secure communication between clients and servers over the Internet. Although TLS 1.3 is the latest recommended version, older versions remain supported in practice due to interoperability and legacy compatibility requirements. Consequently, the security of a TLS deployment depends not only on the protocol version, but also on the selected cipher suites and cryptographic algorithms. In this paper, we propose a quantitative scoring framework for evaluating the security strength of TLS configurations on web servers. The framework considers factors such as supported TLS versions, symmetric and asymmetric cipher suites, preferred cipher suites, post-quantum readiness, and conformance to recognised cryptographic standards. Our approach differs from the SSL Labs SSL Server Rating Guide (SRG) by incorporating a more granular assessment of cryptographic primitives and algorithm suitability.

Keywords: TLS · Cryptographic Protocol · Scoring

1 Introduction

In the modern digital era, the widespread use of Internet-based services such as online banking, e-commerce, cloud computing, and electronic communication has made secure communication a fundamental requirement. Transport Layer Security (TLS) is a widely used cryptographic protocol designed to provide secure communication over computer networks [53]. It provides confidentiality through encryption, integrity through message authentication, and authentication via digital certificates. TLS is widely deployed across applications including secure web browsing (HTTPS), email systems, and cloud services.

TLS is the successor to the Secure Sockets Layer (SSL) protocol, originally developed by Netscape Communications in the mid-1990s to enable secure data transmission over the Internet, particularly for e-commerce applications. SSL 1.0 was never publicly released [54, §1]; it was followed by SSL 2.0 in 1995 and SSL 3.0 in 1996 [28]. The protocol was subsequently standardised by the Internet Engineering Task Force (IETF), leading to the development of TLS 1.0 in 1999 [21], TLS 1.1 in 2006 [22], TLS 1.2 in 2008 [23], and the latest version, TLS 1.3, in 2018 [53].

TLS has been extensively analysed and shown to be vulnerable to various attacks across different versions [57, 32, 40]. For TLS 1.2 and earlier, these include timing attacks [4], downgrade attacks to weak Diffie–Hellman groups [1], cross-protocol attacks leveraging SSL 2.0 to break TLS [6], the use of insecure cipher suites [3, 41, 13], and various implementation flaws [11]. TLS 1.3 has likewise undergone extensive analysis, both during its standardisation and after deployment. During its development, the handshake protocol [24, 34, 18, 37, 27] and record layer [7, 20] were rigorously examined. Subsequent work has continued to explore aspects such as session ticket implementations [31],

the handshake protocol [25], privacy considerations [5, 12] and post-quantum security considerations [17, 55, 56, 65, 60].

Although legacy protocols such as SSL 3.0 and TLS 1.0/1.1 have been deprecated in 2015 [9] and 2021 [42] respectively, many websites continue to support older versions and suboptimal configurations for backward compatibility. Even when modern protocols such as TLS 1.3 are deployed, weak cipher suite selection can significantly reduce the overall security level. This heterogeneity in TLS deployments highlights the need for a quantitative approach to evaluate security strength. A scoring system enables consistent comparison across websites, identification of weak configurations, and prioritisation of remediation efforts. While several website evaluation tools exist [62], to the best of our knowledge, only Qualys SSL Server Rating Guide (SRG) [51] provides an assessment focused specifically on TLS configurations.

In this paper, we propose a new TLS scoring approach to evaluate the security strength of website configurations. Our method focuses on key parameters, including supported TLS protocol versions and cipher suite configurations, as primary factors influencing overall security strength. Our approach differs from the SRG methodology by incorporating a more granular assessment of supported cipher suites, accounting for server-side cipher preference, and considering additional factors such as post-quantum readiness and conformance to relevant cryptographic standards. The proposed framework aims to provide a practical and objective tool for administrators and security practitioners to assess and improve TLS deployments.

This paper is organised as follows. Section 2 provides background on TLS and outlines the SRG scoring methodology. Section 3 presents the proposed scoring framework, while Section 4 demonstrates its application. Section 5 applies the SRG methodology to the same case study and compares the results with those of the proposed framework. Finally, Section 6 summarises the main observations and conclusions.

2 Background

This section provides relevant information related to TLS, cryptography standards and the SSL Server Rating Guide (SRG).

2.1 Transport Layer Security

In this paper, we focus on TLS configurations deployed on web servers. In practice, servers typically support multiple TLS versions simultaneously, such as TLS 1.2 and TLS 1.3, to maintain backward compatibility with older clients. During connection establishment, version negotiation selects the highest mutually supported version between the client and server. Each TLS version defines a set of cipher suites specifying the cryptographic algorithms used by the protocol.

Cipher suites are commonly represented using either the Internet Assigned Numbers Authority (IANA) naming convention (e.g., `TLS_RSA_WITH_AES_256_GCM_SHA384`) or the OpenSSL convention (e.g., `AES256-GCM-SHA384`). The naming structure of cipher suites differs between TLS versions. In TLS 1.2 and earlier, cipher suite names explicitly specify the key exchange mechanism, authentication method, encryption algorithm, and hash function. In contrast, TLS 1.3 simplifies the design by separating key exchange negotiation from the cipher suite itself. Consequently, TLS 1.3 cipher suites only specify the symmetric encryption and hash algorithms, such as in `TLS_AES_256_GCM_SHA384`.

For each supported TLS version, the server designates one of the supported cipher suites as the preferred cipher suite. If the client supports the server's preferred cipher suite, that suite is selected for the connection.

Table 1: SRG score classification

Score Range	Grade
≥ 80	A
≥ 65	B
≥ 50	C
≥ 35	D
≥ 20	E
< 20	F

In our scoring system, each supported cipher suite is assigned a score based on several factors, including the specific algorithms used and their inclusion in recognised cryptographic standards.

2.2 Cryptography Standards and Recommendations

Numerous cryptographic standards and recommendations have been developed by organisations such as the National Institute of Standards and Technology (NIST), the International Organisation for Standardisation / International Electrotechnical Commission (ISO/IEC) and Japan's Cryptography Research and Evaluation Committees (CRYPTREC). These standards specify a list of cryptographic algorithms that are deemed secure, which includes block ciphers, hash functions, and digital signature schemes.

However, the algorithms deployed within real-world protocols do not necessarily align with the latest recommendations or approved standards. In practice, protocols may continue to support legacy or deprecated algorithms for reasons such as backward compatibility and interoperability. For example, TLS 1.2 historically supported the RC4 stream cipher despite known security weaknesses and subsequent recommendations against its use [50].

As a case study, this paper uses Malaysia's National Trusted Cryptographic Algorithm List (MySEAL) as a reference framework for assessing the suitability of cryptographic algorithms used in TLS configurations within the proposed scoring system. Within the MySEAL framework, a cryptographic algorithm is categorized as Approved, Neutral, or Monitored. If an algorithm does not fall under any of these categories, it is designated as Unlisted. Annex A compares the recommendation status of selected TLS algorithms across MySEAL and other cryptographic standards and recommendation frameworks.

2.3 SSL Server Rating Guide (SRG)

The SSL Server Rating Guide (SRG), developed by Qualys SSL Labs, is a framework for evaluating SSL/TLS server configurations. Last updated on 16 May 2025 [51], the methodology assesses the overall security of a server by validating its certificate and analysing its TLS configuration across three main categories with the following weightage:

1. Protocol support 30%,
2. Key exchange 30%, and
3. Cipher support 40%.

Each category is assigned a numerical score based on the strength and security of the supported configurations. These scores are combined to produce an overall score ranging from 0 to 100. If any category receives a score of zero, the overall score is also set to zero. The final score is then mapped to a letter grade according to Table 1.

After computing the numerical score, the SRG applies additional rules to account for configuration issues and vulnerabilities not reflected in the score itself. These rules may reduce the grade (e.g., from A to B or F) or increase it up to A+ for strong security practices. In certain cases, alternative grades are assigned, such as M for certificate name

mismatch and T for untrusted certificates, where the standard grading scale is no longer meaningful.

2.3.1 Certificate Inspection

The server certificate is often the weakest component of a TLS configuration. If a certificate is untrusted, it cannot prevent man-in-the-middle (MITM) attacks, effectively rendering TLS protection ineffective. Other certificate issues, such as expiration or incorrect details, also undermine trust and weaken overall security. Consequently, any certificate-related issue results in a score of zero.

2.3.2 Protocol Support

The first aspect evaluated is the protocols supported by a TLS server. Older protocols, such as SSL 2.0 and SSL 3.0, are known to have critical vulnerabilities. Since a server may support multiple protocols simultaneously, the SRG calculates the protocol score using the following formula:

$$\text{protocol score} = \frac{\text{best protocol} + \text{worst protocol}}{2}$$

Table 2 shows the protocol support rating guide and the corresponding scores for each version of the protocol.

Table 2: Protocol support rating guide

Protocol	Score
SSL 2.0	0 %
SSL 3.0	80 %
TLS 1.0	90 %
TLS 1.1	95 %
TLS 1.2	100 %
TLS 1.3	100 %

2.3.3 Key Exchange

In TLS, the key exchange phase has two primary functions. First, it performs authentication, allowing at least one party to verify the identity of the other. Second, it ensures the secure generation and exchange of the secret keys that will be used for the remainder of the session. Table 3 shows the scores assigned for the key exchange portion of the rating.

Table 3: Key exchange rating guide

Key exchange aspect	Score
Weak key (Debian OpenSSL flaw)	0 %
Anonymous key exchange (no authentication)	0 %
Key or DH parameter strength $<$ 512 bits	20 %
Exportable key exchange (limited to 512 bits)	40 %
Key or DH parameter strength $<$ 1024 bits	40 %
Key or DH parameter strength $<$ 2048 bits	80 %
Key or DH parameter strength $<$ 4096 bits	90 %
Key or DH parameter strength $\geq$ 4096 bits	100 %

2.3.4 Cipher Strength

To compromise a communication session, an attacker may attempt to break the symmetric cipher used to encrypt most of the transmitted data. Stronger ciphers provide stronger encryption and therefore require greater effort to break. Since a server may support multiple ciphers with different security levels, a scoring system is used that penalises the use of weaker ciphers. The score for this category is calculated using the following formula:

$$\text{cipher strength score} = \frac{\text{strongest cipher} + \text{weakest cipher}}{2}$$

Table 4 shows the cipher strength rating guide and the corresponding scores for each cipher strength.

Table 4: Cipher strength rating guide

Key length	Score
0 bits (no encryption)	0 %
$<$ 128 bits	20 %
$<$ 256 bits	80 %
$\geq$ 256 bits	100 %

2.3.5 Final Score

The final SRG numerical score is computed as:

$$\text{final score} = \text{protocol score} \times 0.3 + \text{key exchange score} \times 0.3 + \text{cipher strength score} \times 0.4.$$

The resulting score is then mapped to a grade according to Table 1. However, this grade may subsequently be adjusted based on additional considerations described in the next section.

2.3.6 Grading Adjustments and Warnings

In addition to the numerical scoring system, the SRG applies several grading adjustments based on specific vulnerabilities, protocol support, and configuration weaknesses. Certain insecure configurations automatically result in a failing grade (F). Examples include the use of SSL 2.0, insecure renegotiation, weak cryptographic keys, export cipher suites, or vulnerabilities such as Heartbleed [14], DROWN [6], ROBOT [16], and various POODLE-related [41] attacks.

Some weaknesses do not immediately fail the server but instead cap the maximum achievable grade. For instance, supporting outdated protocols such as SSL 3.0 or TLS 1.0 may limit the grade to B. Similarly, the use of weak Diffie–Hellman parameters, RC4 ciphers, or incomplete certificate chains may reduce the achievable grade. Servers that do not support modern security mechanisms such as forward secrecy or authenticated encryption may also receive a lower grade.

SRG also introduces warnings for configurations that are not currently critical but may become security risks in the future. Examples include the use of RC4 with newer TLS versions, lack of forward secrecy, or the absence of secure renegotiation support. When a server has an otherwise strong configuration but still triggers one or more warnings, the grade may be reduced to A-.

An exceptional grade of A+ is awarded to servers that demonstrate strong security configurations without warnings and implement additional protections such as HTTP Strict Transport Security (HSTS) with a sufficiently long maximum age. These grading adjustments ensure that the evaluation reflects not only current security practices but also preparedness for emerging threats.

3 Proposed Scoring System

This section outlines our proposed scoring method for TLS configurations for a specific website.

3.1 Assessing Symmetric Schemes

To assess symmetric schemes, a cipher suite is divided into its encryption and MAC/hash components. The algorithm in each component is assessed against the MySEAL classification and PQC-readiness. The encryption mode of operation is additionally checked to assess whether it is secure.

3.1.1 Algorithm Standards: MySEAL

As stated in Section 2.2, we make use of MySEAL, which comprises a curated list of vetted cryptographic algorithms and schemes deemed to be secure for use in Malaysia. Algorithms that are listed as Approved receive 2 marks, while algorithms in other categories, including Neutral, Monitored and Unlisted, receive reduced marks. This is shown in Table 5(a) where the MySEAL status score for encryption and hash is denoted by $\mathsf{myseal}_{\mathsf{enc}}$ and $\mathsf{myseal}_{\mathsf{hash}}$, respectively.

Table 5: Scoring of MySEAL status, PQC-readiness and encryption mode of operation for the symmetric component of cipher suites.

Status	Score
Approved	2
Neutral	1
Monitored	0
Unlisted	0

(a) MySEAL status (myseal)

Status	Score
Ready	2
Not Ready	1

(b) PQC Readiness (pqc)

Status	Score
Secure	1
Insecure	0

(c) Mode of Operation (mode)

3.1.2 PQC-Readiness

The United States (US) National Institute of Standards and Technology (NIST) defines five security categories for its post-quantum cryptography (PQC) schemes for each of the following security requirements, listed in order of increasing strength [44, 48]:

- **Category 1**: Key search on a block cipher with a 128-bit key.
- **Category 2**: Collision search on a 256-bit hash function.
- **Category 3**: Key search on a block cipher with a 192-bit key.
- **Category 4**: Collision search on a 384-bit hash function.
- **Category 5**: Key search on a block cipher with a 256-bit key.

Grover's algorithm is conjectured to be capable of performing exhaustive key search on a n-bit key cipher with a time complexity of $\mathcal{O}(2^{n/2})$ [30]. Similarly, the Brassard-Høyer-Tapp (BHT) algorithm is claimed to be able to detect collisions on a n-bit hash function in $\mathcal{O}(2^{n/3})$ time. NIST contends that, due to the sequential nature of these algorithms, the circuit size required to implement them is excessively large [29]. Furthermore, NIST asserts that Category 1 provides adequate protection against quantum attacks [44, 48, §4.A.5].

Despite the aforementioned, ongoing efforts are being made to enhance the implementation of these quantum algorithms (e.g., [35, 33, 39, 58, 38]). Consequently, for the purpose of our scoring, we adopt a conservative approach and assume cryptographic algorithms or schemes that meet the minimum requirements of Category 4 to be considered as PQC-ready,

thereby assigning them a score of 2. Conversely, we assume that the algorithms or schemes are not yet ready and are assigned a score of 1. This is given in Table 5(b) where the score for encryption and hash is denoted by $\mathsf{pqc}_{\mathsf{enc}}$ and $\mathsf{pqc}_{\mathsf{hash}}$, respectively.

3.1.3 Encryption Mode of Operation

TLS v1.2 supports Cipher Block Chaining (CBC) as a mode of operation for block ciphers such as AES. This mode provides confidentiality but does not ensure data integrity or authenticity. Therefore, CBC is typically used together with a message authentication code (MAC) in constructions such as MAC-then-Encrypt, which provide integrity and authenticity in addition to confidentiality. In contrast, Galois/Counter Mode (GCM) and Counter with CBC-MAC (CCM) are modes of operation that natively provide both confidentiality and authenticity. These modes, commonly referred to as authenticated encryption (AE) modes, offer stronger and more robust security guarantees than approaches that combine encryption and authentication as separate mechanisms. In TLS v1.3, only AE modes are allowed, which include ChaCha20-Poly1305 [49].

Based on the above considerations, authenticated encryption (AE) modes are regarded as secure within our scoring framework and are assigned a mode of operation score of 1, i.e. $\mathsf{mode} = 1$. In contrast, the use of CBC mode in TLS has been shown to exhibit several security weaknesses. In particular, vulnerabilities arising from improper padding schemes were disclosed by Vaudenay [63]. Subsequent work demonstrated practical attacks exploiting predictable initialisation vectors in CBC mode to recover sensitive HTTPS authentication tokens [26]. Furthermore, CBC-based constructions are susceptible to padding oracle attacks, where an adversary can distinguish between valid and invalid padding through observable side channels, such as timing differences, ultimately enabling plaintext recovery [4, 2]. In light of these vulnerabilities, CBC is considered insecure in our scoring model and is assigned $\mathsf{mode} = 0$, as summarised in Table 5(c).

3.1.4 Symmetric Cipher Suite Score

Each TLS version supports multiple cipher suites. The score assigned to an individual cipher suite, denoted by $\mathsf{cs}_{\mathsf{sym}}$, is computed as follows

$$\begin{aligned}\mathsf{cs}_{\mathsf{sym}} &= \frac{(\mathsf{myseal}_{\mathsf{enc}} \times \mathsf{mode}) + \mathsf{myseal}_{\mathsf{hash}}}{2} + \frac{(\mathsf{pqc}_{\mathsf{enc}} \times \mathsf{mode}) + \mathsf{pqc}_{\mathsf{hash}}}{2} \\ &= \frac{(\mathsf{myseal}_{\mathsf{enc}} + \mathsf{pqc}_{\mathsf{enc}}) \times \mathsf{mode} + \mathsf{myseal}_{\mathsf{hash}} + \mathsf{pqc}_{\mathsf{hash}}}{2}.\end{aligned}$$

Here, $\mathsf{myseal}_{\mathsf{enc}}$ and $\mathsf{myseal}_{\mathsf{hash}}$ denote the MySEAL status scores assigned to the encryption and hash functions, respectively, whereas mode represents the score assigned to the mode of operation used by the cipher suite. Furthermore, $\mathsf{pqc}_{\mathsf{enc}}$ and $\mathsf{pqc}_{\mathsf{hash}}$ represent the PQC-readiness scores of the encryption and hash functions, respectively. The maximum possible cipher suite score is 4.

3.2 Assessing Asymmetric Schemes

This section explains how our scoring system evaluates asymmetric schemes in TLS. The asymmetric component is divided into key exchange and authentication and each scheme is individually assessed and scored.

3.2.1 Algorithm Standards: MySEAL

The method to evaluate asymmetric schemes under an algorithms standards list is the same as symmetric schemes, as stated in Section 3.1.1. The MySEAL status score for key exchange is denoted $\mathsf{myseal}_{\mathsf{keyx}}$ while the score for authentication is denoted $\mathsf{myseal}_{\mathsf{auth}}$.

3.2.2 PQC-Readiness

Algorithms standardised by NIST for post-quantum security and listed as Approved in MySEAL, including ML-KEM [46], ML-DSA [45], and SLH-DSA [47], are classified as *Ready* and assigned a score of 2. In contrast, traditional asymmetric schemes based on integer factorisation or discrete logarithms, including RSA-PSS, ECDSA, ECDHE, DH, and PSK, are classified as *Not Ready* and assigned a score of 1. Although secure against classical attacks, these schemes are vulnerable to quantum attacks enabled by Shor's algorithm [59].

3.2.3 Security of Key Exchange Schemes

The primary security classification in this evaluation follows the MySEAL guidelines, complemented by analyses of forward secrecy and supporting cryptographic literature. Key exchange mechanisms are mainly assessed based on their ability to provide forward secrecy, ensuring that compromise of a server's long-term private key does not expose previously established session keys.

Ephemeral key exchange mechanisms, such as Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) and DHE, are classified as secure and assigned a security score $\mathsf{sec}_{\mathsf{keyx}}$ of 1 because they generate fresh session-specific keys for each connection. In contrast, static or legacy mechanisms, including RSA, non-ephemeral DH, Kerberos 5, and PSK, are assigned a score of 0 due to their lack of forward secrecy. In such schemes, compromise of the server's long-term private key may enable retrospective decryption of previously recorded traffic, resulting in a harvest-now-decrypt-later (HNDL) risk. The security score for key exchange is depicted in Table 6(a). Additionally, RSA-based key exchange using PKCS #1 v1.5 is vulnerable to attacks such as Bleichenbacher's padding oracle attack [15]. Consequently, RSA PKCS #1 v1.5-based schemes are regarded as obsolete and insecure in this evaluation.

The security of asymmetric cryptographic schemes also depends on the hardness of their underlying mathematical problems, such as the Elliptic Curve Discrete Logarithm Problem (ECDLP) and Integer Factorisation Problem (IFP), as well as secure parameter selection and implementation. For key exchange, this evaluation follows MySEAL-approved DH and ECDH configurations, including Ephemeral–Ephemeral and Ephemeral–Static modes such as C(2e,0s), C(1e,1s), C(1e,2s), and C(2e,2s), without mandating specific elliptic curves [19].

A TLS configuration may support multiple key exchange groups. For TLS 1.3, the proposed scoring system considers all supported groups. For example, a configuration may support the groups secp256r1, x25519, and X25519MLKEM768. Each group is individually evaluated based on its MySEAL conformance, PQC-readiness, and security classification. The final key exchange score is then computed as the average score across all supported groups. Let $\mathcal{G}$ denote the set of TLS 1.3 key exchange groups supported by a domain. The overall key exchange score is computed as:

$$(\mathsf{myseal}_{\mathsf{keyx}} + \mathsf{pqc}_{\mathsf{keyx}}) \times \mathsf{sec}_{\mathsf{keyx}} = \frac{\sum_{i \in \mathcal{G}} ((\mathsf{myseal}_{\mathsf{keyx},i} + \mathsf{pqc}_{\mathsf{keyx},i}) \times \mathsf{sec}_{\mathsf{keyx},i})}{|\mathcal{G}|}.$$

The resulting value is used as the final score for the key exchange component of the cipher suite, and forms part of the final asymmetric cipher suite score described in Section 3.2.5.

Table 6: Scoring of key exchange mechanisms, authentication schemes, and post-quantum readiness for the asymmetric component of cipher suites.

Status	Score
Secure	1
Insecure	0

(a) Key exchange sec_{keyx} and authentication scheme sec_{auth} (same scoring rule applies to both).

Status	Score
Ready	2
Not Ready	1

(b) PQC Readiness (pqc)

3.2.4 Security of Authentication Schemes

Authentication in the TLS handshake relies on digital signature schemes to verify the identities of communicating parties. In this evaluation, RSA-PSS, ECDSA, and EdDSA are classified as secure and assigned a security score $\mathsf{sec}_{\mathsf{auth}}$ of 1, as they are approved under the MySEAL framework. In particular, RSA-PSS provides stronger provable security guarantees and improved resistance against forgery and padding oracle attacks compared to legacy RSA PKCS #1 v1.5 schemes [36, 43, 15]. Consequently, TLS 1.3 mandates RSA-PSS for RSA-based authentication [53].

In contrast, legacy schemes such as RSA PKCS #1 v1.5, DSS, and SHA-1-based RSA variants are classified as insecure and assigned a score of 0 due to weaknesses in their padding constructions and hash functions [15, 61, 64]. Similarly, mechanisms such as Kerberos 5 are not considered sufficiently robust under current cryptographic recommendations.

Modern schemes such as ECDSA and EdDSA are regarded as secure due to their strong security guarantees, efficient performance, and suitability under contemporary cryptographic assumptions [10].

3.2.5 Asymmetric Cipher Suite Score

The asymmetric cipher suite score is determined as follows, taken into account the PQC-readiness, MySEAL status, security of key exchange and authentication as stated in previous subsections:

$$\mathsf{cs}_{\mathsf{asym}} = \frac{(\mathsf{myseal}_{\mathsf{keyx}} + \mathsf{pqc}_{\mathsf{keyx}}) \times \mathsf{sec}_{\mathsf{keyx}} + (\mathsf{myseal}_{\mathsf{auth}} + \mathsf{pqc}_{\mathsf{auth}}) \times \mathsf{sec}_{\mathsf{auth}}}{2}$$

Here, $\mathsf{myseal}_{\mathsf{keyx}}$ and $\mathsf{myseal}_{\mathsf{auth}}$ denote the MySEAL status scores for the key exchange and authentication schemes, respectively. Similarly, $\mathsf{pqc}_{\mathsf{keyx}}$ and $\mathsf{pqc}_{\mathsf{auth}}$ represent their PQC-readiness scores, while $\mathsf{sec}_{\mathsf{keyx}}$ and $\mathsf{sec}_{\mathsf{auth}}$ denote the corresponding security scores.

3.3 Calculating the Overall Domain Score

To determine the final score of an individual symmetric and asymmetric cipher suite, the cipher suite score cs is adjusted according to the TLS version in which the cipher suite is supported. To achieve this, a TLS weighting parameter for version v, denoted by tls_v, is introduced. TLS 1.3 is given a weight of $\mathsf{tls}_{1.3} = 1$, TLS 1.2 is assigned a weight of $\mathsf{tls}_{1.2} = 0.8$, while all other TLS versions are assigned a weight of 0. Form this, the weighted symmetric and asymmetric cipher suite scores are computed as follows:

$$\mathsf{weighted}_{\mathsf{sym}} = \mathsf{cs}_{\mathsf{sym}} \times \mathsf{tls}_v,$$
$$\mathsf{weighted}_{\mathsf{asym}} = \mathsf{cs}_{\mathsf{asym}} \times \mathsf{tls}_v.$$

The weighting factor can be viewed as a penalty for supporting TLS versions other than TLS 1.3. Since TLS 1.2 is still widely used, it is assigned a less severe penalty, whereas older and deprecated TLS versions are assigned a weight of zero. Consequently, each individual cipher suite i is given a weighted score as:

$$\mathsf{weightedcs}_i = \mathsf{weighted}_{\mathsf{sym}} + \mathsf{weighted}_{\mathsf{asym}}$$

Let $\mathcal{C}_v$ denote the set of cipher suites supported by a specific domain under TLS version $v = \{1.3, 1.2, \mathsf{others}\}$. The cumulative score for each TLS version is:

$$\mathsf{cumulative}_v = \sum_{i \in \mathcal{C}_v} \mathsf{weightedcs}_i$$

where the cumulative score for other TLS versions is set to zero, $\mathsf{cumulative}_{\mathsf{others}} = 0$

A preliminary TLS score for version v is computed as:

$$\mathsf{prelim}_v = \frac{\mathsf{cumulative}_v}{2 \times |\mathcal{C}_v|}$$

where $|\mathcal{C}_v|$ denotes the number of cipher suites supported by the domain for TLS version v. Note that this value is multiplied by 2 because every cipher suite contains both symmetric and asymmetric components, each of which is evaluated separately. If the domain does not support TLS version v, i.e. $|\mathcal{C}_v| = 0$, then $\mathsf{prelim}_v = 0$.

A penalty is further introduced when the preferred cipher suite is not the strongest supported one. Let $\mathsf{cs}'_{\mathsf{sym}}$ and $\mathsf{cs}'_{\mathsf{asym}}$ denote the scores of the preferred symmetric and asymmetric cipher suite components, respectively. The penalties for TLS version $v = \{1.3, 1.2\}$ are defined as:

$$\mathsf{penalty}_{v,\mathsf{sym}} = \frac{(4 - \mathsf{cs}'_{\mathsf{sym}})}{4} \times \mathsf{tls}_v,$$
$$\mathsf{penalty}_{v,\mathsf{asym}} = \frac{(4 - \mathsf{cs}'_{\mathsf{asym}})}{4} \times \mathsf{tls}_v.$$

For other TLS versions ($v = \mathsf{others}$), the penalty is set to zero:

$$\mathsf{penalty}_{\mathsf{others,sym}} = \mathsf{penalty}_{\mathsf{others,asym}} = 0.$$

We then calculate the average penalty for TLS version v:

$$\mathsf{penalty}_v = (\mathsf{penalty}_{v,\mathsf{sym}} + \mathsf{penalty}_{v,\mathsf{asym}})/2$$

TLS version v score for the domain:

$$\mathsf{score}_v = \mathsf{prelim}_v - \mathsf{penalty}_v$$

The overall TLS domain score, which incorporates all supported TLS versions, is computed as:

$$\mathsf{score} = \frac{\mathsf{cumulative}_{1.3} + \mathsf{cumulative}_{1.2}}{2 \times (|\mathcal{C}_{1.3}| + |\mathcal{C}_{1.2}| + |\mathcal{C}_{\mathsf{others}}|)} - \mathsf{penalty}_{1.3} - \mathsf{penalty}_{1.2}.$$

Here, $|\mathcal{C}_{\mathsf{others}}|$ denotes the number of cipher suites supported by TLS versions earlier than 1.2. Consequently, support for older TLS versions lowers the overall domain score, with a larger number of legacy cipher suites resulting in a greater reduction. Again, the number of supported cipher suites is multiplied by 2 because both the symmetric and asymmetric components are evaluated separately.

Table 7: Sample cipher suites supported by a domain. Cipher suites shown in bold denote the preferred cipher suites for each TLS version.

TLS	Cipher Suite
v1.3	**TLS_AES_256_GCM_SHA384**
v1.3	TLS_CHACHA20_POLY1305_SHA256
v1.3	TLS_AES_128_GCM_SHA256
v1.2	**ECDHE-ECDSA-AES256-GCM-SHA384**

4 Applying the Scoring System

This section demonstrates the practical application of the proposed TLS scoring framework. The domain and cipher suite configuration used in this section are a constructed example intended to illustrate the calculation procedure. Consider a domain supporting TLS 1.2 and TLS 1.3 with the cipher suites listed in Table 7. The cipher suites shown in bold denote the preferred cipher suite for each TLS version. Assume further that the domain supports secp256r1, x25519 and ffdhe3072 key exchange groups for TLS 1.3. The following subsections describe the steps used to compute the overall TLS domain score. In practice, tools such as sslscan [52] may be used to obtain the list of cipher suites and other TLS parameters supported by a domain.

4.1 Assessing Symmetric and Asymmetric Schemes

The detailed steps to evaluate symmetric and asymmetric cryptography component for each cipher suite are elaborated in Sections 3.1 and 3.2. As an example, the symmetric cipher suite score for TLS_AES_256_GCM_SHA384 is calculated as follows:

$$\begin{aligned} \mathsf{cs}_{\mathsf{sym}} &= \frac{(\mathsf{myseal}_{\mathsf{enc}} + \mathsf{pqc}_{\mathsf{enc}}) \times \mathsf{mode} + \mathsf{myseal}_{\mathsf{hash}} + \mathsf{pqc}_{\mathsf{hash}}}{2} \\ &= \frac{(2+2) \times 1 + 2 + 2}{2} \\ &= 4. \end{aligned}$$

The complete parameters for obtaining the symmetric cipher suite score are illustrated in Table 8. The corresponding asymmetric score for the same cipher suite is computed as:

$$\begin{aligned} \mathsf{cs}_{\mathsf{asym}} &= \frac{(\mathsf{myseal}_{\mathsf{keyx}} + \mathsf{pqc}_{\mathsf{keyx}}) \times \mathsf{sec}_{\mathsf{keyx}} + (\mathsf{myseal}_{\mathsf{auth}} + \mathsf{pqc}_{\mathsf{auth}}) \times \mathsf{sec}_{\mathsf{auth}}}{2} \\ &= \frac{((2+1) \times 1) + ((2+1) \times 1)}{2} \\ &= 3. \end{aligned}$$

The scores for other cipher suites can be similarly obtained.

4.2 Calculating the Overall Domain Score

We then compute the symmetric and asymmetric scores for each cipher suite to obtain the cumulative score for every supported TLS version. For the sample domain, the resulting

Table 8: Computation of the symmetric cipher suite scores for the sample domain.

Cipher Suite	Encryption			Hash		$\mathsf{cs}_{\mathsf{sym}}$
	myseal	pqc	mode	myseal	pqc	
TLS_AES_256_GCM_SHA384	2	2	1	2	2	4
TLS_CHACHA20_POLY1305_SHA256	2	2	1	1	1	3
TLS_AES_128_GCM_SHA256	2	1	1	1	1	2.5
ECDHE-ECDSA-AES256-GCM-SHA384	2	2	1	2	2	4

cumulative scores are as follows:

$$\begin{aligned}\mathsf{cumulative}_{1.3} &= \sum_{i \in \mathcal{C}_{1.3}} \mathsf{weightedcs}_i \\ &= 18.5 \\ \mathsf{cumulative}_{1.2} &= \sum_{j \in \mathcal{C}_{1.2}} \mathsf{weightedcs}_j \\ &= 5.6\end{aligned}$$

The cumulative scores are then used to calculate the preliminary scores for each supported TLS version:

$$\begin{aligned}\mathsf{prelim}_{1.3} &= \frac{\mathsf{cumulative}_{1.3}}{2 \times |\mathcal{C}_{1.3}|} \\ &= 18.5/(2 \times 3) = 3.0833 \\ \mathsf{prelim}_{1.2} &= \frac{\mathsf{cumulative}_{1.2}}{2 \times |\mathcal{C}_{1.2}|} \\ &= 5.6/2 = 2.8\end{aligned}$$

We next compute the penalty scores for the symmetric and asymmetric components of each TLS version. For TLS 1.3, the penalties are:

$$\begin{aligned}\mathsf{penalty}_{1.3,\mathsf{sym}} &= ((4 - \mathsf{cs}'_{\mathsf{sym}})) \times \mathsf{tls}_{1.3})/4 = ((4-4) \times 1)/4 = 0, \\ \mathsf{penalty}_{1.3,\mathsf{asym}} &= ((4 - \mathsf{cs}'_{\mathsf{asym}}) \times \mathsf{tls}_{1.3})/4 = ((4-3) \times 1)/4 = 0.25.\end{aligned}$$

Similarly, for TLS 1.2:

$$\begin{aligned}\mathsf{penalty}_{1.2,\mathsf{sym}} &= ((4 - \mathsf{cs}'_{\mathsf{sym}})) \times \mathsf{tls}_{1.2})/4 = ((4-4) \times 0.8)/4 = 0, \\ \mathsf{penalty}_{1.2,\mathsf{asym}} &= ((4 - \mathsf{cs}'_{\mathsf{asym}}) \times \mathsf{tls}_{1.2})/4 = ((4-3) \times 0.8)/4 = 0.2.\end{aligned}$$

The overall penalty for each TLS version is then computed as the average of the symmetric and asymmetric penalties:

$$\begin{aligned}\mathsf{penalty}_{1.3} &= (\mathsf{penalty}_{v,\mathsf{sym}} + \mathsf{penalty}_{v,\mathsf{asym}})/2 \\ &= (0 + 0.25)/2 = 0.125 \\ \mathsf{penalty}_{1.2} &= (\mathsf{penalty}_{v,\mathsf{sym}} + \mathsf{penalty}_{v,\mathsf{asym}})/2 \\ &= (0 + 0.2)/2 = 0.1\end{aligned}$$

The TLS version-specific scores for the domain are therefore:

$$\begin{aligned}\mathsf{score}_{1.3} &= \mathsf{prelim}_{1.3} - \mathsf{penalty}_{1.3} \\ &= 3.0833 - 0.125 = 2.9583 \\ \mathsf{score}_{1.2} &= \mathsf{prelim}_{1.2} - \mathsf{penalty}_{1.2} \\ &= 2.8 - 0.1 = 2.7\end{aligned}$$

The per-version scores $\mathsf{score}_{1.3}$ and $\mathsf{score}_{1.2}$ provide administrators with a diagnostic view of the security strength of each individually supported TLS version. The overall domain score, computed next, instead aggregates cipher suite performance across all supported versions using a shared denominator, so that a domain's use of multiple TLS versions is reflected in a single combined metric rather than a simple average of the per-version scores. The overall domain score, incorporating cipher suites across all supported TLS versions, is computed as:

$$\begin{aligned}\mathsf{score} &= \frac{\mathsf{cumulative}_{1.3} + \mathsf{cumulative}_{1.2}}{2 \times (|\mathcal{C}_{1.3}| + |\mathcal{C}_{1.2}| + |\mathcal{C}_{\mathsf{others}}|)} - \mathsf{penalty}_{1.3} - \mathsf{penalty}_{1.2} \\ &= \frac{18.5 + 5.6}{2(3+1)} - 0.125 - 0.1 \\ &= 2.7875.\end{aligned}$$

In the next section, we are going to apply the SSL Server Rating Guide (SRG) to the same domain.

5 Applying the SSL Server Rating Guide (SRG)

This section demonstrates the practical application of SRG scoring methodology using the same example domain described in Section 4. The domain supports TLS 1.2 and TLS 1.3 with the cipher suites listed in Table 7. As stated in Section 2.3, the SRG methodology evaluates the server based on three weighted categories: protocol support (30%), key exchange (30%), and cipher strength (40%). Additional grading adjustments are then applied based on vulnerabilities and insecure configurations.

5.1 Protocol Support Evaluation

The protocol evaluation considers both the strongest supported protocol and the weakest still-enabled protocol. In the example domain, TLS 1.3 is the strongest supported protocol and represents modern secure communication standards. TLS 1.2 is the weakest enabled protocol but is still considered secure under current cryptographic guidelines according to the SRG. Since both TLS 1.3 and TLS 1.2 are assigned a maximum score of 100 in the scoring model, the protocol score is computed as:

$$\begin{aligned}\mathsf{protocol\ score} &= \frac{100 + 100}{2} \\ &= 100\end{aligned}$$

Although TLS 1.2 is the weakest enabled protocol in this configuration, it does not reduce the score since it is still considered acceptable and no deprecated protocols are enabled. In contrast, our scoring system penalises configurations that continue to support TLS versions older than TLS 1.3.

5.2 Key Exchange Evaluation

The SRG evaluates key exchange strength based on the effective security level of the negotiated key exchange mechanism, with particular emphasis on forward secrecy.

In the example, the server uses ECDHE (Elliptic Curve Diffie-Hellman Ephemeral) based on the X25519 elliptic curve. X25519 uses a 255-bit elliptic curve. Based on NIST SP 800-57's comparable-strength equivalence table, a 255-bit elliptic curve key such as X25519 provides approximately 128-bit symmetric-equivalent security, comparable to a

3072-bit RSA or finite-field DH key [8]. According to Table 3, key exchange mechanisms at this security strength, below the 4096-bit equivalent level, receive a score of 90%.

Since both TLS 1.2 and TLS 1.3 use ECDHE-based key exchange mechanisms with equivalent security strength, the final key exchange score is computed as:

$$\begin{aligned}\mathsf{key\ exchange\ score} &= \frac{90 + 90}{2} \\ &= 90.\end{aligned}$$

In assessing key exchange schemes, our scoring system provides a more granular evaluation, particularly for TLS 1.3. It considers all key exchange groups supported by a domain and assigns an individual score to each group.

5.3 Cipher Strength Evaluation

Cipher strength is evaluated based on the weakest and strongest supported cipher suites. In this example domain, AES-128-GCM is considered moderately strong, while AES-256-GCM and ChaCha20-Poly1305 are classified as strong modern AE ciphers. Their individual scores are provided in Table 9. The final score is computed as the average of the supported cipher strengths:

$$\begin{aligned}\mathsf{cipher\ strength\ score} &= \frac{100 + 80}{2} \\ &= 90.\end{aligned}$$

Our scoring system also considers key length alongside conformance to recognised standards, which, in this work, are based on MySEAL.

Table 9: Cipher Strength Evaluation

Cipher Suite	Score
AES-256-GCM	100
ChaCha20-Poly1305	100
AES-128-GCM	80

5.4 Overall SRG Numerical Score

The final SRG numerical score is calculated using the weighted formula:

$$\begin{aligned}\mathsf{final\ score} &= \mathsf{protocol\ score} \times 0.3 + \mathsf{key\ exchange\ score} \times 0.3 + \mathsf{cipher\ strength\ score} \times 0.4 \\ &= (100 \times 0.30) + (90 \times 0.30) + (90 \times 0.40) \\ &= 30 + 27 + 36 \\ &= 93\end{aligned}$$

According to Table 1, the score 93 translates into the SRG grade A. Since the example configuration does not contain deprecated protocols, weak cipher suites, certificate issues, or known critical vulnerabilities, no grading penalties are applied. Our scoring system uses a maximum score of 4, with penalties applied when the preferred cipher suites are not the strongest available choices.

5.5 Comparison to Our Scoring System

Using the proposed TLS scoring framework in Section 3, the sample domain analysed in Section 4 achieves an overall score of 2.7875 out of a maximum of 4. In contrast, the same domain receives an A grade under the SRG methodology. Since an A grade corresponds to a minimum score of 80%, this would roughly translate to $0.8 \times 4 = 3.2$ in our scoring scale. However, the two scoring systems are not directly comparable because our framework incorporates additional evaluation factors.

Unlike SRG, our framework explicitly evaluates PQC-readiness. Although the domain employs strong symmetric algorithms such as AES-GCM and ChaCha20-Poly1305, its asymmetric components still rely on traditional schemes such as RSA and ECDHE, which are not fully post-quantum resistant. Furthermore, TLS 1.2 is assigned a lower weighting factor than TLS 1.3. Consequently, the overall score in our scoring system is reduced.

6 Conclusion

In this paper, we proposed a quantitative TLS scoring system for evaluating the security strength of TLS configurations deployed on web servers. The proposed method incorporates a more granular assessment of cryptographic primitives, including symmetric and asymmetric cipher suites, preferred cipher suite selection, forward secrecy, post-quantum readiness, and conformance to recognised cryptographic standards such as MySEAL. The proposed approach also considers the continued presence of legacy and deprecated TLS versions, allowing the scoring system to penalise insecure configurations while still reflecting practical deployment realities, particularly the widespread use of TLS 1.2 for interoperability purposes.

The numerical values presented in this paper are intended to illustrate the proposed scoring framework. For example, the scoring scale may be adjusted to a percentage-based system instead of the maximum score of 4 used in this work. Similarly, the TLS weighting factors may be modified, such as assigning a weight of 0.5 to TLS 1.2 instead of 0.8. The framework may also be adapted to reference standards other than MySEAL such as NIST and CRYPTREC. Ultimately, the proposed approach highlights the key parameters that can be evaluated when assessing the TLS configuration of a domain.

There are several additional factors that could be incorporated into the proposed scoring system. These include certificate quality (e.g., certificate validity, key size, signature algorithm, and trust chain), protocol-level vulnerabilities (e.g., insecure renegotiation, downgrade attacks, and compression-related attacks), and implementation-specific weaknesses. Furthermore, the proposed scoring system could be evaluated on a large and diverse dataset of real-world domains to assess its robustness, validate its effectiveness, and facilitate comparisons with the SRG.

Overall, the proposed scoring system provides a practical and objective method for assessing TLS deployments and identifying weak or outdated configurations. It may assist administrators and security practitioners in benchmarking security posture, prioritising remediation efforts, and supporting migration towards stronger and post-quantum-ready cryptographic configurations.

Acknowledgments

We thank Moesfa Soeheila Mohamad, Amir Hamzah Abd Ghafar, Zahari Mahad, Muhammad Asyraf Asbullah, Muhammad Haziq Amran, and Nureen Iman Jasmeen Abdullah for their valuable discussions and feedback during the early development of the scoring methodology.

Table 10: Comparison of Selected Algorithm

Algorithm	MySEAL	NIST	CRYPTREC
AES 128	Approved	Acceptable	Recommended
AES 256	Approved	Acceptable	Recommended
CHACHA20-POLY1305	Approved	Acceptable	Recommended
SHA 256	Neutral	Acceptable	Recommended
SHA 384	Approved	Acceptable	Recommended
ECDHE	Approved	Acceptable	Recommended
DHE	Approved	Acceptable	Recommended
RSA	Approved	Acceptable	Recommended
KERBEROS	Unlisted	Unlisted	Unlisted

A Other Cryptography Standards and Recommendations

Table 10 compares the recommendation status of selected TLS algorithms according to the MySEAL, NIST, and CRYPTREC cryptographic guidelines. The comparison is intended to assess the consistency of the proposed TLS scoring framework with established national and international recommendations. The comparison reveals a high degree of consistency among the three frameworks. Most of the algorithms, including AES-128, AES-256, ChaCha20-Poly1305, SHA-384, ECDHE, DHE, and RSA, are consistently recommended across the three frameworks, demonstrating a common consensus on their suitability for secure TLS implementations. Kerberos is consistently unlisted across all three frameworks, reflecting agreement that it falls outside current recommendations rather than agreement on its suitability.

The only notable difference is the recommendation status of SHA-256. MySEAL classifies SHA-256 as Neutral, whereas NIST classifies it as Acceptable and CRYPTREC includes it in the e-Government Recommended Ciphers List. Consequently, the proposed TLS scoring framework assigns a slightly lower score to SHA-256 than to the other selected algorithms. However, this difference has only a minor impact on the overall TLS security assessment because SHA-256 continues to be recognized as a secure cryptographic hash function by all three organizations.

Overall, the comparison indicates that the proposed TLS scoring framework, using MySEAL as the reference guideline, can be adapted to other cryptographic recommendation frameworks such as NIST and CRYPTREC while producing largely consistent results. minor variation in the score for SHA-256 reflects differences in the classification methodology adopted by each organization rather than a significant difference in the algorithm's security.

References

[1] David Adrian, Karthikeyan Bhargavan, Zakir Durumeric, Pierrick Gaudry, Matthew Green, J. Alex Halderman, Nadia Heninger, Drew Springall, Emmanuel Thomé, Luke Valenta, Benjamin VanderSloot, Eric Wustrow, Santiago Zanella-Béguelin, and Paul Zimmermann. Imperfect forward secrecy: How Diffie-Hellman fails in practice. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 5–17. ACM Press, October 2015.

[2] Martin R. Albrecht and Kenneth G. Paterson. Lucky microseconds: A timing attack on amazon's s2n implementation of TLS. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part I*, volume 9665 of *LNCS*, pages 622–643. Springer, Berlin, Heidelberg, May 2016.

[3] Nadhem J. AlFardan, Daniel J. Bernstein, Kenneth G. Paterson, Bertram Poettering, and Jacob C. N. Schuldt. On the security of RC4 in TLS. In Samuel T. King, editor, *USENIX Security 2013*, pages 305–320. USENIX Association, August 2013.

[4] Nadhem J. AlFardan and Kenneth G. Paterson. Lucky thirteen: Breaking the TLS and DTLS record protocols. In *2013 IEEE Symposium on Security and Privacy*, pages 526–540. IEEE Computer Society Press, May 2013.

[5] Ghada Arfaoui, Xavier Bultel, Pierre-Alain Fouque, Adina Nedelcu, and Cristina Onete. The privacy of the TLS 1.3 protocol. *PoPETs*, 2019(4):190–210, October 2019.

[6] Nimrod Aviram, Sebastian Schinzel, Juraj Somorovsky, Nadia Heninger, Maik Dankel, Jens Steube, Luke Valenta, David Adrian, J. Alex Halderman, Viktor Dukhovni, Emilia Käsper, Shaanan Cohney, Susanne Engels, Christof Paar, and Yuval Shavitt. DROWN: Breaking TLS using SSLv2. In Thorsten Holz and Stefan Savage, editors, *USENIX Security 2016*, pages 689–706. USENIX Association, August 2016.

[7] Christian Badertscher, Christian Matt, Ueli Maurer, Phillip Rogaway, and Björn Tackmann. Augmented secure channels and the goal of the TLS 1.3 record layer. In Man Ho Au and Atsuko Miyaji, editors, *ProvSec 2015*, volume 9451 of *LNCS*, pages 85–104. Springer, Cham, November 2015.

[8] Elaine Barker. SP 800-57 Part 1 Revision 4: Recommendation for Key Management, Part 1: General. National Institute of Standards and Technology, 1 2016.

[9] Richard Barnes, Martin Thomson, Alfredo Pironti, and Adam Langley. Deprecating Secure Sockets Layer Version 3.0. IETF RFC 7568, June 2015.

[10] Daniel J. Bernstein, Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang. High-speed high-security signatures. In Bart Preneel and Tsuyoshi Takagi, editors, *CHES 2011*, volume 6917 of *LNCS*, pages 124–142. Springer, Berlin, Heidelberg, September / October 2011.

[11] Benjamin Beurdouche, Karthikeyan Bhargavan, Antoine Delignat-Lavaud, Cédric Fournet, Markulf Kohlweiss, Alfredo Pironti, Pierre-Yves Strub, and Jean Karim Zinzindohoue. A messy state of the union: Taming the composite state machines of TLS. In *2015 IEEE Symposium on Security and Privacy*, pages 535–552. IEEE Computer Society Press, May 2015.

[12] Karthikeyan Bhargavan, Vincent Cheval, and Christopher A. Wood. A symbolic analysis of privacy for TLS 1.3 with encrypted client hello. In Heng Yin, Angelos Stavrou, Cas Cremers, and Elaine Shi, editors, *ACM CCS 2022*, pages 365–379. ACM Press, November 2022.

[13] Karthikeyan Bhargavan and Gaëtan Leurent. On the practical (in-)security of 64-bit block ciphers: Collision attacks on HTTP over TLS and OpenVPN. In Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi, editors, *ACM CCS 2016*, pages 456–467. ACM Press, October 2016.

[14] Blackduck, Inc. The Heartbleed Bug, 2025. Accessed: 2026-04-30.

[15] Daniel Bleichenbacher. Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1. In Hugo Krawczyk, editor, *CRYPTO'98*, volume 1462 of *LNCS*, pages 1–12. Springer, Berlin, Heidelberg, August 1998.

[16] Hanno Böck, Juraj Somorovsky, and Craig Young. Return of Bleichenbacher's oracle threat (ROBOT). In William Enck and Adrienne Porter Felt, editors, *USENIX Security 2018*, pages 817–849. USENIX Association, August 2018.

[17] Kevin Bürstinghaus-Steinbach, Christoph Krauß, Ruben Niederhagen, and Michael Schneider. Post-quantum TLS on embedded systems: Integrating and evaluating Kyber and SPHINCS+ with mbed TLS. In Hung-Min Sun, Shiuh-Pyng Shieh, Guofei Gu, and Giuseppe Ateniese, editors, *ASIACCS 20*, pages 841–852. ACM Press, October 2020.

[18] Cas Cremers, Marko Horvat, Sam Scott, and Thyla van der Merwe. Automated analysis and verification of TLS 1.3: 0-RTT, resumption and delayed authentication. In *2016 IEEE Symposium on Security and Privacy*, pages 470–485. IEEE Computer Society Press, May 2016.

[19] CyberSecurity Malaysia. AKSA MySEAL 2.1 Approved. https://mykripto.cybersecurity.my/index.php/services/myseal/myseal-category/aksa-myseal/aksa-myseal-approved, 2024. Accessed: 2026-05-07.

[20] Antoine Delignat-Lavaud, Cédric Fournet, Markulf Kohlweiss, Jonathan Protzenko, Aseem Rastogi, Nikhil Swamy, Santiago Zanella-Béguelin, Karthikeyan Bhargavan, Jianyang Pan, and Jean Karim Zinzindohoue. Implementing and proving the TLS 1.3 record layer. In *2017 IEEE Symposium on Security and Privacy*, pages 463–482. IEEE Computer Society Press, May 2017.

[21] Tim Dierks and Christopher Allen. RFC 2246: The TLS Protocol Version 1.0. IETF RFC 2246, January 1999.

[22] Tim Dierks and Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.1. IETF RFC 4346, April 2006.

[23] Tim Dierks and Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2. IETF RFC 5246, April 2006.

[24] Benjamin Dowling, Marc Fischlin, Felix Günther, and Douglas Stebila. A cryptographic analysis of the TLS 1.3 handshake protocol candidates. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 1197–1210. ACM Press, October 2015.

[25] Benjamin Dowling, Marc Fischlin, Felix Günther, and Douglas Stebila. A cryptographic analysis of the TLS 1.3 handshake protocol. *Journal of Cryptology*, 34(4):37, October 2021.

[26] Thai Duong and Juniano Rizzo. Here Come the XOR Ninjas. Available at http://www.hpcc.ecs.soton.ac.uk/~dan/talks/bullrun/Beast.pdf, May 2011.

[27] Marc Fischlin, Felix Günther, Benedikt Schmidt, and Bogdan Warinschi. Key confirmation in key exchange: A formal treatment and implications for TLS 1.3. In *2016 IEEE Symposium on Security and Privacy*, pages 452–469. IEEE Computer Society Press, May 2016.

[28] Alan Freier, Philip Karlton, and Paul Kocher. The Secure Sockets Layer (SSL) Protocol Version 3.0. IETF RFC 6101, August 2011.

[29] Markus Grassl, Brandon Langenberg, Martin Roetteler, and Rainer Steinwandt. Applying Grover's Algorithm to AES: Quantum Resource Estimates. In Tsuyoshi Takagi, editor, *Post-Quantum Cryptography – PQCrypto 2016*, volume 9606 of *Lecture Notes in Computer Science*, pages 29–43. Springer-Verlag, 2016.

[30] Lov K. Grover. A Fast Quantum Mechanical Algorithm for Database Search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, pages 212–219, New York, NY, USA, 1996. Association for Computing Machinery.

[31] Sven Hebrok, Simon Nachtigall, Marcel Maehren, Nurullah Erinola, Robert Merget, Juraj Somorovsky, and Jörg Schwenk. We really need to talk about session tickets: A large-scale analysis of cryptographic dangers with TLS session tickets. In Joseph A. Calandrino and Carmela Troncoso, editors, *USENIX Security 2023*, pages 4877–4894. USENIX Association, August 2023.

[32] Ralph Holz, Johanna Amann, Olivier Mehani, Mohamed Ali Kâafar, and Matthias Wachs. TLS in the wild: An internet-wide analysis of TLS-based protocols for electronic communication. In *NDSS 2016*. The Internet Society, February 2016.

[33] Zhenyu Huang and Siwei Sun. Synthesizing Quantum Circuits of AES with Lower T-depth and Less Qubits. In Shweta Agrawal and Dongdai Lin, editors, *Advances in Cryptology – ASIACRYPT 2022*, volume 13791 of *Lecture Notes in Computer Science*, pages 614–644. Springer-Verlag, 2022.

[34] Tibor Jager, Jörg Schwenk, and Juraj Somorovsky. On the security of TLS 1.3 and QUIC against weaknesses in PKCS#1 v1.5 encryption. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS 2015*, pages 1185–1196. ACM Press, October 2015.

[35] Samuel Jaques, Michael Naehrig, Martin Roetteler, and Fernando Virdia. Implementing Grover Oracles for Quantum Key Search on AES and LowMC. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology – EUROCRYPT 2020*, volume 12106 of *Lecture Notes in Computer Science*, pages 280–310. Springer-Verlag, 2020.

[36] Jakob Jonsson. Security proofs for the RSA-PSS signature scheme and its variants. Cryptology ePrint Archive, Report 2001/053, 2001.

[37] Hugo Krawczyk and Hoeteck Wee. The OPTLS protocol and TLS 1.3. In *2016 IEEE European Symposium on Security and Privacy*, pages 81–96. IEEE Computer Society Press, March 2016.

[38] Haoyu Liao and Qingbin Luo. Quantum Circuit Synthesis for AES with Low DW-Cost. In Goichiro Hanaoka and Bo-Yin Yang, editors, *Advances in Cryptology – ASIACRYPT 2025*, volume 16246 of *Lecture Notes in Computer Science*, pages 475–507. Springer-Verlag, 2026.

[39] Qun Liu, Bart Preneel, Zheng Zhao, and Meiqin Wang. Improved Quantum Circuits for AES: Reducing the Depth and the Number of Qubits. In Jian Guo and Ron Steinfeld, editors, *Advances in Cryptology – ASIACRYPT 2023*, volume 14440 of *Lecture Notes in Computer Science*, pages 67–98. Springer-Verlag, 2023.

[40] Robert Merget. *On the Security of TLS in the Real World*. PhD thesis, Ruhr-Universitat Bochum, 2022.

[41] Bodo Möller, Thai Duong, and Krzysztof Kotowicz. This POODLE Bites: Exploiting the SSL 3.0 Fallback, 2014.

[42] Kathleen Moriarty and Stephen Farrell. Deprecating TLS 1.0 and TLS 1.1. IETF RFC 8996, March 2021.

[43] Kathleen Moriarty, Burt Kaliski, Jakob Jonsson, and Andreas Rusch. PKCS #1: RSA Cryptography Specifications Version 2.2. IETF RFC 8017, November 2016. https://datatracker.ietf.org/doc/rfc8017.

[44] National Institute of Standards and Technology. Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process. Technical report, 2016.

[45] National Institute of Standards and Technology. Module-Lattice-Based Digital Signature Standard. Technical report, 2024.

[46] National Institute of Standards and Technology. Module-Lattice-Based Key-Encapsulation Mechanism Standard. Technical report, 2024.

[47] National Institute of Standards and Technology. Stateless Hash-Based Digital Signature Standard. Technical report, 2024.

[48] National Institute of Standards and Technology. Security (Evaluation Criteria, December 2025. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/evaluation-criteria/security-(evaluation-criteria).

[49] Yoav Nir and Adam Langley. ChaCha20 and Poly1305 for IETF Protocols. IETF RFC 8439, June 2018.

[50] A. Popov. Prohibiting RC4 Cipher Suites. IETF RFC 7465, 2015. https://datatracker.ietf.org/doc/rfc7465.

[51] Qualys. SSL Server Rating Guide Version 2009r, 2025. Accessed: 2026-04-30.

[52] rbsec. sslscan, 2026. Accessed: 2026-05-09.

[53] Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. IETF RFC 8446, 2018. https://datatracker.ietf.org/doc/rfc8446.

[54] Ivan Ristić. *Bulletproof TLS and PKI*. Feisty Duck, 2022.

[55] Peter Schwabe, Douglas Stebila, and Thom Wiggers. Post-quantum TLS without handshake signatures. In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *ACM CCS 2020*, pages 1461–1480. ACM Press, November 2020.

[56] Peter Schwabe, Douglas Stebila, and Thom Wiggers. More efficient post-quantum KEMTLS with pre-distributed public keys. In Elisa Bertino, Haya Shulman, and Michael Waidner, editors, *ESORICS 2021, Part I*, volume 12972 of *LNCS*, pages 3–22. Springer, Cham, October 2021.

[57] Yaron Sheffer, Ralph Holz, and Peter Saint-Andre. Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS). IETF RFC 7457, February 2015.

[58] Haotian Shi and Xiutao Feng. Quantum Circuits of AES with a Low-Depth Linear Layer and a New Structure. In Kai-Min Chung and Yu Sasaki, editors, *Advances in Cryptology – ASIACRYPT 2024*, volume 15491 of *Lecture Notes in Computer Science*, pages 358–395. Springer-Verlag, 2025.

[59] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th FOCS*, pages 124–134. IEEE Computer Society Press, November 1994.

[60] Douglas Stebila, Scott Fluhrer, and Shay Gueron. Hybrid Key Exchange in TLS 1.3 draft-ietf-tls-hybrid-design-16. IETF draft, 2025.

[61] Marc Stevens, Elie Bursztein, Pierre Karpman, Ange Albertini, and Yarik Markov. The first collision for full SHA-1. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part I*, volume 10401 of *LNCS*, pages 570–596. Springer, Cham, August 2017.

[62] Daan Vansteenhuyse, Victor Le Pochat, Gertjan Franken, and Lieven Desmet. One Does Not Simply Score a Website: Evaluating Website Security Scoring Algorithms. In *IEEE European Symposium on Security and Privacy, EuroS&P 2025*, pages 187–194. IEEE, 2025.

[63] Serge Vaudenay. Security flaws induced by CBC padding - applications to SSL, IPSEC, WTLS... In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 534–546. Springer, Berlin, Heidelberg, April / May 2002.

[64] Xiaoyun Wang and Hongbo Yu. How to break MD5 and other hash functions. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 19–35. Springer, Berlin, Heidelberg, May 2005.

[65] Biming Zhou, Haodong Jiang, and Yunlei Zhao. CPA-secure KEMs are also sufficient for post-quantum TLS 1.3. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024, Part III*, volume 15486 of *LNCS*, pages 433–464. Springer, Singapore, December 2024.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 305–318.

Loss Functions in GAN-Based Audio Steganography: A Systematic Review of Limitations and Performance Trade-Offs

Usman Ibrahim Musa[1], Farida Ridzuan[1,2], A H Azni[1,2] and Nur Hafiza Zakaria[1,2]

[1]Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai, 71800 Negeri Sembilan, Malaysia {4269001@raudah.,farida,ahazni,mzhafiza}@usim.edu.my
[2]CyberSecurity and Systems Research Unit, Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai, 71800 Negeri Sembilan, Malaysia
{farida,ahazni,mzhafiza}@usim.edu.my

Abstract. Audio steganography conceals secret information within audio signals to enable covert communication. Generative Adversarial Networks (GANs) have emerged as a powerful paradigm for audio steganography, with loss functions serving as the central mechanism governing training performance. Unlike prior surveys that catalogue GAN-based steganography architectures in general, this paper narrows its focus specifically to loss function design, presenting a systematic review that analyses limitations and performance trade-offs across three key metrics: imperceptibility, robustness, and embedding capacity. Six loss function categories are examined: adversarial loss, Wasserstein-GP loss, perceptual loss, cycle consistency loss, embedding loss, and multi-adversarial loss. Findings reveal that perceptual loss achieves the highest imperceptibility but remains vulnerable to steganalysis, Wasserstein-GP loss ensures training stability at high computational cost, and multi-adversarial loss advances security without improving embedding capacity. The existing loss functions operate with fixed weighting that does not take into consideration any audio-related parameters, such as SNR (local), Spectral Flatness or Temporal Masking. Future research should be directed to using adaptive/hybrid loss functions that integrate the audio-specific parameters in a dynamic manner to optimise for the three characteristics of imperceptibility, robustness and capacity simultaneously in GAN-based audio steganography systems.

Keywords: Generative Adversarial Networks · Loss Functions · Audio Steganography · Imperceptibility · Robustness · Embedding Capacity

1 Introduction

Audio steganography is a method of embedding hidden messages into audio signals while maintaining an undetectable form of sound. Three important objectives need to be accomplished simultaneously: imperceptibility, robustness, and embedding capacity. Traditional methods such as LSB substitution and phase coding have achieved sufficient levels of imperceptibility while having very little resistance to statistical methods (steganalysis) and signal processing (manipulation). More recent signal-processing-based approaches, such as foreground-background segmentation of the audio signal [2], have partially improved this trade-off, but they remain fundamentally hand-engineered rather than learned. Because of the limitations associated with these types of approaches, researchers are beginning to turn to deep learning approaches and more specifically to generative adversarial networks.

GAN stands for Generative Adversarial Network and consists of 2 neural networks (a generator G and a discriminator D). In audio steganography, the generator learns to incorporate hidden secret messages into cover audio without the sound being perceptible, while the discriminator attempts to identify the presence of hidden secret messages by examining the covert samples. This adversarial relationship enables the use of automatic learning for embedding strategies without the need for manual design or performance coaching, resulting in better detection performance than similar conventional approaches [1, 2]. The loss functions calculate the loss of the discriminator (L_D) and the generator (L_G) based on the accuracy of the classification; all weights are updated iteratively until the generator produces samples indistinguishable from the original samples.

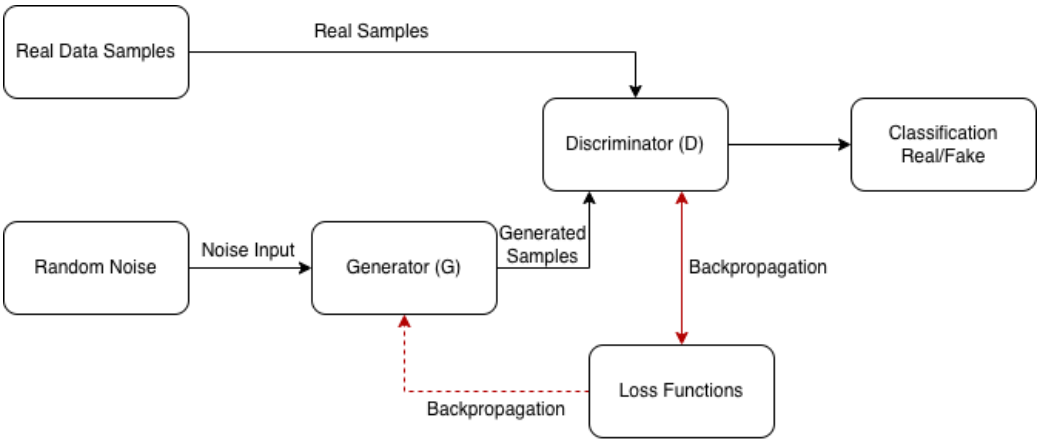


Figure 1: GAN Architecture Process

A strong correlation exists between loss function design and quality of generated outputs. The generator-discriminator competition is driven by adversarial loss, whereas training stability is a result of Wasserstein-GP (Gradient Penalty) loss. Additionally, perceptual loss serves to preserve high-level audio features, while cycle consistency loss guarantees successful bidirectional recovery of messages. Furthermore, embedding loss serves to minimise the bit error rate (BER), while multi-adversarial loss coordinates operation with multiple independent steganalyzers. Despite these numerous advantages, the core limitation present in all current variants is that the weights used in loss functions are defined and set as fixed constants; hence, they cannot be dynamically adjusted according to either audio-specific parameters or training dynamics. This fixed-weighting problem, and its downstream consequences for imperceptibility, robustness, and capacity, is the central thread that motivates and connects every section of this review.

Existing surveys on GAN-based steganography, such as [16], are organised primarily around network architectures and application domains, and treat the loss function as one implementation detail among many. Consequently, no prior work has isolated loss function design as the unit of analysis, mapped the specific limitations of each loss category against the three core audio steganography objectives, or examined how far audio-specific signal properties have actually been incorporated into these formulations. Many recent studies on GAN-based steganography have focused primarily on improving the architecture of GAN models, with little research attention given to how the loss function is designed [3]. The embedding of audio content requires an understanding of unique features such as temporal dependencies, psychoacoustic masking behaviours, and spectral characteristics of sound [4]. Therefore, any existing loss functions from the domain of image steganography should be treated with extreme caution as they may result in less than optimal trade-offs

between embedding quality and detection rate [3, 5]. This study seeks to address this research gap by conducting an in-depth review of the loss functions used for GAN-based audio steganography. The remaining sections are organised as follows: Section 2 covers research objectives and questions; Section 3 describes the review methodology; Section 4 provides the literature review and mathematical formulations; Section 5 offers analysis, discussion, and future directions; and Section 6 concludes this paper.

2 Research Objectives and Questions

The review process is informed by three specific research objectives, each accompanied by a research question, as shown in Table 1. This mapping ensures the logical flow from loss function types to limitations and future research.

Table 1: Research Objectives and Corresponding Research Questions

Research Objectives	Research Questions
RO1: Identify and map loss function types used in GAN-based audio steganography.	RQ1: What loss function types are employed in GAN-based audio steganography research, and how do they relate to GAN training components?
RO2: Analyse the limitations and performance trade-offs of each loss function type.	RQ2: What are the specific limitations of existing loss functions in GAN-based audio steganography, and how do they affect imperceptibility, robustness, and embedding capacity?
RO3: Identify future research directions for adaptive loss function design.	RQ3: What adaptive and audio-specific improvements to loss function design are needed to advance GAN-based audio steganography systems?

3 Review Methodology

This section describes the systematic approach undertaken in identifying, filtering, and selecting the 33 articles used in this review. The process follows the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) reporting structure, comprising identification, screening, eligibility, and inclusion stages, to ensure reproducibility and transparency.

Figure 2 summarises the review's selection process using the PRISMA reporting structure: 120 records identified across the five databases were reduced to 108 after duplicate removal, 63 after title/abstract screening, and a final 33 studies after full-text eligibility and quality assessment, with the reasons for exclusion reported at each stage.

3.1 Search Strategy and Database Selection

A systematic process was followed using five key database repositories: IEEE Xplore, Springer, Elsevier, Wiley, and ArXiv. Boolean search strings combining GAN, loss-function, and steganography-related terms were applied consistently across all five databases (with minor syntax adaptations to match each database's query parser):

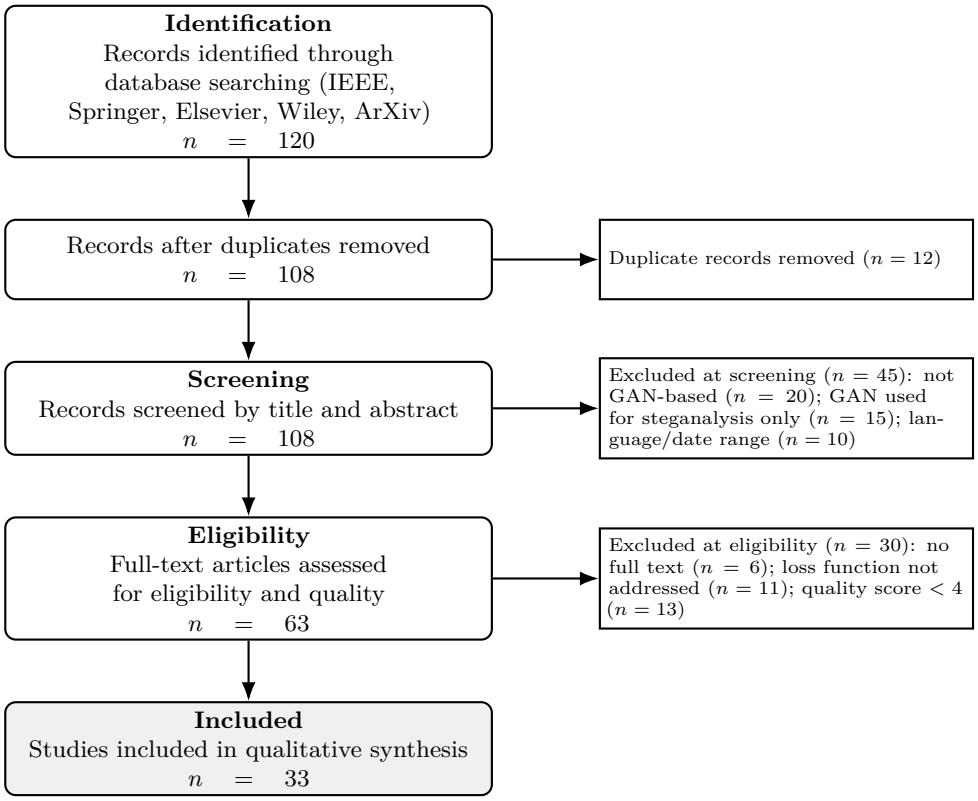


Figure 2: Flow diagram of the identification, screening, eligibility, and inclusion stages of the review

- **String 1:** `("GAN" OR "Generative Adversarial Network*") AND ("audio steganography" OR "speech steganography" OR "audio data hiding")`
- **String 2:** `("GAN" OR "Generative Adversarial Network*") AND ("loss function" OR "objective function") AND ("steganography" OR "data hiding")`
- **String 3:** `("audio steganography" OR "audio data hiding") AND ("imperceptibility" OR "robustness" OR "embedding capacity")`

These strings were applied to titles, abstracts, and keywords, restricted to publications from 2017 to 2026, yielding an initial pool of 120 records after removing obvious cross-database duplicates at the point of export.

3.2 Inclusion and Exclusion Criteria

Inclusion criteria: articles published in peer-reviewed journals or recognized conference proceedings, written in English, covering GAN-based steganography with specific reference to the design or evaluation of the loss function, and available in full in the selected databases. A small number of closely related, highly cited foundational works on GAN loss functions from the image domain and general GAN literature were also included where they were directly cited as the theoretical basis for an audio steganography loss formulation, in order to preserve the mathematical lineage of each loss category.

Exclusion criteria: articles covering traditional (non-GAN-based) steganography, articles using GAN purely for steganalysis without reference to a steganographic embedding

component, duplicate records, articles without full-text access, and articles that did not survive the quality assessment described below.

3.3 Quality Assessment

Each full-text article that passed the eligibility stage was scored against three criteria, each rated on a 0–2 scale (0 = not met, 1 = partially met, 2 = fully met): (i) *venue standing*, indexed journal or recognised conference proceedings; (ii) *citation traction*, normalised by years since publication, to avoid penalising very recent work; and (iii) *methodological clarity*, explicit reporting of the loss function formulation and evaluation protocol. Articles scoring 4 or above (out of 6) were retained. This rubric is deliberately lightweight rather than a validated instrument such as the Cochrane or CASP checklists, acknowledging this as a limitation of the review methodology; it is reported here explicitly so that the process is reproducible rather than left implicit.

3.4 Screening Process

Screening was conducted independently by two of the authors at both the title/abstract and full-text eligibility stages. Disagreements on borderline inclusion/exclusion decisions (fewer than 10% of full-text records) were resolved by discussion with a third author until consensus was reached; no formal inter-rater reliability statistic was computed, which we note as a limitation for future iterations of this review. The overall review process, reported using a PRISMA-style flow diagram, is presented in Figure **??** and Figure 2.

4 Literature Review and Analysis

This section presents the results of the systematic review. It first maps the reviewed articles to their GAN components and loss function types (addressing RO1), then provides the mathematical formulations for each loss function category identified in the review.

4.1 Mapping of Loss Functions to GAN Components (RO1)

Table 2 presents a systematic ordering of all articles according to their domain, primary GAN component, loss function type, performance metrics, and key limitation. Articles are ordered by domain (audio before image before general) and then by year of publication within each domain. The image- and general-domain entries (Table 2, Part 2) are retained deliberately rather than as scope creep: each is the direct theoretical or architectural ancestor of an audio-domain loss formulation discussed in Section 4.2 (e.g., WGAN-GP [11] underlies [18]; cycle consistency [17, 20] underlies the bidirectional recovery formulation in [6, 32]), and they are used exclusively as reference points for the cross-domain comparison in Section 5, never as evidence about audio-domain performance.

Table 2 shows that adversarial loss is the most popular formulation, used in 22 studies (66.7%). Perceptual loss and embedding loss each appear in six studies (18.2%), cycle consistency in four (12.1%), while WGAN-GP and multi-adversarial loss appear in only two studies each (6.1%). Most interestingly, only nine studies (27.3%) focus on the loss function formulation, while 24 (72.7%) simply apply existing loss formulations.

4.2 Mathematical Formulations of Loss Functions

The following subsections provide the mathematical formulations of each of these six classes of loss functions defined in Table 2. Beyond stating each formula, every subsection closes with a short analytical note that ties the equation's structure directly to the performance limitation it produces, so that Eqs. (1)–(6) are read as explanatory devices rather than as

Table 2: Systematic Review of Loss Functions in GAN-Based Audio Steganography (Part 1: Audio Domain)

Ref.	Authors (Year)	Domain	GAN Component	Loss Function	Metrics Evaluated	Key Limitation
[1]	Chen et al. (2021)	Audio	Generator, Discriminator	Adversarial	Imperceptibility, BER	Fixed adversarial weights; no audio-specific parameters integrated
[5]	Yang et al. (2020)	Audio	Generator, Training	Adversarial + Embedding	SNR, Detectability	Embedding loss not adaptive to local SNR variations
[6]	Ye et al. (2019)	Audio	Generator, Discriminator	Adversarial + Embedding	Imperceptibility, Capacity	Three-network architecture; high computational overhead
[7]	Chen et al. (2023)	Audio	Generator, Discriminator	Adversarial + Perceptual (Psychoacoustic)	SNR, PESQ, Imperceptibility	Perceptual loss uses fixed psychoacoustic threshold
[8]	Li et al. (2023)	Audio	Generator, Discriminator	Adversarial + Reconstruction	MOS, Capacity, Steganalysis Resistance	WaveGAN-based synthesis; no perceptual (psychoacoustic) loss
[9]	Feng et al. (2025)	Audio	Generator, Discriminator	Adversarial + Robustness	SNR, BER, Steganalysis Resistance	Differential privacy clustering does not feed back into loss function
[10]	Su et al. (2024)	Audio	Generator, Overall Architecture	Adversarial	Imperceptibility, Steganalysis Resistance	Music-specific model; not generalisable to speech audio
[18]	Yue et al. (2022)	Audio	Generator, Training	Adversarial (BN-SNGAN)	SNR, Detectability	BN optimisation improves stability but not audio quality metrics
[19]	Wang et al. (2020)	Audio	Generator, Discriminator	Adversarial + Distortion	Imperceptibility, SNR	Derivative distortion loss does not model temporal masking
[20]	Moorthy & Venkataraman (2021)	Audio	Generator	Adversarial	Imperceptibility, Capacity	U-LSB limits anti-steganalysis performance
[25]	Duan et al. (2023)	Audio	Overall Architecture	Adversarial + Regularisation	BER, SNR, Quality	Gradient balancing manual; not integrated into loss design
[28]	Helmy & Torkey (2025)	Audio	Generator, Training	Adversarial + Reconstruction	SNR, PESQ, Capacity	Chaotic encryption layer not integrated into GAN loss function
[29]	Chen et al. (2023)	Audio	Generator, Discriminator	Adversarial + Perceptual (Psychoacoustic)	SNR, PESQ, Imperceptibility	Ignores phase and temporal masking effects
[31]	Mahmoud & Elshoush (2022)	Audio	Generator	Adversarial + LSB	Capacity, Transparency	LSB component vulnerable to steganalysis; no GAN loss tuning
[32]	Ying et al. (2021)	Audio	Generator, Training	Adversarial + STC	Imperceptibility, BER	STC-based embedding does not feed back into loss optimisation
[33]	Wu et al. (2020)	Audio	Generator, Training	Adversarial (Iterative)	SNR, Detectability	Iterative adversarial attack loss computationally intensive
[34]	Yan et al. (2026)	Audio	Generator, Latent Optimization	Diffusion-based + Reconstruction + Latent Optimization	BER, Robustness, Compression Resistance	Diffusion inversion increases computational complexity
[35]	Yan et al. (2025)	Audio	Generator, Decoder	Adversarial + Perturbation Optimization	PSNR, Steganalysis Resistance, Capacity	Requires fixed decoder sharing; limited flexibility in deployment
[36]	Zhuo et al. (2024)	Audio	Generator, Training	Adversarial + Reinforcement	SNR, Imperceptibility, Detectability	Does not incorporate psychoacoustic or spectral flatness constraints

Table 3: *

Table 2 continued — Systematic Review of Loss Functions (General and Image Domains)

Ref.	Authors (Year)	Domain	GAN Component	Loss Function	Metrics Evaluated	Key Limitation
[11]	Gulrajani et al. (2017)	General	Loss Function	Wasserstein-GP	Training Stability	Very high computational cost; not audio-adapted
[17]	Saqlain et al. (2021)	General	Loss Function	Multiple GAN losses	Training stability, Quality	Analysis not specific to audio steganography tasks
[12]	Singh et al. (2022)	Image	Loss Function	Perceptual + Adversarial	SSIM, PSNR, Imperceptibility	Perceptual loss computed on image features
[13]	Chen et al. (2024)	Image	Generator, Overall Architecture	Cycle Consistency	Capacity, Security	No robustness modelling under noise attacks
[14]	Huang et al. (2024)	Image	Discriminator, Training	Multi-Adversarial	Steganalysis Resistance, BER	Audio applicability not validated
[15]	Martín et al. (2023)	Image	Generator, Training	Adversarial (Evolved)	PSNR, Steganalysis	Evolutionary search expensive; no audio-domain evaluation
[16]	Liu et al. (2020)	Image	Overall Architecture	Adversarial	Imperceptibility, Security	Architecture review; loss function analysis is secondary
[21]	Tang et al. (2017)	Image	Generator, Loss Function	Adversarial + Embedding	Imperceptibility, Steganalysis Resistance	Fixed per-pixel cost function not adaptive to audio temporal features
[22]	Tan et al. (2022)	Image	Generator	Adversarial	Capacity, Steganalysis	Channel attention increases capacity but no robustness loss
[23]	Peng et al. (2022)	Image	Generator, Overall Architecture	Adversarial	Robustness, Security	No perceptual quality constraint in loss formulation
[24]	Zeng et al. (2023)	Image	Generator, Encoder	Perceptual + Adversarial	PSNR, SSIM	Perceptual loss layer weights are fixed; no audio adaptation
[26]	Yang et al. (2024)	Image	Generator, Overall Architecture	Adversarial + Robustness	Robustness, Security	Robustness loss fixed; no dynamic adaptation to noise type
[27]	Zhou et al. (2025)	Image	Generator, Discriminator	Cycle Consistency	Imperceptibility, Capacity	Cycle loss increases training time; no capacity improvement
[30]	Holub & Fridrich (2012)	Image	Loss Function	Custom Distortion	Steganalysis Resistance	Heuristic cost; not learnable or audio-adapted

a restatement of well-known GAN theory. These formulations provide the foundation for the comparative performance analysis carried out in Section 5.

4.2.1 Standard Adversarial Loss

The standard adversarial loss [4] is formulated as the minimax objective:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] \tag{1}$$

where x represents cover audio, z is the noise vector, p_{data} is the real audio distribution, and p_z is the noise prior [2]. The primary limitation is no explicit constraint on audio perceptual quality, embedding accuracy, or robustness: because Eq. (1) scores $G(z)$ only through D's real/fake decision, any audio-specific distortion that D does not happen to penalise is free for G to introduce, which is precisely why every audio-domain study in Table 2 that relies on adversarial loss alone augments it with a second term.

4.2.2 Wasserstein-GP Loss

The WGAN-GP loss [11] replaces the Jensen-Shannon divergence with the Earth Mover's distance and enforces the Lipschitz constraint via a gradient penalty:

$$L_{\text{WGAN-GP}} = \mathbb{E}_{\tilde{x}\sim P_g}[D(\tilde{x})] - \mathbb{E}_{x\sim P_r}[D(x)] + \lambda \cdot \mathbb{E}_{\hat{x}\sim P_{\hat{x}}}\left[\left(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1\right)^2\right] \tag{2}$$

where λ is the gradient penalty coefficient, P_r is the real audio distribution, P_g is the generated distribution, and $\hat{x}$ is sampled uniformly between real and generated audio pairs. This formulation resolves mode collapse and vanishing gradient issues [11], but the gradient penalty term requires an additional forward and backward pass through D at every interpolated sample $\hat{x}$, which is why the computational cost scales directly with λ's sampling density and makes WGAN-GP impractical for resource-constrained or real-time deployments [18] – a direct structural trade-off between training stability and cost, rather than an incidental implementation detail.

4.2.3 Perceptual Loss

Perceptual loss [13, 16] measures feature-space similarity between stego-audio S and cover audio C using a pre-trained feature extractor φ:

$$L_{\text{perceptual}} = \sum_i \frac{1}{C_i \cdot H_i \cdot W_i} \|\varphi_i(S) - \varphi_i(C)\|_2^2 \tag{3}$$

where φ_i is the feature map at layer i with dimensions C_i, H_i, W_i. The critical limitation is that feature extractor weights are fixed from image-domain pre-training and not adapted to psychoacoustic properties of the human auditory system [9]: because φ is optimised to discriminate visual textures, Eq. (3) penalises spectrogram-level feature mismatch rather than perceptual audibility, so a stego-audio sample can score well under $L_{\text{perceptual}}$ while still being distinguishable by a steganalyzer trained on raw waveform statistics – which is exactly the disconnect Feng et al. [9] report empirically.

4.2.4 Cycle Consistency Loss

Cycle consistency loss [17, 20] enforces bidirectional reconstruction by ensuring that a message M extracted from stego-audio $G(C, M)$ matches the original:

$$L_{\text{cycle}} = \|M - D_{\text{ext}}(G(C, M))\|_1 \tag{4}$$

where D_{ext} is the extraction network and the L_1 norm reduces sensitivity to outlier bit errors. However, Eq. (4) is evaluated only on the clean, unperturbed stego-audio $G(C, M)$; because no channel or attack model is inserted between G and D_{ext} during optimisation, the loss provides no gradient signal about robustness to signal processing attacks, so low cycle loss at training time is not predictive of extraction accuracy after compression, resampling, or noise [17].

4.2.5 Embedding Loss

Embedding loss directly minimises the squared error between the original and extracted message:

$$L_{\text{embed}} = \|M - D_{\text{ext}}(G(C, M))\|_2^2 \tag{5}$$

Typically combined with adversarial loss in a weighted sum [6, 29], its key limitation is exclusive focus on bit accuracy without constraining audio perceptual quality, and the fixed weighting fails to respond to varying payload densities or local SNR [5, 8]: the L_2 term in Eq. (5) is applied uniformly across the whole message and, by extension, across the whole audio signal, so it drives the generator to spend the imperceptibility budget equally on spectrally sensitive and spectrally insensitive regions, which is the mechanism behind the quality degradation both Yang et al. [5] and Li et al. [8] report.

4.2.6 Multi-Adversarial Loss

Multi-adversarial loss [18] employs K steganalyzers $D_1, \ldots, D_K$ with adaptive weights:

$$L_{\text{multi}} = \sum_{k=1}^{K} w_k \cdot L_{\text{adv}}(G, D_k), \quad \text{where } w_k \text{ is dynamically assigned} \tag{6}$$

where w_k is assigned based on each steganalyzer's detection accuracy at each training step. The generator is updated using gradients from the strongest discriminator, preventing gradient vanishing [18], but because every term in Eq. (6) is itself an adversarial loss of the form in Eq. (1), the summation only ever re-weights detectability signals; it contains no term sensitive to bit-rate or payload size, which is precisely why this formulation does not address embedding capacity and has primarily been validated on image rather than audio domains.

Cross-cutting synthesis. Reading Eqs. (1)–(6) together exposes a consistent structural pattern rather than six unrelated designs: every category is a linear combination of terms drawn from only two underlying signals – a discriminability signal (adversarial, WGAN-GP, multi-adversarial) and a fidelity/accuracy signal (perceptual, cycle consistency, embedding) – combined with scalar weights that are fixed at design time. This explains a trend visible across Table 2: studies that combine one term from each family (e.g., adversarial + perceptual [7, 29], adversarial + embedding [5, 6]) report better joint imperceptibility–capacity behaviour than single-term formulations, yet none report a term sensitive to *local, time-varying* audio structure. This is also where the two most-cited audio-domain studies in the set appear to disagree: Chen et al. [7, 29] treat a fixed psychoacoustic threshold as sufficient for imperceptibility, whereas Feng et al. [9] and Yang et al. [5] present evidence that any fixed threshold or fixed weighting degrades unevenly across spectrally sensitive and insensitive regions – i.e., the reviewed literature has not converged on whether a static perceptual constraint is adequate, which is itself a finding of this review rather than an assumption carried into it.

5 Analysis, Discussion, and Future Directions

This section provides a structured discussion of the findings from the systematic review in Section 4. The discussion is designed to address each of the research questions research question in sequence, thereby answering RQ1, RQ2, and RQ3 with supporting findings from the literature and the mathematical formulations presented aboven Distribution and Component Mapping (RQ1)

Table 4 summarises the frequency of each loss function type across the 33 reviewed studies. Note that percentages exceed 100% because some studies utilise more than one loss function type.

Table 4: Distribution of Loss Function Types Across Reviewed Studies

Loss Function Type	Studies	%	Primary Use
Adversarial Loss	22	66.7	Foundation of GAN training
Perceptual Loss	6	18.2	Feature-space imperceptibility
Embedding Loss	6	18.2	Message recovery accuracy
Cycle Consistency Loss	4	12.1	Bidirectional reconstruction
Wasserstein-GP Loss	2	6.1	Training stability
Multi-Adversarial Loss	2	6.1	Steganalysis resistance

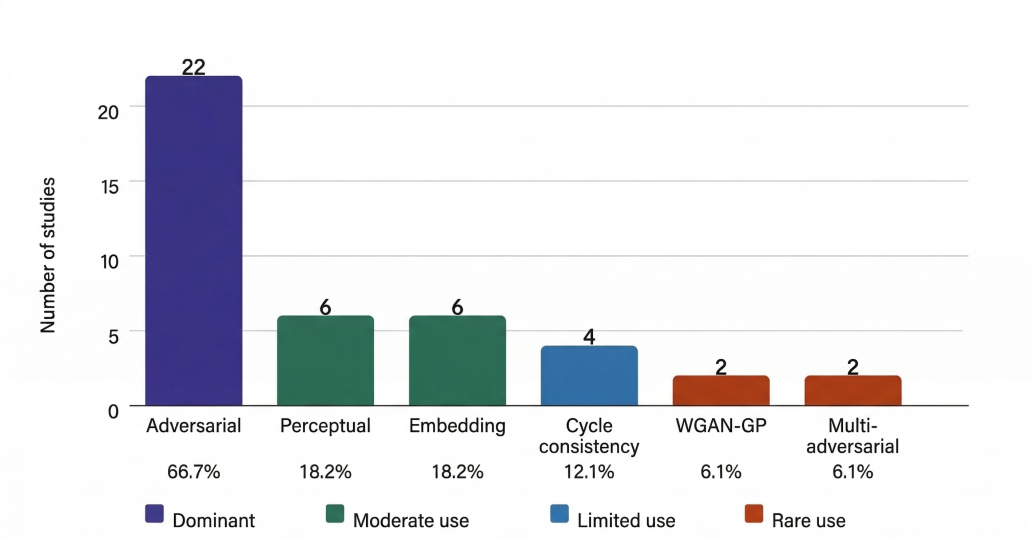


Figure 3: Loss Function Type Distribution Across 33 Reviewed Studies

As expected, adversarial loss dominates (66.7%), reinforcing that most existing GAN-based audio steganography utilises the standard minimax loss without any modification. Specialized types of loss functions are still not well explored: both perceptual loss and embedding loss are utilised in only six studies (18.2%), and both WGAN-GP and multi-adversarial loss are utilised in only two studies (6.1%). The fact that both WGAN-GP and

Table 5: Performance Characteristics of Loss Function Types Across Reviewed Studies

Loss Function	Imperceptibility	Robustness	Embedding Capacity	Refs
Adversarial	✓ Evaluated; SNR degrades without perceptual constraint	– Not evaluated against attacks	✓ Evaluated; moderate payload, no explicit capacity term	[1, 7, 19]
Wasserstein-GP	✓ Evaluated; training stability improved, no audio quality gain	– Not evaluated	– Not evaluated	[11, 18]
Perceptual	✓ Evaluated; PSNR/SSIM improved, but image-pretrained features limit audio alignment	– Not evaluated under compression or noise	– Not evaluated	[12, 24]
Cycle Consistency	✓ Evaluated; lower BER via bidirectional reconstruction; no perceptual metric	– Lossy channel distortion not modelled	– No capacity gain reported	[13, 27]
Embedding Loss	✓ Evaluated; quality degrades in spectrally sensitive regions	– Not evaluated	✓ Evaluated; directly minimises BER; highest capacity focus	[5, 6, 8]
Multi-Adversarial	– No audio quality metric reported	✓ Evaluated; adaptive steganalyzer weighting improves detection resistance	– Not evaluated; no capacity guidance from discriminators	[14]

multi-adversarial loss are not well utilised in audio steganography, despite their advantages, reveals that the field of audio steganography still does not effectively utilise the progress of the broader field of GAN.

5.1 Limitations and Performance Trade-Offs (RQ2)

Table 5 cross-matches each loss function type with the three main performance dimensions: imperceptibility, robustness, and embedding capacity.

The results in Table 5 indicate that none of the loss functions were tested over all three aspects. The perceptual loss is the only one that reports improvements in imperceptibility metrics PSNR and SSIM [12, 24], although Feng et al. [9] verified that these improvements do not extend to resistance against steganalysis since the feature extractors are trained on image data rather than audio data. The embedding loss achieves the best results in capacity evaluation by directly optimizing the bit error rate [5, 6, 8], although Yang et al. [5] and Li et al. [8] verified that audio quality degrades when loss weights are uniformly applied over spectrally sensitive and insensitive regions. The only formulation with robustness evaluation is the multi-adversarial loss, although Huang et al. [14] pointed out that no capacity or imperceptibility feedback is provided by this formulation. The Wasserstein-GP loss resolved gradient instability, though Yue et al. [18] proved there was no quality improvement in the resulting audio.

To move beyond this qualitative cross-tabulation, Table 6 translates the evidence in Table 5 and Section 4.2 into a semi-quantitative synthesis, scoring each loss category on an ordinal 1–3 scale (1 = weak/unaddressed, 2 = partial, 3 = strong) for each performance dimension, together with a qualitative computational-cost rating. The scores are the authors' structured synthesis of the reviewed evidence, not measurements from a single controlled experiment (no reviewed study benchmarks all six categories under identical conditions), and are reported to make the trade-offs directly comparable rather than to imply cross-study statistical equivalence.

Table 6: Semi-Quantitative Synthesis of Loss Function Trade-Offs (1 = weak, 2 = partial, 3 = strong)

Loss Function	Imperc.	Robust.	Capacity	Train. Stability	Comp. Cost
Adversarial	2	1	2	2	Low
Wasserstein-GP	2	1	1	3	High
Perceptual	3	1	1	2	Medium
Cycle Consistency	2	1	2	2	Medium
Embedding Loss	2	1	3	2	Low
Multi-Adversarial	1	3	1	2	High

Table 6 makes the central pattern of the review explicit: no category scores 3 on more than one dimension, and robustness scores 1 across every category except multi-adversarial loss, which is itself the only category not validated on audio data. This is not merely a repetition of the "fixed weights" observation, but its quantitative consequence – because $\alpha, \beta, \gamma, \delta$-style weightings in the hybrid formulations of [6, 7, 29] are set once at design time and held constant across an entire utterance, the marginal gain any one loss term contributes in a spectrally sensitive frame is treated identically to its marginal gain in a spectrally flat frame; the effect compounds as payload density increases, which is consistent with the capacity-imperceptibility degradation reported by both Yang et al. [5] and Li et al. [8]. The major limitation common to all six formulations is that loss weights are fixed and manually tuned during training [5, 8, 9, 25, 29]; none of the reviewed works considers audio parameters such as local SNR, spectral flatness, and temporal masking threshold in loss optimisation, and, as Table 6 shows, this omission is the common cause of the low robustness scores across every category rather than an independent, unrelated weakness of each one.

5.2 Future Directions for Adaptive Loss Function Design (RQ3)

To answer RQ3, three specific and actionable research directions are proposed. First, new adaptive loss weight scheduling mechanisms should be developed to schedule the coefficients α, β, γ, and δ in the hybrid loss function in Eq. (6), updated in real time according to feedback from local SNR values, gradient stability, and payload distribution requirements; concretely, this could take the form of a lightweight controller network that maps a short window of local SNR, spectral flatness, and temporal masking threshold to weight adjustments at each training step, evaluated first on a single audio-domain loss category (e.g., embedding loss) before generalising to hybrid formulations. Second, psychoacoustically informed perceptual loss functions should be developed by replacing the conventional image-based feature extractor [13, 16] with audio-specific feature extractors such as mel frequency cepstral coefficients and gammatone filterbank features, with success measured not only by PSNR/SSIM-style fidelity metrics but by steganalysis detection accuracy under the resulting loss, directly addressing the imperceptibility–steganalysis disconnect identified in [9]. Finally, new lightweight versions of the Wasserstein-GP loss

functions should be developed to minimize the computational cost of the gradient penalty and make training stability accessible in real-time and IoT-based audio steganography applications [10, 24], for instance by restricting gradient penalty sampling to a reduced subset of interpolated points per batch rather than the full interpolation path. The application of the reinforcement learning paradigm in loss weight selection, formulated as a sequential decision process similar to [15], will enable selection of the loss weight that achieves the highest reward based on detection resistance, quality, and capacity metrics; the semi-quantitative dimensions in Table 6 offer a natural starting point for defining such a reward signal.

6 Conclusion

This paper systematically reviewed GAN-based audio steganography loss functions by analysing 33 studies following a PRISMA-style methodology. Six loss function categories were identified, each with distinct trade-offs in imperceptibility, robustness, embedding capacity, and training stability. Based on RQ1, adversarial loss was identified as the dominant approach, accounting for 66.7%. RQ2 showed that the absence of audio-specific parameters is the common cause of limited robustness across all categories. RQ3 recommended developing adaptive and acoustic-aware hybrid loss functions. The review further contributed a unified framework demonstrating that existing methods rely on discriminability and fidelity without adapting to time-varying acoustic characteristics. Limitations include a lightweight quality assessment and the lack of formal inter-rater agreement.

Acknowledgments

This work was funded by the Ministry of Higher Education (MoHE) of Malaysia under the Fundamental Research Grant Scheme (FRGS/1/ 2020/ICT02/USIM/02/1). The authors would like to express their gratitude to Universiti Sains Islam Malaysia (USIM) and MoHE for the support and facilities provided.

References

[1] L. Chen, R. Wang, D. Yan, and J. Wang, "Learning to Generate Steganographic Cover for Audio Steganography Using GAN," *IEEE Access*, vol. 9, pp. 88098–88107, 2021.

[2] X. Wang and Y. Wang, "A novel audio steganography based on the segmentation of the foreground and background of audio," *Comput. Electr. Eng.*, vol. 123, p. 110026, 2025.

[3] J. Xi, J. Liu, L. Dong, D. Yan, and R. Wang, "Diffusion-based model for audio steganography," *Electronics*, vol. 14, no. 20, p. 4019, 2025.

[4] I. Goodfellow et al., "Generative Adversarial Nets," in *Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 2672–2680.

[5] J. Yang, H. Zheng, X. Kang, and Y.-Q. Shi, "Approaching Optimal Embedding in Audio Steganography With GAN," in *Proc. IEEE ICASSP*, 2020, pp. 2827–2831.

[6] D. Ye, S. Jiang, and J. Huang, "Heard More Than Heard: An Audio Steganography Method Based on GAN," *arXiv:1907.04986*, 2019.

[7] L. Chen, R. Wang, L. Dong, and D. Yan, "Imperceptible adversarial audio steganography based on psychoacoustic model," *Multimed. Tools Appl.*, vol. 82, pp. 26451–26463, 2023.

[8] J. Li, K. Wang, and X. Jia, "A Coverless Audio Steganography Based on Generative Adversarial Networks," *Electronics*, vol. 12, no. 5, p. 1253, 2023.

[9] Y. Feng, L. Xu, X. Lu, G. Zhang, and W. Rao, "A Robust Coverless Audio Steganography Based on Differential Privacy Clustering," *IEEE Trans. Multimed.*, 2025.

[10] Z. Su, G. Zhang, Z. Shi, D. Hu, and W. Zhang, "Message-Driven Generative Music Steganography Using MIDI-GAN," *IEEE Trans. Dependable Secure Comput.*, vol. 21, no. 6, pp. 5196–5207, 2024.

[11] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved Training of Wasserstein GANs," in *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5767–5777.

[12] B. Singh, P. K. Sharma, and S. A. Huddedar, "StegGAN: Hiding Image Within Image Using Conditional Generative Adversarial Networks," *Multimed. Tools Appl.*, vol. 81, pp. 40511–40533, 2022.

[13] Z. Chen, L. Zhu, P. Jiang, J. He, and Z. Zhang, "A Generic Blockchain-Based Steganography Framework with High Capacity via Reversible GAN," in *Proc. IEEE INFOCOM*, 2024, pp. 241–250.

[14] D. Huang, W. Luo, M. Liu, W. Tang, and J. Huang, "Steganography Embedding Cost Learning With Generative Multi-Adversarial Network," *IEEE Trans. Inf. Forensics Secur.*, vol. 19, pp. 15–29, 2024.

[15] A. Martín, A. Hernández, M. Alazab, J. Jung, and D. Camacho, "Evolving Generative Adversarial Networks to Improve Image Steganography," *Expert Syst. Appl.*, vol. 222, p. 119841, 2023.

[16] J. Liu et al., "Recent Advances of Image Steganography With Generative Adversarial Networks," *IEEE Access*, vol. 8, pp. 60575–60597, 2020.

[17] A. S. Saqlain, S. M. A. Shah, W. Aslam, I. A. Khan, and B. M. Khan, "Evolution and Effectiveness of Loss Functions in Generative Adversarial Networks," *China Commun.*, vol. 18, no. 10, pp. 45–76, 2021.

[18] F. Yue, H. Zhu, Z. Su, and G. Zhang, "An Adaptive Audio Steganography Using BN Optimizing SNGAN," *Chinese J. Comput.*, vol. 45, pp. 427–440, 2022.

[19] J. Wang, R. Wang, L. Dong, and D. Yan, "Robust, Imperceptible and End-to-End Audio Steganography Based on CNN," in *Proc. SPDE 2020*, Springer, pp. 427–442, 2020.

[20] V. Moorthy and R. Venkataraman, "Generative Adversarial Analysis using U-LSB Based Audio Steganography," in *Proc. IEEE INDICON*, 2021, pp. 1–6.

[21] W. Tang, S. Tan, B. Li, and J. Huang, "Automatic Steganographic Distortion Learning Using a Generative Adversarial Network," *IEEE Signal Process. Lett.*, vol. 24, no. 10, pp. 1547–1551, 2017.

[22] J. Tan, X. Liao, J. Liu, Y. Cao, and H. Jiang, "Channel Attention Image Steganography With Generative Adversarial Networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 2, pp. 888–903, 2022.

[23] F. Peng, G. Chen, and M. Long, "A Robust Coverless Steganography Based on Generative Adversarial Networks and Gradient Descent Approximation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 5817–5829, 2022.

[24] L. Zeng, N. Yang, X. Li, A. Chen, H. Jing, and J. Zhang, "Advanced Image Steganography Using a U-Net-Based Architecture with Multiscale Fusion and Perceptual Loss," *Electronics*, vol. 12, no. 18, p. 3808, 2023.

[25] Y. Duan, J. Ren, H. Yu, and X. Jiang, "GAN-in-GAN for Monaural Speech Enhancement," *IEEE Signal Process. Lett.*, vol. 30, pp. 853–857, 2023.

[26] Z. Yang, K. Chen, K. Zeng, W. Zhang, and N. Yu, "Provably Secure Robust Image Steganography," *IEEE Trans. Multimed.*, vol. 26, pp. 5040–5053, 2024.

[27] Z. Zhou et al., "Progressive Generative Steganography via High-Resolution Image Generation for Covert Communication," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 21, no. 10, 2025.

[28] M. Helmy and H. Torkey, "Secured Audio Framework Based on Chaotic-Steganography Algorithm for Internet of Things Systems," *Computers*, vol. 14, no. 6, p. 207, 2025.

[29] L. Chen, R. Wang, L. Dong, and D. Yan, "Imperceptible adversarial audio steganography based on psychoacoustic model," *Multimed. Tools Appl.*, vol. 82, pp. 26451–26463, 2023.

[30] V. Holub and J. Fridrich, "Designing Steganographic Distortion Using Directional Filters," in *Proc. IEEE WIFS*, 2012, pp. 234–239.

[31] M. M. Mahmoud and H. T. Elshoush, "Enhancing LSB Using Binary Message Size Encoding for High Capacity, Transparent and Secure Audio Steganography," *IEEE Access*, vol. 10, pp. 29954–29971, 2022.

[32] A. A. Ying, R. Wang, Y. Lin, and D. Yan, "Adaptive Audio Steganography Based on Improved Syndrome-Trellis Codes," *IEEE Access*, vol. 9, pp. 11705–11715, 2021.

[33] J. Wu, B. Chen, W. Luo, and Y. Fang, "Audio Steganography Based on Iterative Adversarial Attacks Against Convolutional Neural Networks," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 2282–2294, 2020.

[34] Y. Yan, Y. Li, Q. Xiao, and Y. Ren, "PRoADS: Provably Secure and Robust Audio Diffusion Steganography with Latent Optimization and Backward Euler Inversion," *arXiv:2603.10314*, 2026.

[35] J. Yan, Y. Cheng, Z. Yin, X. Zhang, S. Wang, T. Sun, and X. Jiang, "FGAS: Fixed Decoder Network-Based Audio Steganography with Adversarial Perturbation Generation," *arXiv:2505.22266*, 2025.

[36] P. Zhuo, D. Yan, K. Ying, and L. Dong, "Audio Steganography Cover Enhancement via Reinforcement Learning," *Signal Image Video Process.*, vol. 18, pp. 537–545, 2024.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 319–331.

A New NTRU Variant with Composite Private Structure

Kamarulhaili, H.[1,4], Salleh, N.[1], Asbullah, M. A.[2], Mandangan, A.[3] and Nasir Ali, N. M.[4]

[1] School of Mathematical Sciences, Universiti Sains Malaysia, 11800 Minden, Penang, Malaysia
hailiza@usm.my,mymy_nsbs@yahoo.com.my
[2] Centre for Foundation Studies in Sciences, Universiti Putra Malaysia,43400 Serdang, Selangor, Malaysia
ma_asyraf@upm.edu.my
[3] Mathematics Visualization Research Group, Faculty of Sciences and Technology, Universiti Malaysia Sabah, Malaysia arifman@ums.edu.my
[4] Malaysia Cryptology Technology and Management Centre(PTPKM), c/o Universiti Putra Malaysia,43400 Serdang, Selangor, Malaysia hailiza@upm.edu.my,marlianasir@upm.edu.my

Abstract. This paper presents a variant of the NTRUEncrypt cryptosystem with an additional private polynomial. The proposed construction modifies the public key structure by using three private polynomials $f(x)$, $r(x)$, and $s(x)$. The scheme operates over the truncated polynomial ring $R = \mathbb{Z}[x]/(x^N - 1)$, where multiplication is performed as cyclic convolution. We describe the key generation, encryption, and decryption algorithms, prove correctness under a bounded coefficient condition, and discuss the structural implications for lattice-based security considering the lattice attack and Meet-In-The-Middle (MITM) attack. A toy example is included to illustrate the full encryption and decryption processes.

Keywords: NTRUEncrypt · Lattices · Polynomial ring

1 Introduction

Public key cryptography plays a central role in securing modern digital communication systems. Among the various public key constructions, lattice-based cryptography has gained significant attention due to its resistance against quantum attacks [6, 8, 7]. One of the earliest and most efficient lattice-based cryptosystems is NTRUEncrypt, introduced by Hoffstein, Pipher, and Silverman [3, 4]. It is based on polynomial arithmetic in a truncated ring and is known for its computational efficiency and practical performance.

The classical NTRUEncrypt scheme operates over the ring

$$R = \mathbb{Z}[x]/(x^N - 1),$$

where encryption and decryption rely on the use of two private polynomials and a structured public key derived from their inverses. The security of NTRU is closely related to the hardness of finding short vectors in a lattice defined by the public key, making it a strong candidate for post-quantum cryptography [10]. Therefore, NTRU became one of the most influential lattice-based cryptographic schemes in post-quantum cryptography research. Over time, several variants and enhancements were proposed to improve both security and performance. Among these, D-NTRU [12] introduced alternative algebraic structures aimed at strengthening resistance against lattice reduction attacks and improving decryption behavior, while DBTRU [11] further extended the design using dual polynomial structures to increase cryptanalytic complexity. Later, Bernstein et al, [2] proposed NTRU Prime, which modifies the algebraic structure of the original NTRU system by avoiding rings with excessive symmetries that could potentially facilitate certain algebraic attacks. NTRU Prime was designed to preserve the efficiency advantages of NTRU while improving confidence in long-term security assumptions, and later became one of the prominent candidates in the NIST Post-Quantum Cryptography standardization process [1]. These developments collectively demonstrate the continuing importance of NTRU-type constructions as a major family of efficient and quantum-resistant public-key cryptography, which in turn motivates the scheme proposed in this paper.

In this scheme, the main algebraic structure used is a polynomial ring. Specifically, the scheme operates over the ring:

$$R = \mathbb{Z}[x]/(x^N - 1),$$

which works with polynomials whose coefficients are integers, but all operations (addition and multiplication) are performed modulo $x^N - 1$.

For the encryption and decryption processes, computations are done modulo two integers, typically parameters p and q:

$$R_p = (\mathbb{Z}/p\mathbb{Z})[x]/(x^N - 1),$$
$$R_q = (\mathbb{Z}/q\mathbb{Z})[x]/(x^N - 1),$$

Thus, computations also advance the method of the center-lifting modulo q and p to R. This means a reduction modulo q is performed coefficient-wise, and center-lifting modulo q maps each coefficient into

$$\left[-\frac{q}{2}, \frac{q}{2}\right).$$

Similarly, center-lifting modulo $p = 3$ maps coefficients into

$$\{-1, 0, 1\}.$$

Some early results by Salleh et al. were discussed in [9] and, in this paper, we further enhance them with more detailed results, discussions, and a toy example illustrating complete encryption and decryption processes.

2 The Proposed NTRU Variant

2.1 Parameters

The scheme uses the following parameters: N, p, q, d_f, d_r, d_s, d_ϕ.

Table 1: Parameters in the proposed NTRU variant.

Parameter	Description
N	Dimension of R
p	Small modulus
q	Large modulus
d_f	Number of coefficients of $f(x)$
d_r	Number of coefficients of $r(x)$
d_s	Number of coefficients of $s(x)$
d_ϕ	Number of coefficients of $\phi(x)$

Here, N controlling polynomial length. p used for message coefficients, and q used in ciphertext to retain correctness, with p and q are co-prime, and $p \ll q$. While, d_f, d_r, d_s, and d_ϕ wrap around modulo p or q.

2.2 Polynomial Spaces and Density Conditions

Let $\mathcal{L}(d_1, d_2)$ denote the set of ternary polynomials in R with exactly d_1 coefficients equal to $+1$, exactly d_2 coefficients equal to -1, and all remaining coefficients equal to 0. The proposed scheme uses the following spaces:

$$\mathcal{L}_f = \mathcal{L}(d_f, d_f - 1),$$
$$\mathcal{L}_r = \mathcal{L}(d_r, d_r + 1),$$
$$\mathcal{L}_s = \mathcal{L}(d_s + 1, d_s),$$
$$\mathcal{L}_\phi = \mathcal{L}(d_\phi, d_\phi).$$

The message space $\mathcal{L}_m$ consists of polynomials whose coefficients lie in

$$\left[-\frac{p-1}{2}, \frac{p-1}{2}\right]$$

when p is odd. For $p = 3$, this means the coefficients of $m(x)$ lie in $\{-1, 0, 1\}$.

The use of d_1 positive coefficients and d_2 negative coefficients for $f(x)$, $r(x)$, and $s(x)$ avoids the problem

$$f(1) = r(1) = s(1) = 0,$$

which would occur if the numbers of $+1$ and -1 coefficients were exactly equal. If $f(1) = 0$, $r(1) = 0$, or $s(1) = 0$, the corresponding polynomial has a factor $x - 1$, which may prevent the required inverse from existing.

2.3 Key Generation

Choose $f(x) \in \mathcal{L}_f$ and $s(x) \in \mathcal{L}_s$ such that $f(x)$ and $s(x)$ are invertible modulo q. That is, there exists

$$F_q(x) = f^{-1}(x) \pmod{(x^N - 1, q)} \qquad \text{and} \qquad S_q(x) = s^{-1}(x) \pmod{(x^N - 1, q)}$$

satisfying

$$f(x) * F_q(x) \equiv 1 \pmod{(x^N - 1, q)} \qquad \text{and} \qquad s(x) * S_q(x) \equiv 1 \pmod{(x^N - 1, q)}$$

respectively. Choose $r(x) \in \mathcal{L}_r$ such that $r(x)$ is invertible modulo p. That is, there exists

$$R_p(x) = r^{-1}(x) \pmod{(x^N - 1, p)}$$

satisfying

$$r(x) * R_p(x) \equiv 1 \pmod{(x^N - 1, p)}.$$

The public key is computed as

$$h(x) \equiv F_q(x) * r(x) * S_q(x) \pmod{(x^N - 1, q)},$$

which equivalently,

$$h(x) \equiv f^{-1}(x) * r(x) * s^{-1}(x) \pmod{(x^N - 1, q)}.$$

Thus, the public key is h, and the private key is $\big(f, s, R_p\big)$.

2.4 Encryption

To encrypt a message $m(x) \in \mathcal{L}_m$, choose a random polynomial $\phi(x) \in \mathcal{L}_\phi$. The ciphertext is

$$e(x) \equiv \big(p\phi(x) + m(x)\big) * h(x) \pmod{(x^N - 1, q)}.$$

2.5 Decryption

Given ciphertext $e(x)$, compute

$$a(x) \equiv f(x) * s(x) * e(x) \pmod{(x^N - 1, q)}.$$

Then center-lift $a(x)$ modulo q to obtain $a'(x)$.

Next compute

$$b(x) \equiv R_p(x) * a'(x) \pmod{(x^N - 1, p)}.$$

After center-lifting $b(x)$ modulo p, the recovered message is

$$m(x) = b'(x).$$

2.6 Algorithms

The following algorithms illustrate the scheme's key generation, encryption, and decryption.

Algorithm 1 : Key Generation

Input: N, p, q, d_f, d_r, d_s
Output: Private key (f, F_q), Private key (r, R_p), Private key (s, S_q), Public key h

Define $R = \mathbb{Z}[x]/(x^N - 1)$
Choose $f \leftarrow \mathcal{L}(d_f, d_f - 1)$ until f is invertible in modulo q
Choose $r \leftarrow \mathcal{L}(d_r, d_r + 1)$ until r is invertible in modulo p
Choose $s \leftarrow \mathcal{L}(d_s + 1, d_s)$ until s is invertible in modulo q
Compute $F_q(x) = f_q^{-1}(x) \quad (mod\ x^N - 1, mod\ q)$
Compute $R_p(x) = r_p^{-1}(x) \quad (mod\ x^N - 1, mod\ p)$
Compute $S_q(x) = s_q^{-1}(x) \quad (mod\ x^N - 1, mod\ q)$
Compute $h(x) = F_q(x) * r(x) * S_q(x) \quad (mod\ x^N - 1, mod\ q)$
return Private key (f, F_q), Private key (r, R_p), Private key (s, S_q), Public key h

Algorithm 2 : Encryption

Input: N, p, q, d_ϕ, Public key h, Message m
Output: Ciphertext e

Encode a message m as a message polynomial $m(x)$ with coefficients in $[-(p-1)/2, (p+1)/2]$
Choose randomly $\phi \leftarrow \mathcal{L}(d_\phi, d_\phi)$
Compute $e(x) = \big(p\phi(x) + m(x)\big) * h(x) \quad (mod\ x^N - 1, mod\ q)$
return Ciphertext e

Algorithm 3 : Decryption

Input: N, p, q, Ciphertext e, Private key f, Inverse R_p, Private key s
Output: Message m

Compute $a(x) = f(x) * s(x) * e(x) \quad (mod\ x^N - 1, mod\ q)$
for $i = 0$ to $N - 1$ **do**
    Apply the center-lift method to the coefficients of $a(x)$
    if the coefficient $a_i < -\frac{q}{2}$ **then**
        $a_i^{'} = a_i + q$
    else if the coefficient $a_i > \frac{q}{2}$ **then**
        $a_i^{'} = a_i - q$
    else
        $a_i^{'} = a_i$
    end if
end for
Obtain $a'(x) = \sum_{i=0}^{N-1} a_i^{'} x^i$
Compute $b(x) = R_p(x) * a'(x) \quad (mod\ x^N - 1, mod\ p)$
for $i = 0$ to $N - 1$ **do**
    Apply the center-lift method to the coefficients of $b(x)$
    if the coefficient $b_i < -\frac{p}{2}$ **then**
        $b_i^{'} = b_i + p$
    else if the coefficient $b_i > \frac{p}{2}$ **then**
        $b_i^{'} = b_i - p$
    else
        $b_i^{'} = b_i$
    end if
end for
Obtain $b'(x) = \sum_{i=0}^{N-1} b_i^{'} x^i = m(x)$
Decode a message polynomial $m(x)$ into a message m
return Message m

3 Correctness

Theorem 1. *Let*

$$e(x) \equiv \big(p\phi(x) + m(x)\big) * h(x) \pmod{(x^N - 1, q)}.$$

If

$$\left\| \big(p\phi(x) + m(x)\big) * r(x) \right\|_\infty < \frac{q}{2},$$

then decryption correctly recovers $m(x)$.

Proof. In the decryption,

$$e(x) \equiv \big(p\phi(x) + m(x)\big) * h(x) \pmod{(x^N - 1, q)}$$

is multiplying with $f(x) * s(x)$ to get

$$a(x) \equiv f(x) * s(x) * \big(p\phi(x) + m(x)\big) * h(x) \pmod{(x^N - 1, q)}.$$

Since $h(x) \equiv f^{-1}(x) * r(x) * s^{-1}(x) \pmod{(x^N - 1, q)}$, we have

$$a(x) \equiv f(x) * s(x) * \big(p\phi(x) + m(x)\big) * f^{-1}(x) * r(x) * s^{-1}(x) \pmod{(x^N - 1, q)}.$$

Because multiplication in R_q is commutative,

$$a(x) \equiv \big(p\phi(x) + m(x)\big) * r(x) \pmod{(x^N - 1, q)}.$$

This implies that

$$\left\| \big(p\phi(x) + m(x)\big) * r(x) \right\|_\infty < \frac{q}{2}.$$

Then the center-lifting modulo q of $a(x)$ gives

$$a'(x) = \big(p\phi(x) + m(x)\big) * r(x).$$

Next, multiplying by $R_p(x) = r^{-1}(x)$,

$$b(x) \equiv R_p(x) * a'(x) = r^{-1}(x) * \big(p\phi(x) + m(x)\big) * r(x) \equiv p\phi(x) + m(x) \pmod{(x^N - 1, p)}.$$

Since $p\phi(x) \equiv 0 \pmod{p}$, then

$$b(x) \equiv m(x) \pmod{(x^N - 1, p)}.$$

After center-lift modulo p of $b(x)$, the original message $m(x)$ is recovered. □

Ultimately, correctness requires

$$\left\| \big(p\phi(x) + m(x)\big) * r(x) \right\|_\infty < \frac{q}{2},$$

where the expression $\big(p\phi(x) + m(x)\big) * r(x)$ has magnitude

$$\big(p\phi(x) + m(x)\big) * r(x) = p\phi(x) * r(x) + m(x) * r(x) = p \cdot (2d_r + 1) + \frac{p}{2} \cdot (2d_r + 1),$$

which, equivalently,

$$\left(3d_r + \frac{3}{2}\right)p < \frac{q}{2} \qquad \text{or} \qquad (6d_r + 3)p < q$$

to guarantee that the decryption is performed correctly.

4 Comparison with Classical NTRU and NTRU Prime

The following is a comparison of the proposed NTRU variant with the classical NTRU and NTRU Prime.

Table 2: Feature comparison of the proposed variant with the classical NTRU and NTRU Prime.

Feature	Proposed NTRU Variant	Classical NTRU	NTRU Prime
Private polynomials	f, r, s	f, g	f, g
Public key	$h = f^{-1}rs^{-1}$	$h = f^{-1}g$	$h = gf^{-1} \pmod{q}$
Decryption multiplier	fs	f	f
Hidden relation	$fsh \equiv r$	$fh \equiv g$	$hf \equiv g \pmod{q}$
Lattice target	$(f * s, r)$	(f, g)	(f, g)

It can be seen that the proposed scheme has an additional private key compared to classical NTRU and NTRU Prime. In addition, the following presents experimental performance comparisons of the proposed NTRU variant with the classical NTRU and NTRU Prime, including computational complexity, key generation time, encryption time, and decryption time.

Table 3: Arithmetic operation of the proposed NTRU variant, classical NTRU, and NTRU Prime.

	Proposed NTRU Variant	Classical NTRU	NTRU Prime
Key Generation	$h(x) = F_q(x) * r(x) * S_q(x)$	$h(x) = F_q(x) * g(x)$	$h(x) = F_q(x) * g(x)$
	2 Multiply	1 Multiply	1 Multiply
Encryption	$e(x) = p\phi(x) * h(x) + m(x) * h(x)$	$e(x) = p\phi(x) * h(x) + m(x)$	$e(x) = p\phi(x) * h(x) + m(x)$
	2 Multiply & 1 Add	1 Multiply & 1 Add	1 Multiply & 1 Add
Decryption	$a(x) = p\phi(X) * r(x) + m(x) * r(x)$	$a(x) = p\phi(x) * g(x) + f(x) * m(x)$	$a(x) = p\phi(x) * g(x) + f(x) * m(x)$
	2 Multiply & 1 Add	2 Multiply & 1 Add	2 Multiply & 1 Add

* Multiply = Convolution multiplication ** Add = Polynomial addition

Table 4: Real-time performance of the proposed NTRU variant, classical NTRU, and NTRU Prime.

Real-Time Performance	Proposed Variant	Classical NTRU	NTRU Prime
Key Generation Time	0.0854474	0.0751725	0.163688
Encryption Time	0.0000278	0.0000254	0.0000276
Decryption Time	0.0000038	0.0000038	0.0000047

5 Toy Example

We now illustrate the scheme using $N = 101, \quad p = 3, \quad q = 1009$. Choose parameters $d_f = 5, \quad d_r = 4, \quad d_s = 4, \quad d_\phi = 3$. Thus, choose

$$f(x) = x^6 - x^{20} + x^{21} - x^{38} - x^{56} + x^{70} + x^{76} - x^{78} + x^{96} \in \mathcal{L}(5, 4)$$
$$r(x) = -x^{17} - x^{19} - x^{31} + x^{32} - x^{42} + x^{44} - x^{49} + x^{72} + x^{98} \in \mathcal{L}(4, 5)$$

and

$$s(x) = x^{10} + x^{15} - x^{51} - x^{52} + x^{56} + x^{64} + x^{73} - x^{88} - x^{95} \in \mathcal{L}(5, 4)$$

Using the chosen private polynomials $f(x)$, $r(x)$, and $s(x)$, we compute

$$h(x) \equiv f^{-1}(x) * r(x) * s^{-1}(x) \pmod{(x^{101} - 1, 1009)}.$$

The resulting public key polynomial is:

$$\begin{aligned} h(x) = {} & 144 + 461x + 985x^2 + 892x^3 + 663x^4 + 220x^5 + 612x^6 + 945x^7 \\ & + 844x^8 + 55x^9 + 195x^{10} + 184x^{11} + 383x^{12} + 356x^{13} + 962x^{14} \\ & + 489x^{15} + 488x^{16} + 121x^{17} + 817x^{18} + 315x^{19} + 490x^{20} + 779x^{21} \\ & + 199x^{22} + 314x^{23} + 662x^{24} + 634x^{25} + 543x^{26} + 870x^{27} + 430x^{28} \\ & + 25x^{29} + 382x^{30} + 388x^{31} + 932x^{32} + 602x^{33} + 508x^{34} + 173x^{35} \\ & + 432x^{36} + 538x^{37} + 244x^{38} + 710x^{39} + 920x^{40} + 188x^{41} + 933x^{42} \\ & + 892x^{43} + 547x^{44} + 859x^{45} + 956x^{46} + 721x^{47} + 262x^{48} + 386x^{49} \\ & + 557x^{50} + 294x^{51} + 616x^{52} + 200x^{53} + 794x^{54} + 900x^{55} + 880x^{56} \\ & + 417x^{57} + 577x^{58} + 807x^{59} + 251x^{60} + 244x^{61} + 999x^{62} + 526x^{63} \\ & + 322x^{64} + 396x^{65} + 518x^{66} + 175x^{67} + 597x^{68} + 565x^{69} + 561x^{70} \\ & + 938x^{71} + 958x^{72} + 350x^{73} + 1000x^{74} + 947x^{75} + 475x^{76} + 850x^{77} \\ & + 583x^{78} + 925x^{79} + 375x^{80} + 570x^{81} + 660x^{82} + 51x^{83} + 454x^{84} \\ & + 317x^{85} + 477x^{86} + 218x^{87} + 807x^{88} + 760x^{89} + 9x^{90} + 206x^{91} \\ & + 829x^{92} + 227x^{93} + 49x^{94} + 304x^{95} + 988x^{96} + 619x^{97} + 586x^{98} \\ & + 486x^{99} + 621x^{100}. \end{aligned}$$

Remark The polynomial $h(x)$ is computed using polynomial inversion and convolution modulo $(x^{101} - 1, 1009)$ using computer algebra software.

Let the message be

$$\texttt{Secret}.$$

Using 8-bit ASCII,

$$\texttt{Secret} = 01010011\ 01100101\ 01100011\ 01110010\ 01100101\ 01110100.$$

The corresponding polynomial is

$$\begin{aligned} m(x) = {} & x + x^3 + x^6 + x^7 + x^9 + x^{10} + x^{13} + x^{15} + x^{17} + x^{18} + x^{22} + x^{23} + x^{25} \\ & + x^{26} + x^{27} + x^{30} + x^{33} + x^{34} + x^{37} + x^{39} + x^{41} + x^{42} + x^{43} + x^{45}. \end{aligned}$$

Choose

$$\phi(x) = -x + x^3 - x^5 - x^{25} + x^{55} + x^{73} \in \mathcal{L}(3, 3).$$

We compute the ciphertext

$$e(x) \equiv \big(3\phi(x) + m(x)\big) * h(x) \pmod{(x^{101} - 1, 1009)}.$$

The resulting ciphertext is:

$$\begin{aligned} e(x) = {} & 407 + 818x + 630x^2 + 67x^3 + 290x^4 + 134x^5 + 597x^6 + 504x^7 \\ & + 503x^8 + 206x^9 + 825x^{10} + 14x^{11} + 793x^{12} + 255x^{13} + 305x^{14} \\ & + 77x^{15} + 662x^{16} + 874x^{17} + 590x^{18} + 915x^{19} + 537x^{20} + 55x^{21} \\ & + 616x^{22} + 248x^{23} + 572x^{24} + 894x^{25} + 687x^{26} + 913x^{27} + 777x^{28} \\ & + 507x^{29} + 327x^{30} + 902x^{31} + 911x^{32} + 305x^{33} + 770x^{34} + 810x^{35} \\ & + 829x^{36} + 700x^{37} + 511x^{38} + 679x^{39} + 270x^{40} + 138x^{41} + 904x^{42} \\ & + 320x^{43} + 698x^{44} + 481x^{45} + 369x^{46} + 762x^{47} + 114x^{48} + 228x^{49} \\ & + 634x^{50} + 345x^{51} + 554x^{52} + 644x^{53} + 690x^{54} + 836x^{55} + 660x^{56} \\ & + 363x^{57} + 550x^{58} + 669x^{59} + 762x^{60} + 856x^{61} + 533x^{62} + 340x^{63} \\ & + 991x^{64} + 129x^{65} + 611x^{66} + 232x^{67} + 0x^{68} + 211x^{69} + 536x^{70} \\ & + 888x^{71} + 977x^{72} + 50x^{73} + 849x^{74} + 198x^{75} + 619x^{76} + 977x^{77} \\ & + 599x^{78} + 266x^{79} + 441x^{80} + 361x^{81} + 222x^{82} + 532x^{83} + 69x^{84} \\ & + 706x^{85} + 775x^{86} + 640x^{87} + 826x^{88} + 996x^{89} + 864x^{90} + 453x^{91} \\ & + 747x^{92} + 661x^{93} + 233x^{94} + 835x^{95} + 746x^{96} + 653x^{97} + 36x^{98} \\ & + 784x^{99} + 13x^{100}. \end{aligned}$$

Remark The polynomial $e(x)$ is computed using polynomial inversion and convolution modulo $(x^{101} - 1, 1009)$ using computer algebra software.

Next, we compute

$$a(x) \equiv f(x) * s(x) * e(x) \pmod{(x^{101} - 1, 1009)}.$$

Noting that

$$\|a(x)\|_\infty = \left(3d_r + \frac{3}{2}\right)p = \left(3(4) + \frac{3}{2}\right)3 = \frac{81}{2} = 40.5$$

Since

$$40.5 < \frac{q}{2} = \frac{1009}{2} = 504.5,$$

the correctness condition is satisfied.

Then center-lifting of $a(x)$ modulo 1009 yields

$$\begin{aligned}a'(x) = 4 &+ x - 3x^2 - 5x^3 + 5x^4 + x^5 + x^6 + x^7 + x^8 + 2x^{10} + 2x^{12} + x^{13}\\ &- x^{14} + x^{15} + 4x^{16} + 2x^{18} + x^{19} - x^{20} - 2x^{21} - 3x^{22} + 3x^{24} - x^{25}\\ &+ x^{26} - x^{28} - x^{29} + x^{31} - 2x^{33} - 5x^{34} + 3x^{35} + 3x^{36} - 5x^{37} + x^{38}\\ &+ x^{39} - x^{40} - x^{41} + 3x^{42} + x^{43} + 3x^{44} - 6x^{45} - 2x^{46} + 7x^{47} - 2x^{48}\\ &- 5x^{49} + 3x^{50} - x^{51} - 3x^{52} - x^{53} + 3x^{54} - x^{55} - x^{56} - 3x^{57} - 3x^{58}\\ &- x^{59} - 3x^{60} - x^{61} - x^{62} - 4x^{64} - x^{65} + x^{66} + 2x^{67} - 2x^{68} - 2x^{69}\\ &+ 3x^{70} + x^{71} - 6x^{72} - 2x^{73} + 3x^{75} - 3x^{76} - x^{77} + 2x^{78} - x^{79} + x^{81}\\ &+ x^{82} - x^{83} - x^{84} + x^{85} - 3x^{86} + 4x^{87} - x^{88} + 2x^{89} - 3x^{90} - x^{91}\\ &- 4x^{92} + x^{95} - 5x^{97} + x^{98} + 2x^{99}.\end{aligned}$$

Remark The polynomial $a'(x)$ is computed using polynomial inversion and convolution modulo $(x^{101} - 1, 1009)$ using computer algebra software.

Multiplying $a'(x)$ with $R_p(x)$ resulting polynomial $b(x)$

$$\begin{aligned}b(x) &\equiv R_p(x) * a'(x) \pmod{(x^{101} - 1, 3)}\\ &= x + x^3 + x^6 + x^7 + x^9 + x^{10} + x^{13} + x^{15} + x^{17} + x^{18} + x^{22}\\ &\quad + x^{23} + x^{25} + x^{26} + x^{27} + x^{30} + x^{33} + x^{34} + x^{37} + x^{39} + x^{41}\\ &\quad + x^{42} + x^{43} + x^{45}.\end{aligned}$$

Remark The polynomial $b(x)$ is computed using polynomial inversion and convolution modulo $(x^{101} - 1, 1009)$ using computer algebra software.

After center-lifting of $b(x)$ modulo 3, the recovered message polynomial is

$$\begin{aligned}m(x) = x &+ x^3 + x^6 + x^7 + x^9 + x^{10} + x^{13} + x^{15} + x^{17} + x^{18} + x^{22}\\ &+ x^{23} + x^{25} + x^{26} + x^{27} + x^{30} + x^{33} + x^{34} + x^{37} + x^{39} + x^{41}\\ &+ x^{42} + x^{43} + x^{45}.\end{aligned}$$

Therefore,

$$b(x) = m(x).$$

Decoding the binary sequence gives

01010011 01100101 01100011 01110010 01100101 01110100,

which corresponds to

Secret.

6 Security Analysis

6.1 Meet-In-The-Middle Attack

The meet-in-the-middle attack is applied to the private keys f and s by splitting these private keys into two parts as follows:

$$f(x) = f_1(x) + f_2(x) = \sum_{i=0}^{[(N+1)/2]-1} f_i x^i + \sum_{i=(N+1)/2}^{N-1} f_i x^i \in \mathcal{L}_f,$$

and

$$s(x) = s_1(x) + s_2(x) = \sum_{i=0}^{[(N+1)/2]-1} s_i x^i + \sum_{i=(N+1)/2}^{N-1} s_i x^i \in \mathcal{L}_s,$$

assuming $N > k$ where N is an odd number and k is a positive number.

The meet-in-the-middle technique is described in the following steps:

1. Enumerate the polynomial $\big(f_1(x) * s_1(x) + f_2(x) * s_2(x)\big)$.
   Compute the polynomial $\big(f_1(x) * s_1(x) + f_2(x) * s_2(x)\big) * h(x) \pmod q$ and choose k. Then take the first k coefficients of a polynomial $\big(f_1(x) * s_1(x) + f_2(x) * s_2(x)\big) * h(x) \pmod q$ and convert each of them into a binary number. After that take the leading bit of each binary number to form the bin labeled corresponding to the polynomial $\big(f_1(x) * s_1(x) + f_2(x) * s_2(x)\big) * h(x) \pmod q$.

2. Enumerate the polynomial $\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big)$.
   Compute the polynomial $-\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x) \pmod q$ and choose the same value of k as Step 1. Then take the first k coefficients of the polynomial $-\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x) \pmod q$ and convert each of them into a binary number. After that add 1 to all the first k coefficients of the polynomial $-\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x) \pmod q$ and convert each of them again into a binary number. Next, observe the change of the leading bit of each binary number for the first k coefficients of the polynomial $-\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x) \pmod q$ before and after added by 1. If there are changes, do the modification on the corresponding leading bit. If no changes, take the leading bit of each binary number to form the bin labeled corresponding to the polynomial $-\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x) \pmod q$.

3. Check for matches.
   Match the obtained bin labeled for the polynomial $\big(f_1(x) * s_1(x) + f_2(x) * s_2(x)\big) * h(x) \pmod q$ from Step 1 with the obtained bin labeled for the polynomial $-\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x) \pmod q$ from Step 2. If the matches have the same bits, then the split parts of private keys satisfy $\big(f_1(x) + f_2(x)\big) * \big(s_1(x) + s_2(x)\big) * h(x) = r(x) \pmod q$ and this implies that the private keys f and s can be found.

Theorem 2. *If the matches for the split part of private keys are untraceable, then the proposed scheme is secure against a meet-in-the-middle attack.*

Proof. Let the private keys f and s respectively defined by

$$f(x) = f_1(x) + f_2(x) \quad \text{and} \quad s(x) = s_1(x) + s_2(x).$$

Apply these in the formula $f(x) * s(x) * h(x) = r(x) \pmod q$ yields

$$\big(f_1(x) + f_2(x)\big) * \big(s_1(x) + s_2(x)\big) * h(x) = r(x) \pmod q.$$

Expanding this gives

$$\begin{aligned}\big(f_1(x) * s_1(x) + f_2(x) * s_2(x)\big) * h(x)& \\ = r(x) - \big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x) &\pmod q.\end{aligned}$$

Using the meet-in-the-middle technique, the matches of $\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x)$ against $-\big(f_1(x) * s_2(x) + f_2(x) * s_1(x)\big) * h(x)$ do not produce the same bit at all, which concludes that the meet-in-the-middle technique cannot find the private keys f and s. Therefore, the proposed scheme is secure against meet-in-the-middle attacks. □

Remarkably, the meet-in-the-middle attacks rely on splitting the search space into smaller, correlated parts; however, because the public key $h(x)$ is a rational polynomial with coefficients broadly distributed modulo q, and convolutions with $h(x)$ produce outputs that are indistinguishable from random modulo q, an adversary cannot efficiently match partial guesses of the private keys since it cannot reduce the search space below the full exponential complexity in the degree of the secret polynomials. In other words, the independence between the public key and the low-weight private polynomials prevents attackers from partitioning the problem into tractable subproblems, thereby providing resistance against meet-in-the-middle strategies.

In addition, the meet-in-the-middle attack also manages to obtain the key and message security using

$\#\mathcal{L}_f$ and $\#\mathcal{L}_\phi$ from the brute-force attacks as follows:

$$\begin{aligned} Key\ Security = \sqrt{\#\mathcal{L}_f} &= \sqrt{\frac{N!}{(d_f)(N-2d_f+1)!\big((d_f-1)!\big)^2}} \\ &= \frac{1}{(d_f-1)!}\sqrt{\frac{N!}{(d_f)(N-2d_f+1)!}} \\ Message\ Security = \sqrt{\#\mathcal{L}_\phi} &= \sqrt{\frac{N!}{(N-2d_\phi)!\big((d_\phi)!\big)^2}} \\ &= \frac{1}{d_\phi!}\sqrt{\frac{N!}{(N-2d_\phi)!}} \end{aligned}$$

Note that the square root on $\#\mathcal{L}_f$ and $\#\mathcal{L}_\phi$ in key security and message security above can reduce the search time for finding the private keys and the message.

6.2 Lattice Attack

The lattice attack exploits the special proposed lattice, $\mathcal{L}_h^{proposed}$, in which the private keys f, r and s serve as a shortest vector. The proposed lattice, $\mathcal{L}_h^{proposed}$, is defined by the following definition.

Definition 1. The proposed lattice, $\mathcal{L}_h^{proposed}$ associated to the proposed public key $h(x) = h_0 + h_1x + \cdots + h_{N-1}x^{N-1}$ is the $2N$-dimensional lattice generated by the rows of the following proposed matrix

$$\mathcal{M}_h^{proposed} = \left(\begin{array}{cccc|cccc} 1 & 0 & \cdots & 0 & h_0 & h_1 & \cdots & h_{N-1} \\ 0 & 1 & \cdots & 0 & h_{N-1} & h_0 & \cdots & h_{N-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & h_1 & h_2 & \cdots & h_0 \\ \hline 0 & 0 & \cdots & 0 & q & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & q \end{array}\right)$$

which composed by four N-by-N quadrants:

- Upper left quadrant = Identity matrix,
- Upper right quadrant = Circulant matrix h,
- Lower left quadrant = Zero matrix,
- Lower right quadrant = q times the identity matrix.

Note that the proposed matrix, $\mathcal{M}_h^{proposed}$ is convenient to abbreviate as a 2-by-2 block matrix as follows:

$$\mathcal{M}_h^{proposed} = \begin{pmatrix} I & h \\ 0 & q \end{pmatrix}.$$

This definition will be used in the following proposition.

Proposition 1. *Suppose that* $f(x) * s(x) * h(x) = r(x) \pmod q$. *For any polynomial* $u(x) \in R$, *let*

$$f(x) * s(x) * h(x) = r(x) + q * u(x), \tag{1}$$

then

$$(f * s \quad -u)\mathcal{M}_h^{proposed} = (f * s \quad r). \tag{2}$$

Therefore, the vector $(f * s \quad r)$ *is in* $\mathcal{L}_h^{proposed}$.

Proof. Let $(f * s \quad -u)$ be a row vector. The multiplication of the vector $(f * s \quad -u)$ with the proposed matrix $\mathcal{M}_h^{proposed}$ yields

$$(f * s \quad -u)\mathcal{M}_h^{proposed} = (f * s \quad -u)\begin{pmatrix} I & h \\ 0 & q \end{pmatrix} = (f * s \quad f * s * h - qu).$$

In particular, when the vector $(f*s \quad -u)$ is multiply with the first column of proposed matrix $\mathcal{M}_h^{proposed}$, which consists the identity matrix at the first row followed by the zero matrix at the second row, produces the vector $(f*s)$. Next, when the vector $(f*s \quad -u)$ is multiply with the second column of proposed matrix $\mathcal{M}_h^{proposed}$, which is the circular matrix of h at the first row followed by q times the identity matrix at the second row, yields the vector $(f*s*h-qu)$.

Note that, the kth entry of the vector $(f*s*h-qu)$ can be represented as $\big(f_0*s_0*h_k+f_1*s_1*h_{k-1}+\cdots+f_{N-1}*s_{N-1}*h_{k+1}\big)-qu_k$. From Eq. (1), the kth entry of the vector $(f*s*h-qu)$ is equal to the kth entry of the vector r. This implies that $f*s*h-qu=r$, and hence

$$(f*s \quad f*s*h-qu)=(f*s \quad r),$$

which equivalent to Eq. (2). Therefore, the vector $(f*s \quad r)$ is in the proposed lattice, $\mathcal{L}_h^{proposed}$ since the vector $(f*s \quad r)$ is a linear combination of the column of proposed matrix $\mathcal{M}_h^{proposed}$. □

By **Proposition 1** above, the vector $(f*s \quad r)$ is explicitly in $\mathcal{L}_h^{proposed}$. In addition, the vector $(f*s \quad r)$ is possibly a short nonzero vector in $\mathcal{L}_h^{proposed}$ since it has small coefficients. This fact is verified by the proposition below.

Proposition 2. *Let the proposed parameters (N,p,q,d_f,d_r,d_s) with assumption*

$$p=3 \qquad \text{and} \qquad d_f=d_r=d_s\approx N/3 \qquad \text{and} \qquad q\approx 2pN.$$

Let the vector $(f*s \quad r)$ is a basis of the proposed lattice, $\mathcal{L}_h^{proposed}$, then

(a) $\det(\mathcal{L}_h^{proposed})=q^N$.

(b) $\|(f*s \quad r)\|\approx 0.471\sqrt{N(2N+3)}$.

(c) The Gaussian heuristic predicts that the expected shortest nonzero vector length of $\mathcal{L}_h^{proposed}$ is

$$\sigma(\mathcal{L}_h^{proposed})=\sqrt{\frac{6}{\pi e}}N. \tag{3}$$

If N is large, then there is a high probability that the shortest nonzero vectors in $\mathcal{L}_h^{proposed}$ are the vector $(f*s \quad r)$ and its rotations.

Further,

$$\frac{\|(f*s \quad r)\|}{\sigma(\mathcal{L}_h^{proposed})}\approx 0.562\sqrt{2+\frac{3}{N}}, \tag{4}$$

hence, the vector $(f*s \quad r)$ is a factor of $\mathcal{O}(\sqrt{2+3/N})$ shorter than predicted by the Gaussian heuristic.

Proof. .

(a) Since the proposed matrix $\mathcal{M}_h^{proposed}$ is an upper triangular block matrix, its determinant is the product of the diagonal entries. Its diagonal entries consist of N blocks of the identity matrix and N blocks of q times the identity matrix. Therefore, $\det(\mathcal{L}_h^{proposed})=1^N\times q^N=q^N$.

(b) It is known that $f(x)\in\mathcal{L}_f, r(x)\in\mathcal{L}_r$ and $s(x)\in\mathcal{L}_s$. Then

$$\begin{aligned}
\|(f*s \quad r)\| &= \sqrt{\|f*s\|^2+\|r\|^2}\\
&= \sqrt{(2d_f-1)(2d_s+1)+(2d_r+1)}\\
&= \sqrt{4d_fd_s+2d_f-2d_s+2d_r}\\
&= \sqrt{4\left(\frac{N}{3}\right)\left(\frac{N}{3}\right)+2\left(\frac{N}{3}\right)-2\left(\frac{N}{3}\right)+2\left(\frac{N}{3}\right)}\\
&= \sqrt{\frac{2}{9}(2N^2+3N)}\\
&= 0.471\sqrt{N(2N+3)}
\end{aligned}$$

since $d_f=d_r=d_s\approx N/3$.

(c) It is known that the dimension of $\mathcal{L}_h^{proposed}$ is $2N$. By using this fact together with Cond. (a) above in the definition of Gaussian expected shortest length [5] gives

$$\sigma(\mathcal{L}_h^{proposed}) = \sqrt{\frac{2N}{2\pi e}}\left(q^N\right)^{1/2N} = \sqrt{\frac{Nq}{\pi e}}.$$

Thus, the above equation yields Eq. (3) when applying $q \approx 2pN \approx 6N$ since $p = 3$. Note that the expected shortest nonzero vector length of $\mathcal{L}_h^{proposed}$, $\sigma(\mathcal{L}_h^{proposed})$ will produces a long distance when N is large. This implies that the vector $(f * s \quad r)$ and its rotations will have a high possibility of being the shortest nonzero vectors in $\mathcal{L}_h^{proposed}$ when N is large since the vector $(f * s \quad r)$ is a short nonzero vector in $\mathcal{L}_h^{proposed}$.

From Definition of Gaussian expected shortest length [5], the Gaussian heuristic predicts that $\|v_{\text{shortest}}\| \approx \sigma(\mathcal{L})$ and the ratio of $\|v_{\text{shortest}}\|$ to $\sigma(\mathcal{L})$ is equal to 1. Using the fact that the vector $(f*s \quad r)$ is the shortest nonzero vectors in $\mathcal{L}_h^{proposed}$ when N is large, Gaussian heuristic predicts that $\|(f*s \quad r)\| \approx \sigma(\mathcal{L}_h^{proposed})$ and the ratio of $\|(f*s \quad r)\|$ to $\sigma(\mathcal{L}_h^{proposed})$, applying Cond. (b) and Eq. (3) into it yields

$$\frac{\|(f*s \quad r)\|}{\sigma(\mathcal{L}_h^{proposed})} \approx \frac{0.471\sqrt{N(2N+3)}}{\sqrt{\frac{6}{\pi e}}N} \approx \frac{0.471}{0.838}\sqrt{\frac{N(2N+3)}{N^2}} \approx 0.562\sqrt{2+\frac{3}{N}},$$

which corresponds to Eq. (4). As N grows larger, an upper bound on the ratio of $\|(f*s \quad r)\|$ to $\sigma(\mathcal{L}_h^{proposed})$ is $\mathcal{O}(\sqrt{2+3/N})$. Therefore, the vector $(f*s \quad r)$ is a factor of $\mathcal{O}(\sqrt{2+3/N})$ shorter than predicted by the Gaussian heuristic. □

Theorem 3. *If the vector $(f*s \quad r)$ fail to be the shortest nonzero vector in $\mathcal{L}_h^{proposed}$, then the proposed scheme is secure against lattice attacks.*

Proof. **Proposition 2** tells us that the vector $(f*s \quad r)$ and its rotations will have a high possibility of being the shortest nonzero vector in $\mathcal{L}_h^{proposed}$ when N is large. Since the vector $(f*s \quad r)$ is known to be able to generate the public key h, then there are opportunities for the LLL algorithm to attack the public key h. Thus, assume that the LLL algorithm can attack the public key h in $\mathcal{L}_h^{proposed}$ if the vector $(f*s \quad r)$ is the shortest nonzero vector in $\mathcal{L}_h^{proposed}$. This implies that the LLL algorithm will produce the LLL-reduced basis with its first row corresponding to the vector $(f*s \quad r)$.

However, the LLL algorithm has difficulty producing an LLL-reduced basis when N and q grow larger. Consequently, the vector $(f*s \quad r)$ has hardly been found by the LLL algorithm and hence, the LLL algorithm fails to attack the public key h. This contradicts the assumption that the LLL algorithm is able to attack the public key h in $\mathcal{L}_h^{proposed}$ when the vector $(f*s \quad r)$ becomes the shortest nonzero vector in $\mathcal{L}_h^{proposed}$. Therefore, the proposed scheme is secure against lattice attacks. □

7 Conclusion

This paper presented a modified NTRU-type encryption scheme with an additional private polynomial, bringing the total to three. Unlike the classical NTRU framework, the proposed scheme increases the algebraic complexity of the hidden private relation while maintaining efficient polynomial convolution operations over the truncated polynomial ring. The public key is constructed as

$$h(x) \equiv f^{-1}(x) * r(x) * s^{-1}(x) \pmod{(x^N - 1, q)}.$$

Correctness follows from the relation

$$f(x) * s(x) * h(x) \equiv r(x) \pmod{(x^N - 1, p)}$$

and from the bounded coefficient condition

$$\left\|\left(p\phi(x) + m(x)\right) * r(x)\right\|_\infty < \frac{q}{2}.$$

The toy example with $N = 101$, $p = 3$, and $q = 1009$ demonstrates that the proposed construction remains practical for successful encryption and decryption processes. The main security consideration is the difficulty of recovering a short pair (u, v) satisfying

$$u(x)h(x) \equiv v(x) \pmod{q},$$

where the secret relation corresponds to

$$(u(x), v(x)) = (f(x) * s(x), r(x)).$$

In addition, the study discusses the implications of the composite private structure against meet-in-the-middle and lattice attacks, where the hidden relation among the private components adds additional difficulty to secret recovery. Overall, this work provides an alternative direction for lattice-based cryptographic design and offers a useful foundation for future research on provable security analysis, optimized parameter selection, and stronger resistance against emerging post-quantum attacks. Currently, the IND-CPA analysis follows the classical intuition of NTRUEncrypt and is not provided in this work, as the analysis is part of a work in progress.

References

[1] Daniel J. Bernstein, Billy Bob Brumley, Ming-Shing Chen, Chitchanok Chuengsatiansup, Tanja Lange, Adrian Marotzke, Bo-Yuan Peng, Nicole Tuveri, Christine van Vredendaal, and Bo-Yin Yang. NTRU prime. Selected as alternate candidates: public-key encryption and key-establishment algorithms. NIST Post-Quantum Cryptography Standardization Process Round 3, 2020.

[2] Daniel J. Bernstein, Chitchanok Chuengsatiansup, Tanja Lange, and Christine van Vredendaal. NTRU prime: reducing attack surface at low cost. Cryptology ePrint Archive, Paper 2016/461, 2016.

[3] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. *International Algorithm Number Theory Symposium*, 1423:267–288, 1998.

[4] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. *An introduction to mathematical cryptography.* Springer, New York, 2008.

[5] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. *An introduction to mathematical cryptography (2 Ed.).* Springer, New York, 2014.

[6] Jonathan Katz and Vadim Lyubashevsky. *Lattice-based cryptography.* CRC Press LLC, 2018.

[7] Daniele Micciancio and Shafi Goldwasser. *Complexity of lattice problems: A cryptographic perspective.* Springer, New York, 2002.

[8] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*, 56(6):1–40, 2009.

[9] Nurshamimi Salleh, Hailiza Kamarulhaili, and Laith Abualigah. A novel security analysis for a new NTRU variant with additional private key. *International Journal of Data and Network Science*, 7(2):765–780, 2023.

[10] Damien Stehle and Ron Steinfeld. Making NTRUEncrypt and NTRUSign as secure as standard worst-case problems over ideal lattices. Cryptology ePrint Archive, Paper 2013/004, 2013.

[11] Xiaofei Tong, Jingguo Bi, Yufei Duan, Lixiang Li, and Licheng Wang. Security analysis of DBTRU cryptosystem. *Entropy*, 24(10):1349, 2022.

[12] Baocang Wang, Hao Lei, and Yupu Hu. D-NTRU: More efficient and average case IND-CPA secure NTRU variant. *Information Sciences*, 438:15–31, 2018.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 332–346.

Mapping the Cryptographic Landscape: A Layered Abstraction Framework Illustrated via Identification Schemes without Certificates

Ji-Jian Chin[1],Yvonne Hwei-Syn Kam[1,2]

[1] School of Engineering, Computing and Mathematics, University of Plymouth, Plymouth, United Kingdom {ji-jian.chin,yvonne.kamhweisyn}@plymouth.ac.uk
[2]Faculty of Artificial Intelligence and Engineering, Multimedia University, Cyberjaya, Malaysia
hskam@mmu.edu.my

Abstract. Drawing on more than a decade of supervisory experience, this paper presents a general framework for the study of cryptography designed to help postgraduate students find a starting point in their research. At the time of writing, to the authors' knowledge, no comparable layered educational framework has been proposed in reputable cryptographic publications for segregating the areas of cryptography, causing confusion and a steep learning curve for newcomers to the field. The framework abstracts complex ideas into five layers, enabling students to become familiar with foundational concepts before engaging with research papers, providing a clear starting point for orientation before further reading across a vast range of academic resources. The five layers each provide a concise description of a different area of cryptology, allowing students to select a suitable starting point for their studies before complementing their learning with established chapters and textbooks. As the author's primary area of research is identification schemes without certificates, specifically identity-based identification (IBI), this framework is illustrated mainly using examples in that domain, but similar mappings can be easily applied to other areas of cryptography. The application of the framework is demonstrated through a detailed body of evidence drawn from the author's supervision of postgraduate students.

Keywords: Tutorial · Study Framework · Entry-level studies · Identity-based · Identification · Postgraduate · Roadmap

1 Introduction

As a final-year undergraduate or early postgraduate student begins a project requiring cryptographic research, a common practice is to seek supervision from an expert in the area. Informal conversations with colleagues suggest a typical approach: provide the student with several relevant reference papers and a collection of textbooks, then allow them to explore independently. This process is often time-consuming and discouraging, as students must navigate the vast landscape of cryptology, and much of their early effort may be lost working through extensive textbook material before reaching the concepts needed to solve their problem.

Drawing from the author's experience in academic supervision, in addition to providing reading materials, a series of lectures is also offered, often conducted individually or in small groups, to help students form a visual understanding of their area of interest and clarify the goals they are working towards.

To this end, as these lectures matured, the material has been compiled into this paper, which the author hopes will serve as a roadmap for newly initiated researchers in cryptography. The key contribution of this work is the abstraction of cryptographic research into five main layers within which academic researchers typically operate, allowing them to select the domain most relevant to their interests. A final dimension, the interface, although not included in the five layers as it is not strictly academic in nature, is also introduced to describe the interaction between academia and industry as students transition into professional practice, where practical concerns become increasingly significant.

The objectives of this paper are as follows:

1. To propose five-layer abstraction for cryptographic education.
2. To explain the fields within each layer that prospective students may wish to explore.
3. To bridge the gap between foundational student knowledge and complex theoretical research.
4. To lower the barrier to entry into cryptography by making the learning curve less discouraging.
5. To highlight the diverse body of evidence across the field of identification schemes this framework successfully captures.

As a limitation, this framework is intended as an introductory guide and is not a substitute for textbooks. It is best used as a complement to conventional study materials. The aim is to help learners navigate the various topics in cryptography more effectively when they undertake further reading.

The remainder of this paper is structured as follows. Section 2 presents a review of related literature and highlights differences between existing models and the proposed framework. Section 3 introduces the five-layer abstraction framework in detail. Section 4 demonstrates its application to identification schemes without certificates. A recent IBI survey by Vangujar et al. captures the majority of the relevant literature [80]. Finally, Section 5 concludes the paper and outlines potential directions for further improvement.

2 Literature Review

Previous work on abstraction models in cryptography, particularly for educational purposes, is limited. A structured literature search inspired by PRISMA was utilised to conduct structured searches, refinement, and filtering of relevant literature. Searches were performed across Google Scholar, IACR ePrint, IEEE Xplore, ACM Digital Library, Scopus, and Web of Science using the search string: ("layers of abstraction" OR "abstraction levels") AND (cryptology OR cryptography) AND (education OR teaching OR pedagogy).

These searches returned several hundred results; however, upon screening, the majority of highly ranked papers focused on cryptographic pedagogy for K–12 education [48], engineering education [40], and non-mathematicians [61]. Other work centred primarily on digital tools for teaching cryptography [3, 5]. To the authors' knowledge, none proposed a layered abstraction model encompassing cryptology for new researchers.

Additional searches using AI-based tools (ChatGPT, Gemini, Claude, Perplexity AI, ResearchRabbit, and SciSpace) identified three potentially relevant works using the query "What are the different layers of abstraction used in teaching cryptology?". Yung (2011) [84] is the closest in spirit, proposing a 3-layer model spanning an "abstract" design level, a "practical" level of standards-ready mechanisms, and an "actual" level of engineering lifecycle contributions; however, its intent is to push towards real-life cryptography deployment from a practitioner's perspective, rather than to serve as an initiation document for new

researchers. Maurer [55, 56] formalises abstraction in cryptographic constructions in a highly technical computer science treatment aimed at identifying the highest level at which a definition or theorem should be stated, rather than at the uninitiated. A third result, Piessens [60], is excluded as it focuses on compiler-level rather than cryptographic-education abstractions.

Overall, this brief review highlights the scarcity of abstraction frameworks specifically designed to support new researchers in cryptology. Yung's paper [84] (the closest in relevance) which has had zero recorded citations since its 2011 publication in the ACM Digital Library (at the time of writing, July 2026) suggests that little to no work has been done in this area.

3 Abstraction Framework

In this section, the five-layer abstraction framework is proposed to guide new researchers. The motivation for this framework stems from the observation that cryptographic research spans mathematics, computer science, and engineering, each with distinct terminology, perspectives, and objectives. Different areas of inquiry within cryptology give rise to significant variations in both the perspective and tone adopted across research papers.

- **Layer 1: Mathematical Foundations**. This layer formalises the mathematical structures and computational assumptions that underpin all subsequent layers of cryptographic design, and is primarily the domain of mathematicians. Core concepts such as one-way functions emerge here, supported by well-established hardness assumptions: number theory underpins public-key cryptography; abstract algebra and combinatorics underpin symmetric-key primitives (substitution, permutation, diffusion); information theory quantifies entropy and randomness for key generation; group and field theory define algebraic structures ensuring closure and invertibility; complexity theory measures asymptotic efficiency, while probability theory supports security proofs concerning negligible "lucky guess" probabilities; and linear algebra underpins lattice- and code-based post-quantum constructs.

  Cryptanalysis at this layer is conducted purely through mathematical reasoning on the structural properties of the underlying constructs, formulating attacks as the solution to hard problems (e.g. factorisation, discrete logarithms) or as weaknesses in algebraic relations, symmetries, or invariants, independent of implementation or protocol considerations.

  A mathematically inclined student who remains focused at Layer 1 can direct their efforts towards particular areas of mathematics of interest within cryptography, while avoiding the computer science and engineering aspects, such as provable security or side-channel considerations, that may be less relevant to their focus.

- **Layer 2: Primitives, Models, and Security Definitions**. At this layer, the student selects which primitive they intend to work with, as each primitive has different characteristics, definitions, and adversarial models. This is usually the stage at which students begin to understand the scope and boundaries of each primitive, and recognise that while proposing a new primitive is relatively straightforward, proving that it satisfies strong security guarantees is considerably more challenging.

  Computer scientists operate extensively at this layer. Most papers written from a computer science perspective generalise components of a cryptographic primitive so that they are not tied to specific mathematical constructs. For example, secrets are often referred to as *witnesses*, public keys as *common reference strings*, and single-use random numbers as *nonces*. Frameworks also tend to operate at this layer, as researchers aim to generalise transformation techniques that take components of

certain primitives and construct new ones, inheriting security properties through the transformation. Proofs are typically made under abstract functional assumptions, moving away from the underlying mathematical structures that instantiate them. The intent is to preserve a level of generality that captures as many instantiations as possible, although in practice this is not always achievable, and exceptions must be carefully addressed.

- **Layer 3: Instantiations and Constructions**. At this layer, using the framework provided in Layer 2 and the necessary mathematical foundations from Layer 1, a student operating at Layer 3 instantiates a cryptographic primitive and proves its security using intractable mathematical assumptions. If Layer 2 provides the blueprint and Layer 1 provides the construction material, Layer 3 is where the construction takes place. A proof of security is then required to establish (1) correctness, (2) resilience within adversarial environments, and (3) evidence that the primitive achieves its intended security goals under those conditions.

- **Layer 4: Protocols and Applications**. Layer 4 builds on the preceding layers by composing higher-layer systems, protocols, or applications into solutions that can form the basis of complete security products in design, but remains one step removed from full implementation in code or hardware. Examples of Layer 4 topics include protocols, key exchange mechanisms, federated learning, FIDO, blockchains, and multiparty computation. Depending on the layer of abstraction desired, authors may construct Layer 4 systems using general primitives from Layer 2 or provide concrete constructions by instantiating their ideas with assumptions from Layer 3.

  The first author's observation (with the exception of key exchange and key agreement papers) is that most work at this layer presents ad hoc security reasoning or high-level security discussions rather than formal proofs. More rigorous papers include formal verification methods using automated tools such as Scyther or ProVerif to provide stronger assurances. As such, Layer 4 is often where students are first introduced to formal methods for protocol verification. Key exchange and agreement protocols remain an exception: many provide reductionist proofs, but because definitions and assumptions can be modified (often to model more realistic scenarios), new cryptanalysis papers frequently emerge by adjusting these assumptions and demonstrating new attack vectors. This contributes to the volatility of the field, while simultaneously offering valuable opportunities for cryptanalysis exploration for the aspiring student.

- **Layer 5: Circuits, Code, and Implementations**. At this final layer, theoretical designs from the previous layers are realised through hardware circuits or software implementations. This includes the detailed process of translating mathematical constructs into practical code libraries, such as BouncyCastle, PyCryptodome, or Apache Milagro, which can then be incorporated as dependencies into larger application systems. Other commonly used tools include elliptic curve implementations such as Ed25519, Ed448, Ristretto255, and P-256, often deployed in C or Rust. Hardware implementations involve deploying cryptosystems on FPGA or ASIC platforms. Although not always production-ready, these proof-of-concept systems demonstrate that cryptographic designs can be correctly translated into working implementations.

  Cryptanalysis at this layer includes laboratory-based attacks such as power analysis, fault injection, and timing attacks, typically guided by theoretical insights from Layer 2 primitives. On the software side, tools such as FastColl or HashClash can be used to exploit weaknesses in hash functions, allowing students to generate colliding outputs in practice. Other investigations may involve weaknesses in key generation, as demonstrated in incidents such as the Taiwan Citizen Digital Certificate issue

or Debian's OpenSSL vulnerability. These cases highlight the risks that arise when there is insufficient collaboration between engineers and software developers at Layer 5 and mathematicians working at Layer 1 on critical foundational aspects.

- **Interface:** . Moving beyond programmatic prototypes, the final dimension of this work is the Interface Layer, where academic identification schemes intersect with industrial realities, national governance, and rigorous compliance architectures. This layer is intentionally excluded from the five-layers of abstraction as students need not necessarily have to engage at this layer during academic studies. However, awareness that such a layer exists during the move to industry once their studies have completed will benefit their understanding of applications of cryptographic schemes within the context of a larger worldview.

The layers are briefly summarised in Table 1. The table presents the brief description for each layer, along with key concepts and outcomes.

4 Applying the Framework

To illustrate how the framework operates in practice, the first author draws on his main body of work on identification schemes without certificates to demonstrate that the framework captures the full spectrum of his research contributions. These works primarily focus on identity-based identification (IBI), much of which has been recently surveyed by Vangujar et al. [80], along with related developments in certificateless cryptography [1] and traditional identification schemes [38], extending even to associated signature schemes.

Identification schemes provide authentication through honest-verifier zero-knowledge (HVZK) protocols [38]. The identity-based variant eliminates the need for certificates by directly binding public keys to user identities [63]. When introducing this topic to students, these schemes are then mapped onto the five abstraction layers, allowing them to select the layer at which they wish to focus their research.

1. Layer 1: Bellare et al. provide one of the two initial studies on IBI, categorising schemes by the mathematical family they are transformed from (factoring, RSA, discrete logarithms, elliptic curves), though their main contribution operates at Layer 2, showing convertibility between traditional identification, signature, and identity-based schemes [9]. Identification schemes require *completeness*, *soundness*, and either *zero-knowledge* or *witness indistinguishability*. Outside the common RSA/discrete-log constructs satisfying these requirements [9], most Layer 1 contributions are standard-model schemes that use novel mathematical structures in place of a random oracle [11], notably: k-resilient matrices, progressing from passive to active/concurrent and adaptive security [45, 25, 26]; Waters-signature-based IBI [27, 67]; and Boneh-Boyen-based IBI, first proposed in [52] and later optimised in [68]. All extend towards Layer 2 and 3 with full instantiations and proofs, but their novelty lies in the Layer 1 mathematical structures used to avoid random oracles.

   Post-quantum IBI schemes also expand Layer 1, providing new lattice- and coding-theory-based assumptions [37, 83, 64]; a change of Layer 1 mathematical components requires a substantial refresh of a researcher's knowledge base even for those familiar with the higher layers. This carries risk: the IKKR encryption scheme [54], intended as a stepping stone to an efficient code-based IBI scheme, was broken by a Layer 1 attack [81], illustrating the fragility of newly proposed assumptions.

2. Layer 2: Kurosawa and Heng's initial IBI proposal operates at Layer 2, transforming BLS signatures [14] into IBI without instantiation [51]; instantiated extensions were

Table 1: Cryptographic Abstraction Layers: Definitions, Topics, and Literature Classification for Identification Schemes without Certificates

Layer	**Brief Description**	**Potential Topics / Focus**	**Literature & Contributions (Certificateless ID)**
Layer 1 Mathematical Foundations	Provides algebraic structures and computational hardness assumptions underpinning cryptography.	Number theory, lattices, elliptic curves, LWE, CDH, factoring, secret key generation	• *k-Resilient Matrix:* [45, 25, 26] • *Waters Signature:* [27, 67] • *Boneh-Boyen:* [13] • *Post-Quantum Code-Based:* [83, 64] • *Attacked Schemes:* [54] (broken by [81])
Layer 2 Primitives & Models	Defines abstract primitives alongside security notions and adversarial models.	Encryption, signatures, hashes, IND-CPA/CCA, EUF-CMA, game/simulation models	• *Baseline Transforms:* [51, 9] • *Certificateless / Implicit:* [30, 33, 29, 15] • *Security-Mediated / Revocation:* [24, 31] • *Reset-Attack Security:* [74, 23]
Layer 3 Instantiations	Realises abstract primitives with concrete algorithms and parameter choices.	AES, RSA, ECC, Kyber, Dilithium, concrete security analysis	• *Tight Proofs:* [70, 59, 58, 17, 20, 19] • *Weaker Assumptions:* [32, 34] • *Efficiency Optimizations:* [39, 68]
Layer 4 Protocols & Applications	Constructs higher-level systems by composing primitives or instantiations.	Key exchange, authentication, MPC, threshold crypto, blockchain, UC security	• *Blockchain & E-Auctions:* [78] • *VANETs & E-Voting:* [79, 76] • *Federated Learning Authentication:* [66, 65]
Layer 5 Circuits & Code	Translates designs into executable systems in software and hardware.	Cryptographic libraries, constant-time coding, hardware acceleration (FPGA/ASIC)	• *Simulators (Java/Android/iOS):* [69, 35, 47, 16] • *Hardware & Libsodium:* [72, 20, 18] • *System Integrations (Linux PAM):* [21]
Interface Cross-Cutting Boundary	Serves as an overarching contextual layer rather than a core abstraction layer; captures real-world constraints, attack vectors, and governance that influence all 5 layers.	Quality control, NIST standards, RFCs, key management, policies, patents	*Operates outside the 5 core abstraction layers; No representative research papers are categorised here because the Interface Layer represents practice rather than a research abstraction.*

later given in Heng's thesis [44] and by Yang et al. [82], while [22] surveyed attribute-based identification against reset attacks without a concrete construction until 2015. Most Layer 2 work extends the key-configuration structures of identification schemes – themselves extensions of identification into signatures via Fiat-Shamir [38], then into IBI via the Kurosawa-Heng transform [51] – and can also be derived from trapdoor sampleable relations [9], commitment schemes [53], or identity-based KEMs [7].

Further extensions towards real-world use include: hierarchical IBI [28, 75, 4]; certificateless identification [30, 33, 29], providing Trust Layer 3 per Girault [41]; security-mediated IBI for easier revocation [24, 31]; reset-attack-secure IBI [74, 23, 8]; group IBI [77, 7]; anonymous IBI [42]; IBI without trusted parties [71]; and implicit-certified IBI [15], sitting between conventional and certificateless IBI in trust level.

3. Layer 3: Almost all Layer 2 works include concrete realisations with proofs, since demonstrating novelty is difficult without instantiation. As the models and proofs are well-established here, contributions come mainly as tighter proofs, weaker assumptions, and marginal efficiency gains. Tight proofs began with Tan et al., using the DDH assumption following Katz-Wang [50] to avoid the reset lemma's square-root loss in tightness [10, 70]; Ng et al. extended this to BLS [59] and Schnorr [58], subsequently applied to IBI [17, 20], with a further 1-bit optimisation in [19]. Weaker assumptions were established via the weak discrete-log assumption on transformed Beth [12] and Schnorr [62] signatures [32, 34], while efficiency improvements progressed from Fujioka et al. [39] on [52] to a further q-SDH-based optimisation [68].

4. Layer 4: IBI has been used to construct higher-level systems in blockchain, VANETs, e-voting, and federated learning. Provable security reductions are uncommon here; papers instead rely on descriptive analyses of how the primitive functions within a larger solution. Examples include IBI for blockchain e-auctions and VANETs [78, 79], Group-IBI e-voting [76], and federated learning authentication, surveyed and then addressed using IBI by Szelag et al. [66, 65]. Layer 4 papers typically feature the primitive under idealised assumptions within a larger architecture, without full implementation detail, the main contribution being that the primitive fits and performs correctly within that architecture.

5. Layer 5: At this final layer, theoretical constructs are implemented in real-world systems, with IBI seeing iterations spanning operating-system and kernel level down to IoT sensors and gateways. Development began with Tan et al.'s Java pairing-library simulation in 2010 [69], extended to Android [35] and, via J2ObjC, to iOS [16], alongside a Java-based Schnorr-IBI performance suite [47]. Teh et al. then built a client-server prototype offloading pairing operations to a PHP/Apache server, driving a magnetic door lock on successful authentication [72], later extended with OpenCV facial recognition as the identity string. Chia et al. implemented Tight-Schnorr IBI on ARM using Curve25519/libsodium, evaluated further elliptic curves, and built a Linux PAM prototype [20, 18, 21]; Braeken et al. showed IBI can match symmetric-cryptography efficiency in IoT gateways [15]; and Szelag et al. demonstrated negligible overhead for IBI-secured federated learning [65]. Layer 5 contributions typically present proof-of-concept implementations, demonstrating that solutions can be realised programmatically as prototypes.

6. Interface: Moving beyond programmatic prototypes, the Interface Layer is where academic identification schemes meet industrial realities, national governance, and compliance architectures. For IBI and certificateless schemes, this means engaging with the statutory guidelines of Malaysia's National Cyber Security Agency (NACSA)'s Dasar Kriptografi Negara (National Cryptography Policy) [57], including

compliance with the National Trusted Cryptographic Algorithm List (MySEAL 2.0) [36], where new primitives (AKBA) or existing standards (AKSA) are vetted for quantum resilience as part of PTPKM's roadmap towards a 2030 post-quantum transition against "harvest now, decrypt later" threats [73]. Commercial deployment further requires formal assurance via Common Criteria (ISO/IEC 15408) [46], guaranteeing the Evaluation Assurance Layers required by National Critical Information Infrastructure sectors, thereby bridging theoretical validation and practical, nation-wide deployment.

This mapping suggests that the proposed framework is capable of capturing research contributions across the various layers represented within the main author's work. While illustrated through identity-based identification, the framework may also be applicable to other areas of cryptography, providing one perspective on how foundational mathematical concepts can develop into practical cryptographic systems.

We conclude this section with Figure 1 which provides a summative reflection of this work.

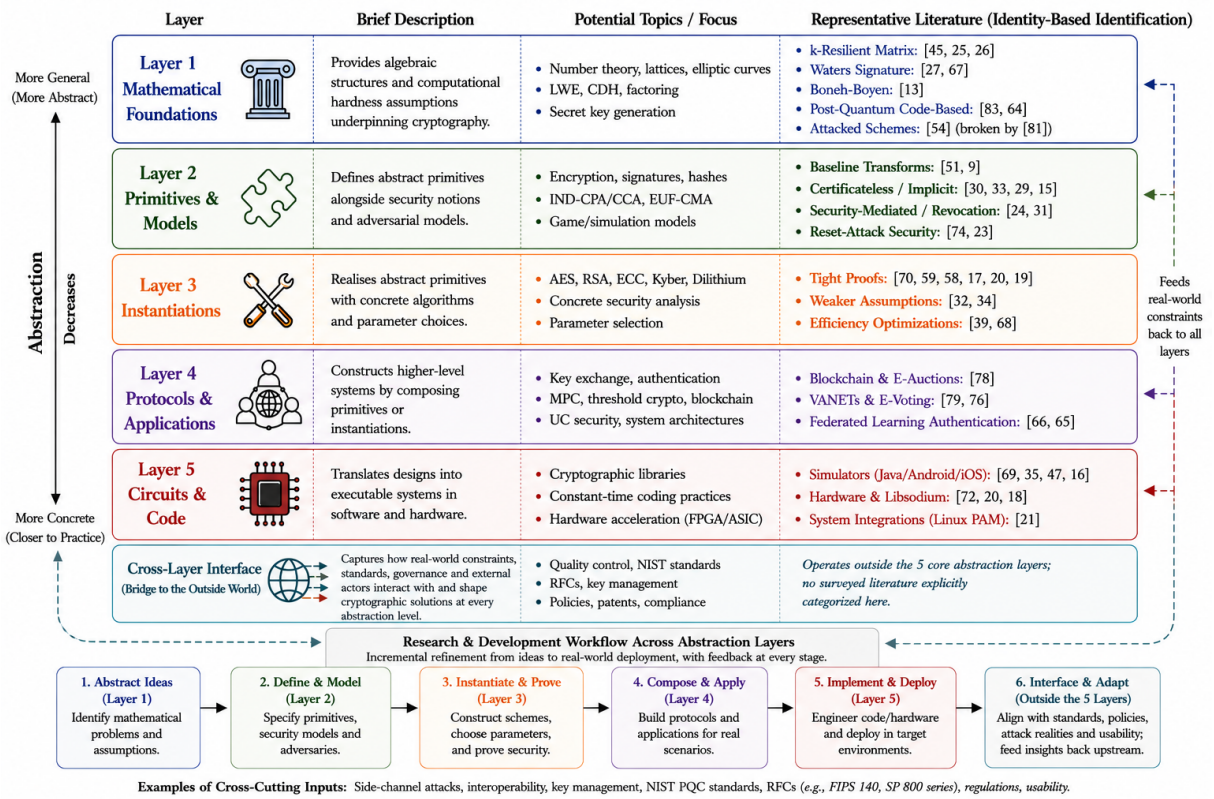


Figure 1: A five-layer framework of abstraction for cryptographic education and research, illustrated with identity-based identification (IBI). The abstraction increases from Layer 5 (most concrete) to Layer 1 (most general). The interface captures real-world feedback that influences all layers.

5 Conclusion

In this work, a five-layer abstraction framework was presented to improve how cryptographic research is approached and studied. The framework provides a structured pathway for navigating complex concepts and assists new researchers in identifying where their work fits within the broader landscape of cryptology.

This framework should be viewed as a supplementary guide rather than a replacement for textbooks. Its primary purpose is to improve conceptual understanding and help researchers navigate the complexity of the field.

Recommended textbooks include: The Handbook of Applied Cryptography by [2], Introduction to Modern Cryptography by Katz and Lindell [49], Introduction to Theoretical Computer Science by Boaz Barak [6] and finally the Foundations of Cryptography by Oded Goldreich [43].

Acknowledgments

In terms of contribution, the conceptual layering framework, examples and mapping were done by the first author. The second author provided editorial contributions to improve the readability and flow of the paper, arranging the examples in a cohesive order and generated the summaries including the creation of the summary tables and image concepts. The author thanks UKRI for supporting participation in this conference through the EPSRC Overseas Travel Fund, Fund ID UKRI4485. Both authors also express sincere gratitude to the many undergraduate and postgraduate students they have supervised over the years.

A Use of AI

Generative AI was used in initial paper searches to find relevant papers outside of the author's bibliography for the literature review, in the proofreading of this paper, and in image generation for the illustration. The application of the framework, as well as all ideas presented, are the author's own, based on his two decades of teaching and research experience in the field of IBI.

References

[1] Sattam S. Al-Riyami and Kenneth G. Paterson. Certificateless public key cryptography. In Chi-Sung Laih, editor, *Advances in Cryptology - ASIACRYPT 2003, 9th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, November 30 - December 4, 2003, Proceedings*, volume 2894 of *Lecture Notes in Computer Science*, pages 452–473. Springer, 2003.

[2] Menezes Alfred, Vanstone Scott, et al. Handbook of applied cryptography, 1997.

[3] Rachid Anane, Kevin Purohit, and Georgios Theodoropoulos. An Animated Cryptographic Learning Object. In *2008 Fifth International Conference on Computer Graphics, Imaging and Visualisation*, pages 61–68, 2008.

[4] Keita XAGAWA Atsushi FUJIOKA, Taiichi SAITO. Secure hierarchical identity-based identification without random oracles. *IEICE TRANSACTIONS on Fundamentals*, E97-A(6):1307–1317, June 2014.

[5] Aydin Aysu. Teaching the Next Generation of Cryptographic Hardware Design to the Next Generation of Engineers. In *Proceedings of the 2019 Great Lakes Symposium*

on VLSI, GLSVLSI '19, pages 237–242, New York, NY, USA, 2019. Association for Computing Machinery.

[6] Boaz Barak. Introduction to theoretical computer science. https://github.com/boazbk/tcs, 2023. GitHub repository accessed 2026-05-25.

[7] Prateek Barapatre and Chandrasekaran Pandu Rangan. Anonymous identity-based identification scheme in ad-hoc groups without pairings. In Benedikt Gierlichs, Sylvain Guilley, and Debdeep Mukhopadhyay, editors, *Security, Privacy, and Applied Cryptography Engineering*, pages 130–146, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[8] Mihir Bellare, Marc Fischlin, Shafi Goldwasser, and Silvio Micali. Identification protocols secure against reset attacks. In Birgit Pfitzmann, editor, *Advances in Cryptology — EUROCRYPT 2001*, pages 495–511, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[9] Mihir Bellare, Chanathip Namprempre, and Gregory Neven. Security proofs for identity-based identification and signature schemes. *IACR Cryptol. ePrint Arch.*, page 252, 2004.

[10] Mihir Bellare and Adriana Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In Moti Yung, editor, *Advances in Cryptology — CRYPTO 2002*, pages 162–177, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[11] Mihir Bellare and Phillip Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, CCS '93, page 62–73, New York, NY, USA, 1993. Association for Computing Machinery.

[12] T Beth. A fiat-shamir-like authentication protocol for the el-gamal-scheme. In *Advances in Cryptology-proceedings of EUROCRYPT*, volume 88, pages 77–86, 1988.

[13] Dan Boneh and Xavier Boyen. Short signatures without random oracles. In Christian Cachin and Jan L. Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004*, pages 56–73, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[14] Dan Boneh, Ben Lynn, and Hovav Shacham. Short Signatures from the Weil Pairing. In Colin Boyd, editor, *Advances in Cryptology — ASIACRYPT 2001*, pages 514–532, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[15] An Braeken, Ji-Jian Chin, and Syh-Yuan Tan. ECQV-IBI: identity-based identification with implicit certification. *J. Inf. Secur. Appl.*, 63:103027, 2021.

[16] Zi-Yik Cheah, Yik-Shu Lee, Thong-Yun The, and Ji-Jian Chin. Simulation of a pairing-based identity-based identification scheme in ios. In *2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 298–303, 2015.

[17] Jason Chia and Ji-Jian Chin. An identity based-identification scheme with tight security against active and concurrent adversaries. *IEEE Access*, 8:61711–61725, 2020.

[18] Jason Chia, Ji-Jian Chin, and Sook-Chin Yip. Evaluating pairing-free identity-based identification using curve25519. In *Advances in Cyber Security: Second International Conference, ACeS 2020, Penang, Malaysia, December 8-9, 2020, Revised Selected Papers 2*, pages 179–193. Springer, 2021.

[19] Jason Chia, Ji-Jian Chin, and Sook-Chin Yip. OrBit: OR-Proof Identity-Based Identification with Tight Security for (as Low As) 1-Bit Loss. In Qiong Huang and Yu Yu, editors, *Provable and Practical Security*, pages 116–135, Cham, 2021. Springer International Publishing.

[20] Jason Chia, Ji-Jian Chin, and Sook-Chin Yip. A pairing-free identity-based identification scheme with tight security using modified-schnorr signatures. *Symmetry*, 13(8), 2021.

[21] Jason Chia, Ji-Jian Chin, and Sook-Chin Yip. Pluggable authentication module meets identity-based identification. In *Advances in Cyber Security: Third International Conference, ACeS 2021, Penang, Malaysia, August 24–25, 2021, Revised Selected Papers 3*, pages 155–175. Springer, 2021.

[22] Ji Jian Chin, Hiroaki Anada, Seiko Arita, Kouichi Sakurai, Swee Huay Heng, and Raphael Phan. Survey and new idea for attribute-based identification scheme secure against reset attacks. *International Journal of Cryptology Research*, pages 39–47, 2014.

[23] Ji-Jian Chin, Hiroaki Anada, and Syh-Yuan Tan. Reset-Secure Identity-Based Identification Schemes Without Pairings. In Man-Ho Au and Atsuko Miyaji, editors, *Provable Security*, pages 227–246, Cham, 2015. Springer International Publishing.

[24] Ji-Jian Chin, Rouzbeh Behnia, Swee-Huay Heng, and Raphael C.-Wei Phan. An efficient and provable secure security-mediated identity-based identification scheme. In *2013 Eighth Asia Joint Conference on Information Security*, pages 27–32, 2013.

[25] Ji-Jian Chin and Swee-Huay Heng. Security upgrade for a k-resilient identity-based identification scheme in the standard model. *Malaysian Journal of Mathematical Sciences*, 7(Special Issue):73–85, 2013. Special Issue: The 3rd International Conference on Cryptology & Computer Security 2012 (CRYPTOLOGY2012).

[26] Ji-Jian Chin and Swee-Huay Heng. An adaptive-secure k-resilient identity-based identification scheme in the standard model. *International Information Institute (Tokyo).Information*, 17(1):197–207, 01 2014. Copyright - Copyright International Information Institute Jan 2014.

[27] Ji-Jian Chin, Swee-Huay Heng, and Bok-Min Goi. An efficient and provable secure identity-based identification scheme in the standard model. In Stig F. Mjølsnes, Sjouke Mauw, and Sokratis K. Katsikas, editors, *Public Key Infrastructure*, pages 60–73, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[28] Ji-Jian Chin, Swee-Huay Heng, and Bok-Min Goi. Hierarchical identity-based identification schemes. In Dominik Ślęzak, Tai-hoon Kim, Wai-Chi Fang, and Kirk P. Arnett, editors, *Security Technology*, pages 93–99, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[29] Ji-Jian Chin, Swee-Huay Heng, and Raphael C.-W. Phan. An efficient and provable secure certificateless identification scheme in the standard model. *KSII Transactions on Internet and Information Systems*, 8(7):2532–2553, July 2014.

[30] Ji-Jian Chin, Raphael C.-W. Phan, Rouzbeh Behnia, and Swee-Huay Heng. An efficient and provably secure certificateless identification scheme. In Pierangela Samarati, editor, *SECRYPT 2013 - Proceedings of the 10th International Conference on Security and Cryptography, Reykjavík, Iceland, 29-31 July, 2013*, pages 371–378. SciTePress, 2013.

[31] Ji-Jian Chin, Syh-Yuan Tan, Swee-Huay Heng, and Raphael C.-W. Phan. Efficient and provable secure pairing-free security-mediated identity-based identification schemes. *The Scientific World Journal*, 2014(1):170906, 2014.

[32] Ji-Jian Chin, Syh-Yuan Tan, Swee-Huay Heng, and Raphael C.-W. Phan. Twin-beth: Security under active and concurrent attacks for the beth identity-based identification scheme. *Cryptogr. Commun.*, 8(4):579–591, 2016.

[33] Ji-Jian Chin, Syh-Yuan Tan, Swee-Huay Heng, Raphael C.-W. Phan, and Rouzbeh Behnia. A provable secure pairing-free certificateless identification scheme. *International Journal of Computer Mathematics*, 92(8):1520–1535, 2015.

[34] Ji-Jian Chin, Syh-Yuan Tan, Swee-Huay Heng, and Raphael Chung-Wei Phan. Twin-schnorr: A security upgrade for the schnorr identity-based identification scheme. *The Scientific World Journal*, 2015(1):237514, 2015.

[35] Ji-Jian Chin, Syh-Yuan Tan, Yvonne Hwei-Syn Kam, and Chee-Hoe Leong. Implementation of identity-based and certificateless identification on android platform. In *Conference Proceedings - Cryptology 2014: Proceedings of the 4th International Cryptology and Information Security Conference 2014*, page 57 – 64, 2014.

[36] CyberSecurity Malaysia. Introduction to the national trusted cryptographic algorithm list (myseal). https://mykripto.cybersecurity.my/index.php/services-2/myseal/introduction, 2021. Accessed: 2026-05-25.

[37] Sidi Mohamed El Yousfi Alaoui, Pierre-Louis Cayrel, and Meziani Mohammed. Improved identity-based identification and signature schemes using quasi-dyadic goppa codes. In Tai-hoon Kim, Hojjat Adeli, Rosslin John Robles, and Maricel Balitanas, editors, *Information Security and Assurance*, pages 146–155, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[38] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *Advances in Cryptology — CRYPTO' 86*, pages 186–194, Berlin, Heidelberg, 1987. Springer Berlin Heidelberg.

[39] Atsushi Fujioka, Taiichi Saito, and Keita Xagawa. Security Enhancements by OR-Proof in Identity-Based Identification. In Feng Bao, Pierangela Samarati, and Jianying Zhou, editors, *Applied Cryptography and Network Security*, pages 135–152, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[40] Moisés García-Martínez, Sergio Rolando Cruz-Ramírez, and José Manuel Olais-Govea. An Early Introduction to Cryptography with Engineering Students. In Samira Hosseini, Diego Hernan Peluffo, Julius Nganji, and Arturo Arrona-Palacios, editors, *Technology-Enabled Innovations in Education*, pages 97–118, Singapore, 2022. Springer Nature.

[41] Marc Girault. Self-certified public keys. In Donald W. Davies, editor, *Advances in Cryptology — EUROCRYPT '91*, pages 490–497, Berlin, Heidelberg, 1991. Springer Berlin Heidelberg.

[42] VT Goh, Ji Jian Chin, R Kannan, and Sook Chin Yip. A pairing-free provable secure and efficient identity-based identification scheme with anonymity. *Malaysian Journal of Mathematical Sciences*, 17(4):19, 2023.

[43] Oded Goldreich. *Foundations of Cryptography*. Cambridge University Press, 2004.

[44] Swee-Huay Heng. *Design and Analysis of some cryptographic primitives*. PhD Thesis, Graduate School of Science and Engineering, Tokyo Institute of Technology, Tokyo, Japan, 2004. Publication Title: Public Key Cryptography – PKC 2004.

[45] Swee-Huay Heng and Ji-Jian Chin. A k-resilient identity-based identification scheme in the standard model. *International Journal of Cryptology Research*, 2(1):15–25, 2010.

[46] International Organization for Standardization. Information technology — security techniques — evaluation criteria for it security (common criteria), 2022.

[47] YHS Kam, JJ Chin, and SY Tan. The schnorr-suite: Simulation of pairing-free identity-based identification schemes using java. In *2015 3rd International Conference on Software Engineering, Knowledge Engineering and Information Engineering*, pages 13–18, 2015.

[48] Christopher Kane, Bo Lin, Saksham Chand, Scott D. Stoller, and Yanhong A. Liu. High-Level Cryptographic Abstractions. In *Proceedings of the 14th ACM SIGSAC Workshop on Programming Languages and Analysis for Security*, PLAS'19, pages 31–43, New York, NY, USA, 2019. Association for Computing Machinery.

[49] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography: Revised Third Edition*. Chapman and Hall/CRC, Boca Raton, 3rd edition, 2025.

[50] Jonathan Katz and Nan Wang. Efficiency improvements for signature schemes with tight security reductions. In *Proceedings of the 10th ACM Conference on Computer and Communications Security*, CCS '03, page 155–164, New York, NY, USA, 2003. Association for Computing Machinery.

[51] Kaoru Kurosawa and Swee-Huay Heng. From digital signature to id-based identification/signature. In Feng Bao, Robert Deng, and Jianying Zhou, editors, *Public Key Cryptography – PKC 2004*, pages 248–261, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[52] Kaoru Kurosawa and Swee-Huay Heng. Identity-based identification without random oracles. In *Proceedings of the 2005 International Conference on Computational Science and Its Applications - Volume Part II*, ICCSA'05, page 603–613, Berlin, Heidelberg, 2005. Springer-Verlag.

[53] Kaoru Kurosawa and Swee-Huay Heng. The power of identification schemes. In Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin, editors, *Public Key Cryptography - PKC 2006*, pages 364–377, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[54] Terry Shue Chien Lau, Fedor Ivanov, Muhammad Rezal Kamel Ariffin, Ji-Jian Chin, and Timothy Tzen Vun Yap. New code-based cryptosystems via the ikkr framework. *Journal of Information Security and Applications*, 76:103530, 2023.

[55] Ueli Maurer. Abstraction in Cryptography. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009*, pages 465–465, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[56] Ueli Maurer. Constructive Cryptography – A New Paradigm for Security Definitions and Proofs. In Sebastian Mödersheim and Catuscia Palamidessi, editors, *Theory of Security and Applications*, pages 33–56, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[57] National Cyber Security Agency (NACSA). Dasar kriptografi negara (national cryptography policy). Policy Document MyKriptografi-NCP, National Security Council, Prime Minister's Department, 2018. Accessed: 2026-05-25.

[58] Tiong-Sik Ng, Syh-Yuan Tan, and Ji-Jian Chin. A variant of schnorr signature scheme with tight security reduction. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 411–415, 2017.

[59] Tiong-Sik Ng, Syh-Yuan Tan, and Ji-Jian Chin. A Variant of BLS Signature Scheme with Tight Security Reduction. In Jiankun Hu, Ibrahim Khalil, Zahir Tari, and Sheng Wen, editors, *Mobile Networks and Management*, pages 150–163, Cham, 2018. Springer International Publishing.

[60] Frank Piessens. Security across abstraction layers: old and new examples. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 271–279, 2020.

[61] Rich Schlesinger. A cryptography course for non-mathematicians. In *Proceedings of the 1st Annual Conference on Information Security Curriculum Development*, InfoSecCD '04, pages 94–98, New York, NY, USA, 2004. Association for Computing Machinery.

[62] C. P. Schnorr. Efficient identification and signatures for smart cards. In Gilles Brassard, editor, *Advances in Cryptology — CRYPTO' 89 Proceedings*, pages 239–252, New York, NY, 1990. Springer New York.

[63] Adi Shamir. Identity-based cryptosystems and signature schemes. In George Robert Blakley and David Chaum, editors, *Advances in Cryptology*, pages 47–53, Berlin, Heidelberg, 1985. Springer Berlin Heidelberg.

[64] Bo Song and Yiming Zhao. Provably secure identity-based identification and signature schemes from code assumptions. *PLOS ONE*, 12(8):1–15, 08 2017.

[65] Jakub Kacper Szeląg, Ji-Jian Chin, Lauren Ansell, and Sook-Chin Yip. Integrating identity-based identification against adaptive adversaries in federated learning. *IEEE Access*, 2025.

[66] Jakub Kacper Szeląg, Ji-Jian Chin, and Sook-Chin Yip. Adaptive adversaries in byzantine-robust federated learning: A survey. Cryptology ePrint Archive, Paper 2025/510, 2025.

[67] Syh-Yuan Tan, Ji-Jian Chin, Swee-Huay Heng, and Bok-Min Goi. An improved efficient provable secure identity-based identification scheme in the standard model. *KSII Trans. Internet Inf. Syst.*, 7(4):910–922, 2013.

[68] Syh-Yuan Tan and Swee-Huay Heng. New identity-based identification and signature schemes in the standard model. In *Proceedings of the 18th International Conference on Availability, Reliability and Security*, ARES '23, New York, NY, USA, 2023. Association for Computing Machinery.

[69] Syh-Yuan Tan, Swee-Huay Heng, and Bok-Min Goi. Java implementation for pairing-based cryptosystems. In *International Conference on Computational Science and Its Applications*, pages 188–198. Springer, 2010.

[70] Syh-Yuan Tan, Swee-Huay Heng, Raphael C. W. Phan, and Bok-Min Goi. A variant of schnorr identity-based identification scheme with tight reduction. In Tai-hoon Kim, Hojjat Adeli, Dominik Slezak, Frode Eika Sandnes, Xiaofeng Song, Kyo-il Chung, and Kirk P. Arnett, editors, *Future Generation Information Technology*, pages 361–370, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[71] Fei Tang, Jiali Bao, Yonghong Huang, Dong Huang, and Fuqun Wang. Identity-based identification scheme without trusted party against concurrent attacks. *Security and Communication Networks*, 2020(1):8820271, 2020.

[72] Thong-Yun Teh, Yik-Shu Lee, Zi-Yik Cheah, and Ji-Jian Chin. Ibi-mobile authentication: A prototype to facilitate access control using identity-based identification on mobile smart devices. *Wirel. Pers. Commun.*, 94(1):127–144, 2017.

[73] The Vibes. Nation to consolidate cyber defences under new central agency, says PM. *The Vibes*, February 2026. Accessed: 2026-05-25.

[74] Pairat Thorncharoensri, Willy Susilo, and Yi Mu. Identity-based identification scheme secure against concurrent-reset attacks without random oracles. In Heung Youl Youm and Moti Yung, editors, *Information Security Applications*, pages 94–108, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[75] AK Vangujar, Ji Jian Chin, Sin Yin Tan, and TS Ng. A hierarchical identity-based identification scheme without pairing. *Malaysian Journal of Mathematical Sciences*, 13:93–109, 2019.

[76] Apurva K. Vangujar, Buvana Ganesh, Alia Umrani, and Paolo Palmieri. A novel approach to e-voting with group identity-based identification and homomorphic encryption scheme. *IEEE Access*, 12:162825–162843, 2024.

[77] Apurva K Vangujar, TS Ng, J Chia, JJ Chin, and SC Yip. Group identity-based identification: Definitions, construction and implementation. *Malaysian Journal of Mathematical Sciences*, 15, 2021.

[78] Apurva K. Vangujar, Alia Umrani, Ji-Jian Chin, and Paolo Palmieri. Secure e-auctions: A blockchain-based cluster consensus identity-based identification scheme. In *2024 6th International Conference on Blockchain Computing and Applications (BCCA)*, pages 189–197, 2024.

[79] Apurva K Vangujar, Alia Umrani, and Paolo Palmieri. ID-CAKE: Identity-based cluster authentication and key exchange scheme for message broadcasting and batch verification in VANETs. Cryptology ePrint Archive, Paper 2023/1835, 2023.

[80] Apurva Kiran Vangujar, Paolo Palmieri, Ji-Jian Chin, and Swee-Huay Heng. Two decades of identity-based identification schemes- a survey on challenges and advances. Cryptology ePrint Archive, Paper 2026/577, 2026.

[81] Kirill Vedenev. On the security of two IKKR-type code-based cryptosystems. *Designs, Codes and Cryptography*, 93(11):4779–4798, November 2025.

[82] Guomin Yang, Jing Chen, Duncan S. Wong, Xiaotie Deng, and Dongsheng Wang. A new framework for the design and analysis of identity-based identification schemes. *Theor. Comput. Sci.*, 407(1-3):370–388, 2008.

[83] Guomin Yang, Chik How Tan, Yi Mu, Willy Susilo, and Duncan S. Wong. Identity based identification from algebraic coding theory. *Theoretical Computer Science*, 520:51–61, 2014.

[84] Moti Yung. Cryptographic Protocols: From the Abstract to the Practical to the Actual. In George Danezis, Sven Dietrich, and Kazue Sako, editors, *Financial Cryptography and Data Security*, pages 1–2, Berlin, Heidelberg, 2012. Springer.

10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 347–365.

Chaos-based Double Image Encryption using 4D Henon-type Hyperchaotic Map for Colour Images

Syahidatul Shafiqah Ramlee[1] and Arif Mandangan[2,3]

[1] Mathematics Computer Graphics Programme, Faculty of Science and Technology, Jalan UMS, 88400 Kota Kinabalu, Sabah, Malaysia syahidatulshafiqahramlee@gmail.com
[2] Mathematics Visualization Research Group, Faculty of Sciences and Technology, Universiti Malaysia Sabah, Jln UMS, 88400 Kota Kinabalu Sabah, Malaysia arifman@ums.edu.my
[3] Malaysia Cryptology Technology and Management Centre, c/o Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia arifman@ums.edu.my

Abstract. Image encryption becomes more significant due to image being one of the most used data forms in communication and information exchange. To ensure high confidentiality, cipher image produced must be of excellent quality to prevent information leakage. As image being largely used in information exchange, a large production for cipher image is demanded, this large storage for cipher image is highly necessary as to prevent unwanted effect on the production from producing high-quality cipher image. To address this, a scheme processing multiple images simultaneously comes in handy as a fitting action to handle the convenience of storage and transmission of many images. Inspired by that, a modified encryption scheme for double images using a hyperchaotic map derived from 4D Henon-type map is proposed. Using this idea, two coloured images will be encrypted to produce one cipher image. The scheme begins by combining the two images and then applying the hyperchaotic map to encrypt the combined images. This approach not only enhance the convenience of the scheme by processing multiple images at once but also enhance the encryption quality in terms of image quality and preservation. This can be proven by the results obtained from evaluating the image quality where it shows the proposed cryptosystem generates a very high-quality cipher image with average PSNR values as low and small at 7.67197, while also proven to be reliably preserving the quality of the plain image with perfect SSIM score at 1.0. The proposed cryptosystem also yields preferred ideal NPCR value averagely at 99.61389% and UACI at 33.45574%. This says that the proposed cryptosystem can effectively resist common cryptographic attacks such as differential attacks. The scheme is also computationally efficient, exhibiting low computational cost with average execution time of less than a second.

Keywords: cipher images · double images · hyperchaotic map · 4D Henon map

1 Introduction

In this digital era, our dependence on the internet for data storage and communication has made cryptography a very essential pillar for cybersecurity. Cryptography plays a crucial role in cyber realms, offering protection and security for the data. Without these protective measures, our data and information remain highly vulnerable and exposed to malicious activities and various kinds of attacks including ransomware attacks which are one of the most alluring threats to cybersecurity [1]. In cyber realms, digital image becomes the popular media and demanded one in conveying messages as it can convey information and contents that cannot be expressed by words. Therefore, a study to protect these images in

the cyber realms and its schemes development becomes the popular hot topic in the study, known as image encryption [2].

Image encryption is one of the cryptosystem techniques in cryptography used to encode and generate secret images known as cipher images. It uses encryption algorithm to prevent access from unauthorized parties. A cipher image should be generated to be completely hidden and unreadable. This is to prevent information leaks as to secure the integrity of the images. Only the parties who own the key to the algorithm will have the access to the images [3]. The procedure in image encryption is to convert a plain image, which is the readable image, to a cipher image, the unreadable image. The conversion can only be performed using a key known as secret key or private key. The procedure to recover the cipher image back to its readable image is known as the decryption of the cipher image. The same key used in the encryption should be used again to decrypt the cipher image [4].

On the other hand, chaotic map is roughly defined as a system with chaotic behaviours. The concept behind chaotic theory is unpredictable and random. It is dependent on the controlling parameters. The arrangement of the parameters is to confuse from locating the outputs which will make it impossible to locate if the parameters is unknown [5]. The dependency on the controlling parameters makes chaotic map to be very sensitive towards any change of the parameters, thus making chaotic map an ideal scheme for image encryption [6].

There are several methods developed to encrypt images using chaotic system. This paper proposed a method using the 4D Henon-type hyperchaotic map which was derived from the common multidimensional which is Henon chaotic map. Henon map is a discrete dynamic chaotic map that exhibits its chaotic behaviour based on its sensitivity to the initial conditions. The randomness of the chaotic behaviour of the maps depends on the value of the parameters which are called as control parameters. Due to its effective characteristic such as randomness of behaviour and uniform non-variation of density variables, Henon map becomes a strong option in cryptography [7].

A high-dimensional chaotic map yields better performance and more chaos chaotic behaviour than one-dimensional map. A one-dimensional chaotic map has a simple structure of chaotic orbits, whereas high-dimensional chaotic map has inherently complex structures thus giving better performance. In addition, one high-dimensional chaotic map is determined to be equally significant to several combined one-dimensional chaotic maps [8]. Henon map is classified as high-dimensional chaotic map.

In the previous study in [8], a fast chaos-based image encryption scheme using the 3D Henon chaotic map was proposed. It stated that the trajectory characteristics of the 3D chaotic map improves the unpredictability of the encryption process thus enforces superior randomness characteristic. The study results in a strengthened security on par with advanced algorithms and improved performance reliable in producing unrecognizable chaotic sequences.

A hyper version of the 3D Henon chaotic map has a chaotic behaviour that is more vigorous reaching hyper states and higher complexity and unpredictability. A previous study by [9] proposed a chaos-based image encryption scheme using this hyper version of 3D Henon chaotic map or simply referred as the 3D hyper-chaotic Henon map. The study claimed that the scheme using the hyper-chaotic map is reliable and offered better performance in various aspects including key space, histogram variance, and correlation coefficient as well as its security and the outputs.

Henon map with higher dimension such as 4D Henon chaotic map contains more control parameters initiated on more variable points. The dynamical behaviour of the map with more variable points becomes more chaotic than a map with smaller dimensions. Another points, the map that contains more positive controlling parameters becomes hyper in chaotic behaviour, thus making it a hyperchaotic map [10]. A previous study by [11] introduced an improved design of the 4D Henon hyperchaotic map derived from

the original map specifically simplifying the memristors, conveniently referred as the 4D-HMM map. The study stated that the proposed scheme resulted in excellent encryption metrics including achieving uniform histograms and ideal values of information entropy, as well as highly capable in resisting statistical, differential, and brute-force attacks. More importantly, the scheme is well maintained in terms of computational reliability, making it reliably suitable for modern communication as it provides both high efficiency and strong cryptographic security.

Images have been one of the most used multimedia in information exchanges, which demands the increasing production of cipher images. However, most of the existing literatures in this area are meant for single image. Therefore, processing multiple images can be the course of action to satisfy the demands for large production. To produce cipher images, processing two or more images simultaneously is a proper way to realize the convenience of encryption, storage, and transmission of multiple images [12]. To put it simply, more cipher images can be generated by the same usual amount of encryption is performed. Processing multiple images at a time also reduces the need and requirement for extra transmission due to the measurement matrix being the secret key in the secret communication [13].

In [14], an encryption algorithm for double images based on spatiotemporal chaos and DNA operations is proposed. The study stated that the developed double image encryption has improved the encryption security and speed. Then, a double image encryption based on compression by combining hyperchaotic map and compressive sensing is proposed in [15]. Their developed algorithm for double images was shown to have high transmission efficiency, as well as high-rate production of good quality cipher image and has effective decryption for favourable image recovery. In [13], a double image encryption scheme is proposed that was based on compressive sensing and elliptic curve using a hyperchaotic map. The study reported that the scheme could save about half of the data transmission even after being compressed to encrypt two images simultaneously. A study by [2] proposed an encryption algorithm for double images based on parallel compressed sensing and chaotic systems. They stated in their paper that the double image encryption scheme can reduce the data and save storage space. Additionally, the paper reported that their scheme yields a more confusing encrypted image and it has the advantages of exceptional robustness to noise and attacks as well as having excellent image reconstruction quality.

There are many goods applications of similar encryption module. For example, in medical fields, when staffs need to transmit multiple medical images to patients. To reduce the inconvenience of storage and impracticality, they may use encryption scheme for multiple images to encrypt multiple medical images without having to do the encryption multiple times or even multiple transmission for every single image [13]. On the other hand, an encryption scheme with strong effectiveness can also provide confidentiality to the sensitive data and medical images of a patient that required protection especially during its transmission through the network, which can be achieved through a strong hyperchaotic system-based algorithm [16]. Another common application is the Internet of Things (IoT) where encryption is very crucial. The IoT accelerated the network connectivity and enabled the digitalization of things across multiple industries through the connected applications. The increasing of connected devices worries the possibility of data leaking and that is when image encryption comes in handy to secure sensitive information and avoid data breaching [17]. Since digital image becomes the high in demand multimedia in data sharing, a scheme for encrypting multiple images serves even more important purpose to realize the practicality of processing multiple images of great number of images [18].

In this paper, an encryption method is proposed for double images using a hyperchaotic map of 4D Henon-type map. The aim for this method is to enhance the quality of the cipher image by encrypting double images using the hyperchaotic map, to make the cipher image more noised and unreadable, to disguise the visual information of the plain images.

This method starts by combining the two plain images to achieve one combination, then followed by the encryption step using the chosen hyperchaotic map.

This study embarks on few objectives including to develop a cryptosystem for encrypting double-coloured images using 4D Henon-type hyperchaotic map, to analyse its performance and the quality of the image products, as well as evaluating the sensitivity of the scheme. The importance of this study is to improve the reliability of the developed cryptosystem in producing a good cipher image. In addition to that, the cryptosystem is designed to maintain the original quality of the plain image after undergoing the encryption process.

2 Methodology

2.1 XOR Operation

The proposed encryption scheme involves the use of two plain coloured images with the same dimension, Image 1 and Image 2. The images are set to be square in shape with the width W, and the height H, are of the same values. The images will be read in colour consisting of RGB (red-green-blue) pixels. To combine the coloured image using XOR (exclusive OR) function, unlike grayscale image where it splits into black and white, the pixels of the image are split into three separate colour values, which are red, green, and blue. In result, each split of the same share appears as a set of random colour values [19].

The XOR function is operated using some simple but powerful calculation. Say a pixel value is A, then the XOR function operates unique properties defined as such:

1. Operating with itself returns 0;

$$A \oplus A = 0 \tag{1}$$

2. Operating with 0 returns itself;

$$A \oplus 0 = A \tag{2}$$

3. Operating with other two same values return itself;

$$A \oplus (B \oplus B) = (A \oplus B) \oplus B = A \oplus 0 = A \tag{3}$$

In RGB colour channel, each pixel is operated with random other values from the different share. The properties are applied independently across the three distinct colour values where individual pixel channel bytes are operated against corresponding values from the secondary image matrix.

Aside from realizing the convenience that it brings in producing one cipher image from the bit-wise combination of two plain images, the symmetric properties of the XOR function ensures that the exact same algorithm are deployed for both the encryption and decryption processes, as to maintain a perfect recovery of the plain image during the decryption and separation processes. Thus, it can significantly reduce the complexity of the system's computation and execution. It also enables efficient image reconstruction that preserves the security features of visual cryptography, ensuring both high security and high-quality image restoration [19].

2.2 4D Henon Hyperchaotic Map

A hyperchaotic map from the improved 4D Henon-type hyperchaotic map in [11] is used in the encryption process of the scheme. The four-dimensional extension of the hyperchaotic map consists of four positive control parameters initiated on four variable dimensions. The algorithm is given by:

$$x_{n+1} = 1 + ((c + d\sin(w_n))y_n) - (a(c + d\sin(z_n)(x_n)^2), \tag{4}$$

$$y_{n+1} = b(x_n), \tag{5}$$

$$z_{n+1} = z_n + k(x_n), \tag{6}$$

$$w_{n+1} = w_n + k(y_n), \tag{7}$$

where $x_n, y_n, z_n, w_n \in \mathbb{R}$ represents the four initial variables with $n = 0, 1, 2, 3, ..., L$ and $x_{n+1}, y_{n+1}, z_{n+1}, w_{n+1} \in \mathbb{R}$ represents the next iterated variables (which are the chaotic sequence generated through the iteration), while $a, b, c, d, k \in \mathbb{R}$ represents the positive control parameters. The algorithm is initialized on value 0.1 for each initial variable, and the selected value for each parameter should be positive.

Initial state variables: $x = 0.1, y = 0.1, z = 0.1, w = 0.1$

Control parameters: $a = 0.001, b = 1.8, c = 0.5, d = 0.4, k = 0.1$

The same hyperchaotic sequence generated using the same selected values are also used in the decryption process. Eventually, a small change in the values will lead to different behaviours. The sequence generation step is described as in Algorithm 1.

Algorithm 1: **Variables and Keystream Generation**

Input: Total sequence length, L, of Plain Image 1, I_1.

Output: Keystream sequence K of length L.

Steps:

Phase 1: Key Initialization

1) Assign control parameters $\mathbf{P}$: $a, b, c, d, k \leftarrow 0.001, 1.8, 0.5, 0.4, 0.1$

2) Set initial key variables $\mathbf{X}_0$: $x_0, y_0, z_0, w_0 \leftarrow 0.1, 0.1, 0.1, 0.1$

Phase 2: Keystream Hyperchaotic Sequence Generation

1) Initialize an empty 1D array K to accumulate the output key bytes.

2) For each iteration n from 1 to L, compute:

$$x_{n+1} = 1 + ((c + d\sin(w_n)) \cdot y_n) - (a \cdot (c + d\sin(z_n) \cdot x_n^2))$$

$$y_{n+1} = b \cdot x_n$$

$$z_{n+1} = z_n + k \cdot x_n$$

$$w_{n+1} = w_n + k \cdot y_n$$

3) Generate an 8-bit key value:

$$K_n = \lfloor |x_{n+1}| \times 10^{13} \rfloor \pmod{256}$$

4) Append K_n to the sequence matrix K.

5) Return the keystream sequence K.

2.3 Double Image Encryption

The encryption process is executed on the combined image produced by the XOR operation. The pixel of the combined image is randomized by the generated keystream hyperchaotic sequence using the 4D Henon hyperchaotic map. The dimension of the combined image has a width equal to its height, $W = H$. Therefore, the dimension of the combined image should be $H \times H$. The image after the randomization is reshaped back into the initial dimension of $H \times H$. Then, the cipher image is produced. The algorithm steps for the encryption process are described as in Algorithm 2.

2.4 Image Decryption and Separation

The decryption process is operated on the generated cipher image by reversing the steps of the encryption process using the same 4D Henon hyperchaotic sequence. The pixel of

the cipher image is scattered to a reverse randomization and restored back to its original position. The image after the reverse randomization is then reshaped into its initial dimension $H \times H$.

The decrypted image is the recovered image of the combined image. To recover the two plain images, the decrypted image will be separated from the combination using the same XOR operation previously used to combine them. To recover one of the images, the decrypted image will be operated with the other image. To put it simply, Image 1 will be recovered by operating the cipher image with Image 2 and vice versa. The algorithm steps for decrypting and separating the images are described as in Algorithm 3.

Algorithm 2:	**Image Encryption**
Input:	Plain Image 1 and 2, I_1 and I_2, Parameters, P, Initial Keys, X_0.
Output:	Cipher Image Matrix, C.
Steps:	**Phase 1: Image Combination** 1) Resize I_2 to match the spatial width, W, and height, H, of I_1 by satisfying the condition of a squared matrix window shape: $(W = H)$. 2) Combine the two images using pixel-by-pixel bitwise XOR operation: $I_{combined} = I_1 \oplus I_2$
	Phase 2: Keystream Sequence Generation 1) Flatten $I_{combined}$ into a linear 1D vector F_{1D} of total length $L = W \times H \times 3$. 2) Set $x_1, y_1, z_1, w_1 \leftarrow x_0, y_0, z_0, w_0$. 3) Generate the hyperchaotic keystream matrix sequence K by executing Algorithm 1 with length L.
	Phase 3: Diffusion Masking 1) Encrypt the flattened data array via bitwise diffusion masking: $C_{1D} = F_{1D} \oplus K$ 2) Reshape C_{1D} back into the original structural 3D image matrix to produce Cipher Image C. 3) Return C

Algorithm 3:	**Image Decryption and Separation**
Input:	Cipher Image, C, Plain Image 2, I_2, Parameters, P, Initial Keys, X_0.
Output:	Recovered Plain Image 1, RI_1.
Steps:	**Phase 1: Keystream Reconstruction & Decryption** 1) Flatten the cipher image matrix C into linear 1D vector C_{1D} of length L. 2) Regenerate the identical 8-bit keystream sequence K by repeating the exact chaotic loop from Algorithm 1 using matching key parameters. 3) Decrypt the structural pixels using the symmetric bitwise XOR operation: $D_{1D} = C_{1D} \oplus K$ 4) Reshape D_{1D} back into the 3D matrix shape to yield the decrypted combined image D.
	Phase 2: XOR Separation 1) Recover the target Plain Image 1 by executing a bitwise XOR operation between D and the known plain image I_2: $RI_1 = D \oplus I_2$ 2) Return the restored image RI_1.

Table 1: The input plain images: Image 1

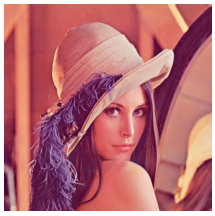
Airplane 512 x 512	Barbara 512 x 512	Boats 576 x 576	Goldhill 576 x 576
			
Sailboat 512 x 512	Lena 512 x 512	Pepper 512 x 512	Zelda 576 x 576

Table 2: The assigned second image: Cover Image

Baboon 480 x 480

3 Results

This section discusses how the test outputs of the proposed scheme are tested and evaluated. An analysis was conducted on several sample images widely used in image processing. Each image is in colour. A single image will be assigned as the second image acting as Image 2 for each test, simply referred to as the Cover Image. Therefore, all the images will act as Image 1, which to be combined with Image 2 or the Cover Image, thus producing one single cipher image. Each image is with their respective dimensions. Table 1 shows the plain images used to act as the first image, and Table 2 shows the assigned cover image to act as the second image.

3.1 Encryption Outputs

The encryption process was carried by using two plain images. These two images are combined simultaneously, where one image is acting as the first image or the Image 1, to be combined with the assigned second image which is the Cover Image. Therefore, each image will undergo respective tests with the same second image, thus producing different

cipher images even with the same second image. While both the first and second plain image can be used in the communication respectively, to ensure accurate evaluation, in this section, only one image will be in the analysis which is the first image.

Table 3 shows all the inputs with their respective outputs including both the after product of XOR combination and the cipher images. The encryption process begins with the first original image acting as Image 1 combined with the second image acting as the Cover Image using the XOR operation. As seen in the table, an image which is image Baboon have been assigned to be the cover image for all the other images. Each combination process will produce different after product Combined Image as it is with different Image 1. The randomizing of the image pixels then will be performed on the combined image to generate the Cipher Image.

4 Analysis

4.1 Image Quality Analysis

The analysis was conducted on the cipher image to evaluate the performance of the proposed scheme in producing reliable cipher images. Measurements like Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measurement (SSIM) are used in the analysis to compare the difference of the cipher image from its original plain form. In general, lower values for PSNR and SSIM are favourable as it indicates the high difference of the noise and similarity, respectively, between the two compared images. A favoured cipher images should appear as random noise or random pattern without any recognizable features, hence the best in quality.

In general, PSNR compares the noise level between two images. The range value of PSNR starts from 0, with lower values below 10 indicates that the images have high difference levels of noise, thus also difference in appearance. The computation of PSNR is as follows [20]:

$$\text{PSNR} = 20 \times \log_{10} \left(\frac{\text{Maxpixel}^2}{\text{MSE}} \right) \tag{8}$$

where Maxpixel is the maximum supported pixel value which is 255 and MSE is the mean squared error defined as:

$$\text{MSE} = \frac{1}{n} \sum_{i=0}^{H-1} \sum_{j=0}^{L-1} [Q_{i,j} - Z_{i,j}]^2 \tag{9}$$

where n represents the total number of pixels in the image, $Q_{i,j}$ is the pixel value of the plain image, and $Z_{i,j}$ is the pixel value of the decrypted image.

Meanwhile, SSIM measures the similarity between two images. The range value of SSIM is [0, 1] with 1 indicates perfect similarity and 0 indicates complete no similarity. The smaller and closer the value to 0, the lower the similarities, therefore the preferred score for cipher images. The computation of SSIM is as follow [21]:

$$\text{SSIM} = \frac{(2Mv_xMv_y + c)\,(2Cv_{xy} + c')}{((Mv_x)^2 + (Mv_y)^2 + c)\,((Sv_x)^2 + (Sv_y)^2 + c')} \tag{10}$$

where Mv_x and Mv_y are the mean values of x and y respectively, Sv_x and Sv_y are the value for standard deviation of x and y respectively, and Cv_{xy} is the covariance value of x and y. Being $c = (kL)^2$ where $k = 0.01$ and $L = (2^n - 1)$, whereas $c' = (k'L)^2$ where $k = 0.03$ and $L = (2^n - 1)$ with both n as the number of bits per pixel.

To evaluate the cipher images, the PSNR values are favoured to be small or in range 0-10, which indicates the high difference of noise between the cipher images and the plain

Table 3: The output for each encryption process.

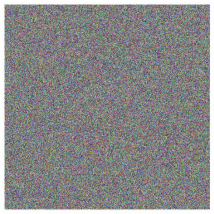
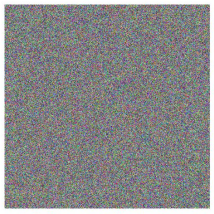
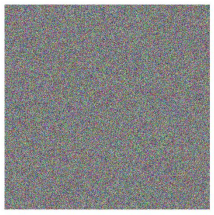
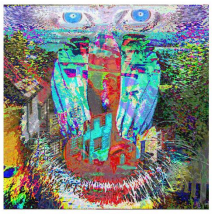
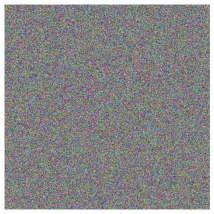
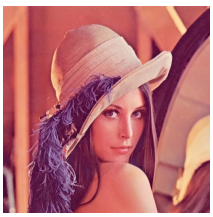
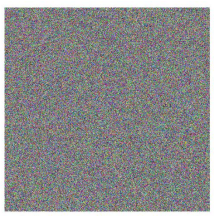
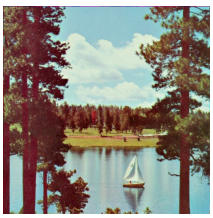
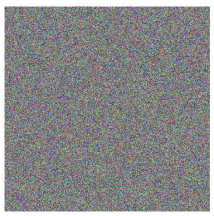
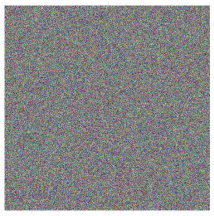
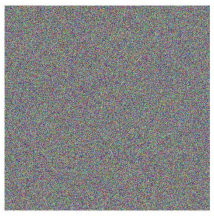
Image 1	Image 2	XOR-ed Image	Cipher Image
			
			
			
			
			
			
			
			

Table 4: PSNR and SSIM values of the cipher images produced by each encryption process.

Image 1	PSNR	SSIM
Airplane	7.67265	0.00946
Barbara	7.67309	0.00921
Boats	7.66442	0.00893
Goldhill	7.67861	0.00877
Lena	7.67651	0.00898
Sailboat	7.66658	0.00886
Pepper	7.67308	0.00810
Zelda	7.67078	0.00881
Average	**7.67197**	**0.00889**

images. SSIM values are also favoured to be smaller and close to 0 which indicate that the cipher images are hugely very different from the original images. Table 4 shows the PSNR and SSIM values measured between the first original images, Image 1, and cipher images produced by each encryption process between plain Image 1 and Image 2 (the acting Cover Image: Baboon). As shown in the table, both PSNR and SSIM yield small values, where PSNR yields values below 10 and SSIM with values closer to 0. In this case, the produced cipher images are all considered favoured and are rich in quality as it yields very small values which indicates that the cipher images are very different in terms of noise level and similarity compared to the original plain images. This is also indicating that the cipher images appear visually very different from the plain images, which can also visually be seen in Table 3 subsection 3.1.

4.1.1 Information Entropy Analysis

Image information entropy analysis (IE) is used to measure the amount of randomness and complexity of the cipher images. A higher entropy values indicate greater randomness in its pixel distribution. In this analysis, IE is used to evaluate the difference between the plain image and the cipher image. The difference of entropy values between before and after the image undergoes encryption process indicates the change in the shape and appearance between its original image and encrypted image. The computation of IE is as follow [22]:

$$\mathrm{IE} = -\sum_{i=0}^{255} p(m_i) \times \log_2 p(m_i) \tag{11}$$

where m_i is the pixel intensity and $p(m_i)$ is the probability occurrence of intensity m_i.

Table 5 shows the calculated entropy values of the original plain images and cipher images produced by each encryption processes. In this context, since each plain image underwent a combination process with one assigned second image, thus yielding different respective cipher image, the analysis will be performed to compare entropy of the original first image, Image 1, and its resulted cipher image. As seen in the table, the entropy value for cipher images averagely is around 7.99932, while the plain images have entropy values between 6.75761 and 7.82287, which indicates the difference in randomness between each plain image and its resulted cipher images.

4.1.2 Histogram Analysis

In general, a histogram of an image represents the distribution of each pixel intensity of an image. It shows the obvious statistical characteristics of the image. A difference

Table 5: Information Entropy analysis results.

Image 1	Original Image	Cipher Image
Airplane	6.75761	7.99939
Barbara	7.57708	7.99926
Boats	7.30307	7.99951
Goldhill	7.54204	7.99940
Lena	7.26762	7.99925
Sailboat	7.31945	7.99910
Pepper	7.35382	7.99922
Zelda	7.82287	7.99941
Average	**7.36795**	**7.99932**

in histogram shape between the original image and its after product, the cipher image, indicates there is a change performed by the encryption process. The histogram of a readable plain image usually has distinctive shape depending on its information. Whereas the histogram of a desirable cipher image appears to be evenly distributed or uniform in shape, indicating the pixels of all intensity are equally distributed. Table 6 shows the histogram of the first input plain image which is Image 1 and its resulted cipher image generated from the encryption process.

4.2 Analysis and Comparison

The cipher images produced by the proposed scheme overall result better quality in terms of its randomness, noise, and unrecognizable appearance. The reliable measurement to compare the randomness of the cipher images is by comparing the PSNR values, as it defines the noise level of the images. As what can be seen in Table 7, it shows the average values of PSNR for the cipher images generated from the proposed scheme and the cipher image generated from other existing selected schemes.

4.2.1 Differential Attacks Analysis

In this analysis, Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are used to evaluate the strength of the proposed scheme. NPCR is used to calculate the percentage of pixel change of the encrypted images, defining the changing rate of the number of pixels between the images with one-pixel difference. UACI is used to calculate the average difference in intensity of the pixels, comparing the pixels value between the two images [28]. An effective encryption algorithm will produce higher NPCR values, indicating that most of the images are greatly affected by the encryption, and low UACI values, indicating the visually noticeable change between the images. The formulas of NPCR and UACI are defined as follows:

$$\text{NPCR} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} D(i,j) \times 100\% \tag{12}$$

$$\text{UACI} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \tag{13}$$

where $M, N \in \mathbb{N}$ represents image dimensions where M denotes the row number and N denotes the column number. $D(i,j)$ represents the total number of changing pixels

Table 6: Histogram of plain Image 1 and its cipher image.

Image 1: Airplane

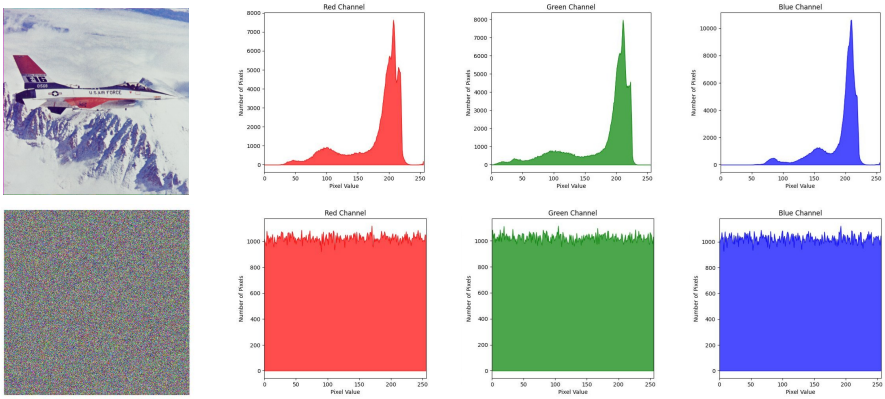



Image 1: Barbara

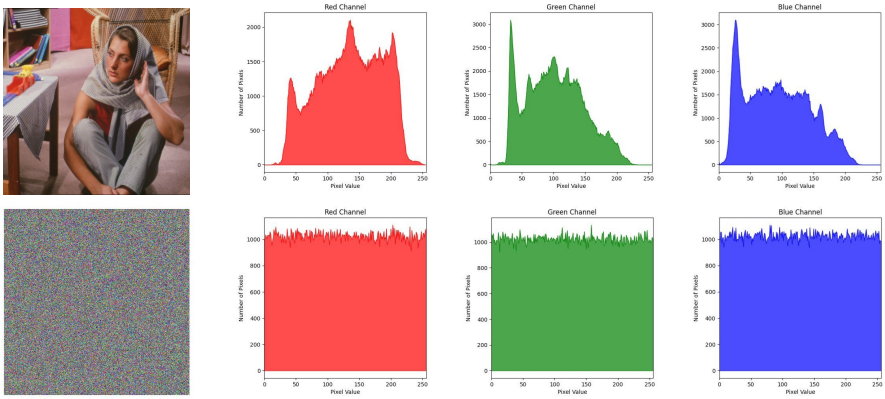



Image 1: Boats

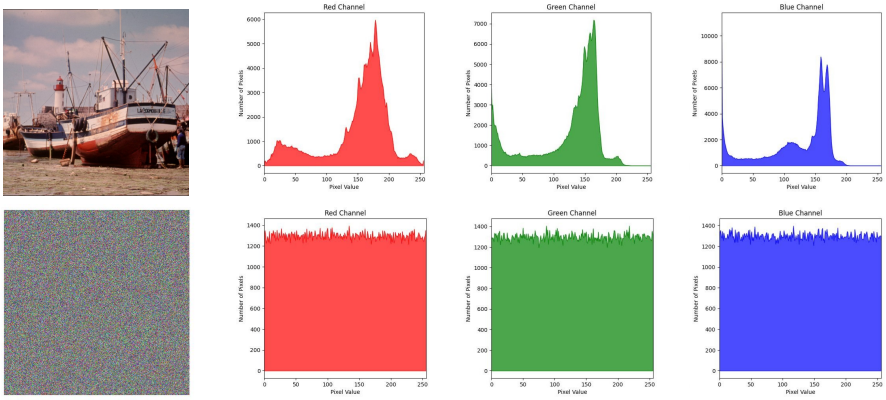



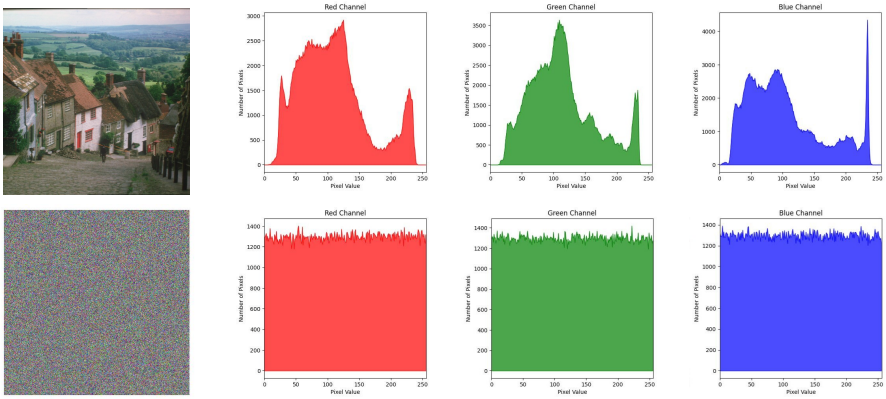
Image 1: Goldhill


Image 1: Lena

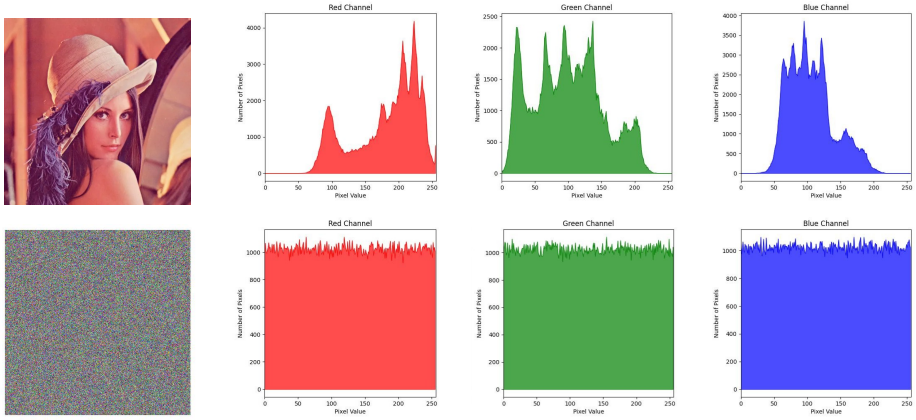


Image 1: Sailboat

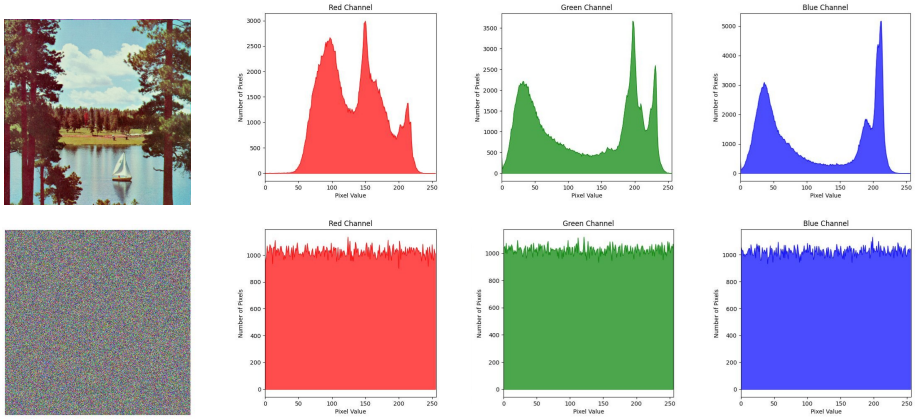


Image 1: Pepper

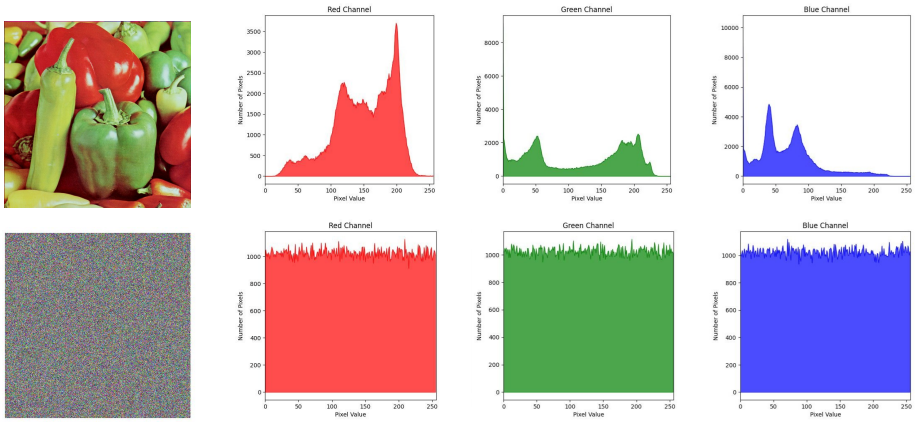


Image 1: Zelda

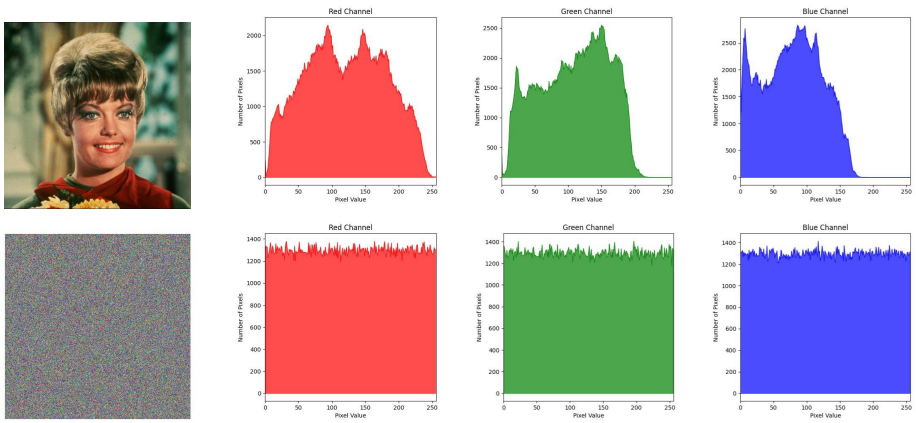


Table 7: PSNR average values comparison.

Encryption Techniques	Number of Images	Average PSNR Values
Proposed Scheme	8	7.67197
Method [23]	4	11.25908
Method [24]	12	7.664225
Method [25]	6	7.67470
Method [26]	4	8.93352
Method [27]	6	7.80833

Table 8: NPCR and UACI analysis result of proposed scheme.

Image 1	NPCR (%)	UACI (%)
Airplane	99.61497	33.46547
Barbara	99.61497	33.44105
Boats	99.61209	33.46095
Goldhill	99.61209	33.47140
Lena	99.61497	33.45279
Sailboat	99.61497	33.45908
Pepper	99.61497	33.44138
Zelda	99.61209	33.45376
Average	**99.61389**	**33.45574**

between the original image and cipher image, while $C_1(i,j)$ and $C_2(i,j)$ represent the intensity of the pixel in position (i,j) of the original image and cipher image respectively.

Table 8 below shows the NPCR and UACI percentage values achieved from the proposed scheme for all cipher images produced, while Table 9 shows the comparison between the NPCR and UACI average values of the proposed scheme and other existing Henon-based scheme. As seen in Table 8, the proposed scheme yields acceptable NPCR and UACI values approaching their respective ideal values, with NPCR values over 99.6% and UACI values within 33.4% to 33.5%. Averagely, in comparison to the other existing schemes that is also based on Henon chaotic map shown in Table 9, the proposed scheme can be seen to be just nicely on par with the others.

Table 9: NPCR and UACI comparison with different algorithms.

Encryption Techniques	Number of Images	Average NPCR (%)	Average UACI (%)
Proposed Scheme	8	99.61	33.46
Method [11]	13	99.61	33.40
Method [29]	6	99.62	33.47
Method [30]		99.62	33.34
Method [31]	1	99.61	33.46
Method [32]	4	99.62	33.46
Method [33]	5	99.60	33.45

Table 10: SSIM results between original Image 1 and its recovered image.

Image 1	SSIM
Airplane	1.0
Barbara	1.0
Boats	1.0
Goldhill	1.0
Lena	1.0
Sailboat	1.0
Pepper	1.0
Zelda	1.0

4.3 Image Reconstruction

In this analysis, to evaluate the performance of the scheme in restoring the plain images, SSIM measurement is used again as it measures and compare the similarities between the images. A good algorithm is reliable in recovering the plain images as similar as possible to the original images. Higher SSIM values at 1 or close to 1 indicate the greater similarity between the images in terms of its data structures as well as its visual appearance, thus indicating better image reconstruction. Table 10 shows the resulted SSIM values between the original plain images and its recovered plain images of the respective plain Image 1, which can be seen all to be in perfect SSIM values at 1.0.

4.4 Computational Efficiency Analysis

To evaluate the computational efficiency of the proposed scheme in terms of its run time and complexity, the execution for each encryption was measured on each image pair of the coloured images of size 512×512 and 576×576. The computation time of each image pair was tested over 10 independent cycles, and the average execution time of each process was recorded and compiled in Table 11. The table reports the run time of the four main parts of the scheme, from the combination process of the original images until the recovery of the plain images by the separation process, along with the total run time. As seen in the table, the average total run time the proposed scheme could offer is around 0.03645 seconds.

As seen in Table 12, in comparison between total execution time for images of different sizes of 512×512 and 576×576, the proposed scheme executes in 0.03275 seconds averagely for images with size 512×512 and 0.04261 seconds for images with size 576×576. Therefore, it can be said that different image size will execute in different run time, where image with smaller size took shorter and faster times. This scaling behaviours are different due to the quadratic relationship between image size and the number of pixel-level operations, with smaller images requiring lesser mathematical operations to process [11]. This result shows that the proposed scheme achieved a favourable execution time achieving nice balance between security and efficiency, which makes it reliable for real-time and practical applications.

Additionally, Table 13 shows the comparison of running time between the proposed system and other encryption algorithms. To make a fair comparison, the compared run time results were conducted only between encryption schemes with test images of the same size as the test image of the proposed scheme. Therefore, only the encryption results for the images of size 512×512 by the proposed scheme are included in the table, along with other encryption schemes with test images of the same size 512×512. And as seen in the table, the average run times of each execution by the proposed scheme are as short and

Table 11: Execution time by each process in the scheme (in seconds)

	Average Execution Time in 10 cycles (s)				
Image 1	Combination	Encryption	Decryption	Separation	Total
Airplane	0.00103	0.01370	0.01382	0.00206	0.03092
Barbara	0.00134	0.01501	0.01389	0.00198	0.03261
Boats	0.00155	0.01797	0.01772	0.00281	0.04125
Goldhill	0.00203	0.01890	0.01795	0.00286	0.04310
Lena	0.00169	0.01450	0.01478	0.00223	0.03425
Sailboat	0.00143	0.01511	0.01466	0.00231	0.03460
Pepper	0.00119	0.01397	0.01383	0.00201	0.03136
Zelda	0.00222	0.01962	0.01741	0.00277	0.04347
Average					**0.03645**

Table 12: Comparison of average execution time for images with different size

Image size	Number of images	Average total execution time (s)
512×512	5	0.03275
576×576	3	0.04261

fast as the other proposed scheme.

5 Conclusion

In summary, the proposed image encryption scheme has been successfully developed. The cryptosystem is markedly able to conduct both processes of combination and separation with the XOR function, and the process of encryption and decryption with the 4D Henon hyperchaotic map. The analysis on the image quality using PSNR and SSIM measurements, IE analysis, and histogram test shows that the proposed scheme is capable in producing a cipher image with excellent quality, leaving no visual resemblance to the original images. Whereas the analysis measured by NPCR and UACI proves that the proposed scheme is strong enough to withstand differential attacks. At the same time, the scheme also capable in restoring the plain image as exactly as the original image with perfect achieved SSIM values for similarity. Furthermore, the proposed scheme is computationally efficient with an average execution time under one second. This can be good for high-volume cipher images production and transactions. However, this study can be further extended toward the application of more hyper chaotic map with way more extreme randomness.

Table 13: Comparison of execution time with others existing schemes

Encryption Technique	Image Size	Average total execution time s
Proposed Scheme	512×512	0.03275
[11]	512×512	0.80369
[29]	512×512	0.92450
[34]	512×512	0.01930

Additionally, the implementation of encrypting double images in producing a single cipher image could be operated using other combination methods.

Acknowledgments

The authors gratefully acknowledge the financial support provided by the Centre for Postgraduate Studies, Universiti Malaysia Sabah (UMS). Sincere appreciation is also extended to the anonymous reviewers for their constructive comments and valuable recommendations, which have significantly improved the quality of this work.

References

[1] Zakaria, Wira Zanoramy A., Nur Mohammad Kamil Mohammad Alta, Mohd Faizal Abdollah, Othman Abdollah, and SM Warusia Mohamed SMM Yassin. Early Detection of Windows Cryptographic Ransomware Based on Pre Attack API Calls Features and Machine Learning. *Journal of Advanced Research in Applied Sciences and Engineering Technology* 39, No.2: 110-131, 2025.

[2] Chaoxia Zhang, Shangzhou Zhang, Kaiqi Liang, and Zhihao Chen. Double Image Encryption Algorithm Based on Parallel Compressed Sensing and Chaotic System. *IEEE Access* Vol. 12: 54745-54757, 2024.

[3] Aarthie, Narasimhan, and Rengarajan Amirtharajan. Image encryption: An information security perceptive. *Journal of Artificial Intelligence* 7, No.3: 123, 2014.

[4] Kaur, Mandeep, Surender Singh, and Manjit Kaur. Computational image encryption techniques: a comprehensive review. *Mathematical Problems in Engineering 2021*: 1-17, 2021.

[5] Manihira, Nitya Ranjan, and Alpesh Kumar Dauda. Image encryption using chaotic maps and DNA encoding. *International Journal of Engineering Research & Technology* 10, No.11: 1-5, 2022.

[6] Al-Maadeed, Somaya, Afnan Al-Ali, and Turki Abdalla. A new chaos-based image-encryption and compression algorithm. *A new chaos-based image-encryption and compression algorithm 2021*: 15-15, 2012.

[7] Shamsa Kawal, Saba Inam, Mohamed Tahar Ben Othman, Ayesha Waqar, Muhammad Ibrahim, Fariha Nawaz, Zainab Nawaz, and Habib Hamam. An Effective Color Image Encryption Based on Henon Map, Tent Chaotic Map, and Orthogonal Matrices. *Sensors 22*, No. 12: 10-12, 2025.

[8] Ramakrishnan Anandkumar and Ramanujam Kalpana. Designing a fast image encryption scheme using fractal function and 3D Henon Map. *Journal of Information Security and Applications* 49, 102390 (2019): 2214-2216, 2019.

[9] Ling Wang, Qiwen Ran, and Junrong Ding. Image Encryption Using Quantum 3D Mobius Scrambling and 3D Hyper-Chaotic Henon Map. *Advanced Technology in Quantum Cryptography* 25(12): 1629, 2023.

[10] Hendrick Richter. The Generalized Henon Maps: Examples for Higher-Dimensional Chaos. *International Journal of Bifurcation and Chaos* 12, No.06: 1371-1384, 2002.

[11] Yiting Lin, Yunlong Liao, Yunan Wei, Wenjun Zeng, Ugur Erkan, Abdurrahim Toktas, Yong Zhang, and Donglong Chen. Lightweight image encryption via four-dimensional Hénon memristor map and fast block permutation. *Nonlinear Science and Control Engineering 2025* 1(2), 025390012: 2025.

[12] Zhenlong Man, Jinqing Li, Xiaoqiang Di, Yaohui Sheng, and Zefei Liu. Double image encryption algorithm based on neural network and chaos. *Chaos, Solitons & Fractals* 152, 111318: 0960-0779, 2021.

[13] Guodong Ye, Min Liu, and Mingfa Wu. Double image encryption algorithm based on compressive sensing and elliptic curve. *Alexandria Engineering Journa* l, Vol 61, Issue 9: 6785-6795, 2022.

[14] Wenqian Yu, Ye Liu, Lihua Gong, Miaomiao Yian, and Liangqiang Tu. Double-image encryption based on spatiotemporal chaos and DNA operations. *Multimedia Tools and Applications* 78: 20037–20064, 2019.

[15] Wei Huang, Donghua Jiang, Yisheng An, Lidong Liu, and Xingyuang Wang. A Novel Double-Image Encryption Algorithm Based on Rossler Hyperchaotic System and Compressive Sensing. *IEEE Access* 9: 41704 – 41716, 2021.

[16] Sara T. Kamal, Khalid M. Hosny, Taha M. Elgindy, Mohamed M. Darwish, and Mostafa M. Fouda. A New Image Encryption Algorithm for Grey and Color Medical Images. *IEEE Access* Vol. 9: 37855-37865, 2021.

[17] Kurunandan Jain, Betrant Titus, Prabhakar Krishnan, Sujitha Sudevan, P. Prabu and Ala Saleh Alluhaidan. A Lightweight Multi-Chaos-Based Image Encryption Scheme for IoT Networks. *IEEE Access* Vol. 12: 62118-62148, 2024.

[18] Huangtao Wang and Qiang Lai. Exploiting Compressed Sensing and DNA Coding for Multi-image Encryption Scheme with Application to Industrial Internet of Things. I*IEEE Transactions on Industrial Electronics* Vol. 72, No. 12: 14543-14552, 2025.

[19] Faizal Nujumudeen, D. Muhammad Noorul Mubarak, and Tarak Hussain. Lightweight XOR-based visual cryptography using random shares for secure colour image sharing with minimal share. *Scientific Reports* 15, 42868: 2025.

[20] Laiphrakpam, Dolendro Singh, Rohit Thingbaijam, Khoirom Motilal Singh, and Moatsum Al Awida. Encrypting multiple images with an enhanced chaotic map. *IEEE Access*, 10: 87844-87859, 2022.

[21] Nilsson, Jim, and Tomas Akenine-Möller. Understanding SSIM. *arXiv preprint arXiv:2006*, 13846: 2020.

[22] Nawal El Ghouate, Mohamed Amine Tahiri, Ahmed Bencherqui, Hanaa Mansouri, Ahmed El Maloufy, Hicham Karmouni, Mhamed Sayyouri, Sameh Askar and Mohamed Abouhawwash. A high-entropy image encryption scheme using optimized chaotic maps with Josephus permutation strategy. *Sci Rep* 15, 29439: 2025.

[23] Siti Nurul Hatikah Mohammad and Arif Mandangan. Colour Image Encryption and Decryption using Arnold's Cat Map and Henon Map. *Colour Image Encryption and Decryption using Arnold's Cat Map and Henon Map* Vol.1, No.1: 41-52, 2024.

[24] Syahidatul Shafiqah Ramlee and Arif Mandangan. Chaotic Encryption Scheme for Double Grayscale Images using Sprott B Hyperchaotic Map. *International Journal of Advanced Research in Computational Thinking and Data Science* Vol.1, No.1: 25-40, 2024.

[25] De Rosal Ignatius Moses Setiadi, Abu Salam, Eko Hari Rachmawanto, and Christy Atika Sari. Improved color image encryption using modified modulus substitution cipher, dual random key and chaos method. *Computacion y Sistemas*, 25(3): 545-556, 2021.

[26] Jib Susanto, De Rosal Ignatius Moses Setiadi, Eko Hari Rachmawanto, Ibnu Utomo Wahyu Mulyono, Christy Atika Sari, Md Kamruzzaman Sarker, and Musfiqur Rahman Sazal. Triple layer image security using bit-shift, chaos, and stream encryption. *Bulletin of Electrical Engineering and Informatics* 9: 2020.

[27] Masood Fawad, Jawad Ahmad, Syed Aziz Shah, Sajjad Shaukat Jamal, and Iqtadar Hussain. A Novel Hybrid Secure Image Encryption Based on Julia Set of Fractals and 3D Lorenz Chaotic Map. *Entropy 22*, No.3: 274, 2020.

[28] Irene Lin Jin Ying and Arif Mandangan. Chaotic Encryption Scheme for Colour Image using 3D Lorenz Chaotic Map and 3D Chen System. *International Journal of Advanced Research in Computational Thinking and Data Science* Vol.1, No.1: 10-24, 2024.

[29] Qisu Gao and Xiaoqiang Zhang. Multiple-image encryption algorithm based on a new composite chaotic system and 3D coordinate matrix. *Chaos, Solitons & Fractals* 189(1), 115587: 0960-0779, 2024.

[30] Priyansi Parida, Chittaranjan Pradhan, Jafar A. Alzubi, Amir Javadpour, Mehdi Gheisari, Yang Liu, and Chen-Chi Lee. Elliptic curve cryptographic image encryption using Henon map and Hopfield chaotic neural network. *Multimedia Tools and Application* 82: 33637–33662, 2023.

[31] Shamsa Kanwal, Saba Inam, Mohamed Tahar Ben Othman, Ayesha Waqar, Muhammad Ibrahim, Fariha Nawaz, Zainab Nawaz, and Habib Hamam. An Effective Color Image Encryption Based on Henon Map, Tent Chaotic Map, and Orthogonal Matrices. *Sensors 2022*, No.12: 4359, 2022.

[32] Yong Chen, Shucui Xie, and Jianzhong Zhang. A Hybrid Domain Image Encryption Algorithm Based on Improved Henon Map. *Entropy* 24, No.2: 287, 2022.

[33] Yongsheng Hu, Han Wu, and Luoyu Zhou. Color image encryption base on a 2D hyperchaotic Henon map and cross diffusion. *Alezandria Engineering Journal* 73: 385-402, 2023.

[34] Saleh Ibrahim, and Ayman Alharbi. Efficient Image Encryption Scheme Using Henon Map, Dynamic S-Boxes and Elliptic Curve Cryptography. *IEEE Access* Vol. 8: 194289-194302, 2020.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 366–378.

An Inventory-Driven Risk Taxonomy for Post-Quantum Cryptography Migration

Isma Norshahila Binti Mohammad Shah[1,2], Faridatul Akhma Binti Ishak[1], Muhammad Haiman Bin Mohd Haizani[1], Hazlin Binti Abdul Rani[1,2], Nor Fatihah Binti Ahmad Majidi[1] and Nurul Aina Binti Rosli[1]

[1] Malaysia Cryptology Technology and Management Center, c/o Universiti Putra Malaysia 43400 UPM, Serdang, Selangor Darul Ehsan, Malaysia
{isma,faridatulakhma,haimanhaizani,hazlin,fatihahmajidi,ainarosli}@upm.edu.my
[2] CyberSecurity Malaysia, Level 7 Tower 1, Menara Cyber Axis, Jalan Impact, 63000 Cyberjaya, Selangor Darul Ehsan, Malaysia {hazlin,isma}@cybersecurity.my

Abstract. Post-Quantum Cryptography (PQC) migration requires organisations to understand where cryptographic mechanisms are used, how they are embedded within systems, and which dependencies may affect migration planning. Although Asset Inventory, Software Bill of Materials (SBOM) and Cryptographic Bill of Materials (CBOM) can provide important visibility into cryptographic use, the information collected from these sources does not automatically translate into structured migration-risk interpretation. Without a way to interpret inventory-derived evidence into migration-relevant categories, findings may remain fragmented and difficult to use in migration planning. This paper proposes an inventory-driven risk taxonomy for PQC migration to support early-stage interpretation of inventory-derived evidence. The taxonomy was developed using a classification-oriented approach informed by PQC migration guidance, cryptographic discovery practices, risk assessment concepts and taxonomy development principles, with candidate categories refined against observable inventory evidence. Five migration-oriented risk categories are proposed: Quantum-Vulnerable Algorithm Exposure, Cryptographic Protocol Exposure, Cryptographic Dependency Constraint, Crypto-Agility Constraint and Long-Term Confidentiality Exposure. The taxonomy provides a structured and traceable basis for mapping findings from Asset Inventory, SBOM and CBOM to one or more migration-risk categories without directly assessing organisational governance maturity. As a conceptual taxonomy, it has not yet been empirically validated against operational cryptographic inventory datasets. The proposed taxonomy therefore provides an interpretation layer for subsequent risk assessment, asset prioritisation, sector-specific refinement and phase-based PQC migration planning.

Keywords: Post-Quantum Cryptography · PQC Migration · Risk Taxonomy · Cryptographic Inventory · SBOM · CBOM · Crypto-Agility · Quantum Risk

1 Introduction

Post-Quantum Cryptography (PQC) migration has become an important cybersecurity priority as advances in quantum computing are expected to undermine the long-term security of widely used public-key cryptographic mechanisms such as RSA, Diffie–Hellman and elliptic curve cryptography. These mechanisms are embedded in many parts of today's digital infrastructure, including secure communication protocols, digital certificates, authentication services, software libraries and trusted digital platforms. With the release of the first National Institute of Standards and Technology (NIST) PQC standards, organisations

are now expected to prepare for a gradual transition from quantum-vulnerable cryptography to quantum-resistant mechanisms [14, 18, 15]. Quantum-related cryptographic risk may arise in different temporal forms. While direct compromise becomes possible once cryptographically relevant quantum capabilities emerge, harvest-now-decrypt-later attacks create a present-day confidentiality concern for encrypted information that must remain sensitive for an extended period [6, 11, 2].

However, PQC migration is not a straightforward replacement of one algorithm with another. Cryptographic mechanisms are often distributed across applications, operating systems, network services, software components, third-party libraries, certificates and vendor-managed platforms. Before migration decisions can be made, organisations need to understand where cryptography is used, which mechanisms may be exposed to quantum-related risks, and which systems may face technical constraints during migration. This makes cryptographic inventory, Software Bill of Materials (SBOM) and Cryptographic Bill of Materials (CBOM) important sources of evidence for early-stage PQC migration assessment [12, 20, 2].

The main challenge is that inventory information alone does not automatically provide structured risk interpretation. Asset Inventory, SBOM and CBOM may identify algorithms, protocols, libraries, certificates, dependencies and data categories, but these findings still need to be interpreted consistently. Without a clear taxonomy, inventory-derived findings may remain fragmented, difficult to compare across systems, and weakly connected to later migration planning.

Existing guidance already links cryptographic inventory with quantum-risk assessment and migration planning [2, 12, 3, 8]. However, these approaches do not explicitly provide a classification structure for organising individual inventory-derived findings into distinct, traceable and potentially overlapping PQC migration risk categories. The gap addressed in this study is therefore an intermediate classification layer between cryptographic discovery and subsequent risk assessment and migration planning. A systematic comparison of existing approaches is presented in Section 2 and Table 1.

This issue is also relevant to the Malaysian implementation context. National Cyber Security Agency (NACSA) Chief Executive Directive No. 9 requires National Critical Information Infrastructure (NCII) entities to prepare data and information for PQC migration, including initial cryptographic inventory, SBOM, CBOM, risk register, risk and dependency assessment, and an initial phased migration plan. The directive also aims to identify cryptographic inventory readiness against quantum computing threats so that PQC migration can be implemented according to priority [1]. This reinforces the need for a practical method to translate inventory-derived information into meaningful PQC migration risk categories.

This paper addresses the following research question: How can inventory-derived findings from Asset Inventory, SBOM and CBOM be classified into structured PQC migration risk categories that support consistent and traceable early-stage interpretation?

To address this question, this paper develops an inventory-driven risk taxonomy for PQC migration. The taxonomy focuses on cryptographic exposure, technical dependency, migration constraint and long-term confidentiality concerns that can be derived from inventory-based evidence. The main contributions of this paper are as follows:

1. It proposes a focused PQC migration risk taxonomy based on evidence obtained from Asset Inventory, SBOM and CBOM.

2. It provides a structured and traceable basis for mapping individual inventory-derived findings to one or more migration-oriented risk categories.

3. It positions the taxonomy as an interpretation layer between cryptographic discovery and subsequent activities such as detailed risk assessment, asset prioritisation and phase-based PQC migration planning.

2 Related Work

This section reviews prior work and guidance relevant to an inventory-driven PQC migration risk taxonomy, focusing on operational migration, cryptographic visibility, migration risk, crypto-agility and long-term confidentiality.

2.1 PQC Migration as an Operational Transition

PQC migration is increasingly recognised as an operational transition rather than a purely cryptographic standardisation activity [4]. NIST's PQC standards and transition guidance provide the technical basis for replacing quantum-vulnerable public-key mechanisms [14, 18, 15, 10]. However, implementation requires organisations to first identify where vulnerable cryptography is deployed. NIST SP 1800-38B and ETSI TR 103 619 therefore emphasise cryptographic discovery, inventory compilation and migration planning [12, 7].

The *PQC Migration Handbook* extends this operational view by linking inventories of cryptographic assets, data and suppliers with system- or application-level assessment of quantum weakness, expected impact and migration effort [2]. Nevertheless, these approaches do not explicitly classify individual inventory-derived findings into distinct PQC migration risk categories. This leaves an intermediate interpretation gap between cryptographic discovery and subsequent risk assessment.

2.2 From Cryptographic Visibility to Risk Interpretation

Asset Inventory, SBOM and CBOM provide complementary visibility into cryptographic use. Asset Inventory provides system and operational context, SBOM identifies software components and dependencies, while CBOM captures cryptographic algorithms, protocols, libraries and related cryptographic information [12, 16, 19]. The *PQC Migration Handbook* similarly treats cryptographic, data and supplier inventories as inputs to quantum-risk assessment [2].

Visibility alone, however, does not identify the type of migration concern represented by each finding. For example, RSA may indicate algorithm exposure, its use within TLS may additionally indicate protocol exposure, and reliance on a vendor-managed implementation may indicate dependency constraint. Existing approaches can use such information in system-level risk assessment, but do not explicitly classify the individual findings into distinct and potentially overlapping migration-risk categories.

This need is also reflected in the Malaysian context. NACSA Chief Executive Directive No. 9 requires cryptographic inventory, SBOM, CBOM, risk and dependency assessment, and an initial phased migration plan [1]. A taxonomy can therefore provide an interpretation layer between collected inventory evidence and subsequent migration assessment and planning.

2.3 Quantum Migration Risk Beyond Algorithms

PQC migration risk cannot be reduced to algorithm weakness alone. Baseri et al. [3] examine quantum-security risk across algorithmic, certificate and protocol layers, reflecting the fact that cryptographic algorithms are deployed through protocols, certificates, software stacks and operational services [5]. The *PQC Migration Handbook* similarly considers application-level quantum weakness together with supplier, hardware and implementation dependencies when estimating migration effort [2].

The proposed taxonomy adapts these broader risk perspectives to inventory-derived findings. Algorithm-related findings are classified as *Quantum-Vulnerable Algorithm Exposure*, while protocol-related findings are classified as *Cryptographic Protocol Exposure*. Certificate-, key- and implementation-related findings are considered where they create

Table 1: Coverage of Existing Frameworks and Guidance Relative to the Proposed Taxonomy

Framework / Guidance	PQC Migration Guidance	Cryptographic Discovery / Inventory	Structured Risk Assessment / Classification	SBOM / CBOM Evidence Basis	Explicit Finding-Level Inventory-to-Risk Classification
[12]	Yes	Yes	No	Yes	No
[7]	Yes	Yes	No	Partial	No
[3]	Yes	Partial	Yes	No	No
[8]	Partial	No	Yes	No	No
[6]	Yes	Partial	No	No	No
[2]	Yes	Yes	Yes	Yes	No
This paper	Partial	Partial	Yes	Yes	Yes

protocol exposure, dependency or migration constraint. The distinction is that the unit of analysis is the inventory finding rather than the system-level risk outcome.

2.4 Crypto-Agility Constraint in PQC Migration

Crypto-agility is important because systems must be able to replace or reconfigure cryptographic mechanisms as standards and technologies evolve. CARAF shows that limited crypto-agility can make cryptographic transitions slower and more complex [8], while the *PQC Migration Handbook* similarly emphasises the ability to modify or replace deployed cryptography with minimal disruption [2].

In this taxonomy, *Crypto-Agility Constraint* concerns whether cryptographic mechanisms can be changed, configured or upgraded without significant redevelopment. It is distinguished from *Cryptographic Dependency Constraint*, which concerns whether the required change depends on an external component, supplier or platform. The two may coexist but represent different migration concerns. Existing work establishes the importance of crypto-agility but does not explicitly classify inventory-derived indicators into this finding-level category.

2.5 Long-Term Confidentiality and Harvest Now, Decrypt Later Risk

Long-term confidentiality is a key driver of PQC migration because encrypted data collected today may be decrypted once sufficiently capable quantum computers become available [6]. This creates a different temporal exposure from direct quantum compromise: the cryptographic break occurs in the future, but the confidentiality risk may begin when protected information is intercepted and retained today [6, 2]. Mosca [11] relates this risk to data security lifetime, migration time and the expected quantum-threat timeline. Systems protecting long-lived sensitive or classified information may therefore require greater migration urgency than systems protecting short-lived data.

The *PQC Migration Handbook* similarly incorporates confidentiality lifetime into system- or application-level impact and migration urgency [2]. In the proposed taxonomy, *Long-Term Confidentiality Exposure* is retained as an explicit, inventory-traceable category so that the relationship between protected data, its confidentiality lifetime and the associated cryptographic mechanism remains visible before severity or migration priority is assigned.

2.6 Research Gap

Existing frameworks and guidance address complementary parts of PQC migration, including cryptographic discovery, risk assessment, crypto-agility and migration planning. Table 1 summarises their coverage relative to the proposed taxonomy. “Yes” indicates direct coverage, “Partial” indicates limited or indirect coverage, and “No” indicates that the aspect is not explicitly addressed.

The comparison shows that NIST and ETSI primarily support cryptographic discovery and migration planning, while Baseri et al. and CARAF provide structured risk perspectives at the cryptographic-layer or crypto-agility level [12, 7, 3, 8]. The *PQC Migration Handbook* provides the closest inventory-to-risk relationship by using cryptographic, data and supplier inventories in system- or application-level quantum-risk assessment [2].

The remaining gap is therefore not the absence of inventory-based PQC risk assessment, but the absence of an explicit interpretation layer that classifies individual Asset Inventory, SBOM and CBOM findings into consistent, traceable and potentially overlapping migration-risk categories. The proposed taxonomy addresses this intermediate role before subsequent severity assessment, asset prioritisation and phase-based migration planning.

3 Taxonomy Development Methodology

This section describes the approach used to develop the proposed PQC migration risk taxonomy. The aim is to define a practical set of risk categories that can support early interpretation of evidence from Asset Inventory, SBOM and CBOM. The taxonomy is intended to classify inventory-derived findings into migration-related concerns, rather than to determine severity, assign priority or prescribe mitigation actions.

3.1 Methodological Basis

The taxonomy was developed using a classification-oriented approach informed by the taxonomy development method of Nickerson et al. [17]. Taxonomy development is used in information systems research to organise objects within a domain according to shared characteristics and distinguishing features. In this study, the method was adapted as a conceptual classification process in which candidate characteristics were derived from relevant literature and guidance and iteratively refined against the types of evidence available from cryptographic inventory artefacts.

The objects being classified are not organisations or systems as a whole, but inventory-derived findings that may indicate PQC migration-relevant concerns. The risk framing was informed by established cybersecurity risk assessment guidance, where risk interpretation should be supported by identifiable factors and consistent terminology [13]. Threat and risk taxonomies are also useful for structuring security information into consistent categories when evidence is obtained from multiple sources [9]. In the PQC context, such evidence may originate from Asset Inventory, SBOM, CBOM, system documentation, protocol configuration, certificate information and related technical validation.

The Malaysian implementation context was considered through NACSA Chief Executive Directive No. 9, which requires the preparation of PQC migration-related data, including initial cryptographic inventory, SBOM, CBOM, risk register, risk and dependency assessment, and an initial phased migration plan [1]. This provides a practical basis for considering how inventory-derived evidence can be interpreted into migration-related risk categories.

3.2 Taxonomy Derivation Process

The taxonomy was derived through three iterative steps. The review was purposive rather than a systematic literature review and focused on sources directly relevant to PQC migration, cryptographic discovery, quantum-security risk, crypto-agility, long-term confidentiality, risk assessment and taxonomy development. These included authoritative guidance and standards as well as peer-reviewed frameworks discussed in Section 2.

First, migration-relevant themes were extracted from the selected sources by identifying recurring technical conditions or exposures that could affect PQC migration. Initial themes

included quantum-vulnerable algorithms, protocol and certificate exposure, cryptographic libraries, third-party and vendor-controlled components, key-related dependencies, crypto-agility, migration constraints, data sensitivity and long-term confidentiality.

Second, each theme was checked against the types of evidence that could reasonably be obtained from Asset Inventory, SBOM and CBOM. A theme was retained only when it could be associated with observable inventory evidence or related technical information. Themes that primarily represented organisational governance, policy maturity or broader risk-management practices were not retained as standalone categories.

Third, the retained themes were consolidated according to the migration concern they represented. Themes were combined when they described the same underlying migration condition, and separated when they represented technically different concerns. For example, vendor-controlled libraries, HSMs and third-party modules were grouped under dependency-related constraints, whereas hard-coded algorithms, limited configurability and unsupported cryptographic implementations were grouped under crypto-agility constraints. Certificate- and key-related findings were not retained as independent categories because their migration relevance depends on whether they create algorithm exposure, protocol exposure, technical dependency or migration constraint.

A finding was allowed to map to more than one category where the same evidence represented different migration concerns. This non-exclusive treatment preserves the distinction between, for example, a quantum-vulnerable algorithm, the protocol in which it is used and the dependency that may constrain its replacement.

The refinement process was concluded when the reviewed themes could be represented without introducing additional migration-relevant categories, each retained category could be traced to observable evidence, and the category boundaries satisfied the design criteria described in Section 3.3. This process resulted in five proposed categories: Quantum-Vulnerable Algorithm Exposure, Cryptographic Protocol Exposure, Cryptographic Dependency Constraint, Crypto-Agility Constraint and Long-Term Confidentiality Exposure.

3.3 Design Criteria

The final categories were checked against four design criteria:

1. **Migration relevance:** the category must represent a concern that can affect PQC migration.
2. **Evidence traceability:** the category must be supported by observable evidence from Asset Inventory, SBOM, CBOM or related technical sources.
3. **Distinctiveness:** the category must represent a migration concern that is analytically distinguishable from the other categories, while allowing a finding to map to multiple categories when different concerns coexist.
4. **Operational neutrality:** the category must describe technical exposure or migration constraint without treating the finding as an organisational governance weakness.

These criteria were applied after each refinement cycle and as a final consistency check. Table 2 summarises the methodological steps and their outputs.

This methodology provides the basis for the proposed taxonomy presented in the next section.

3.4 Scope and Limitations of the Methodology

The taxonomy was developed as a conceptual classification study based on literature, guidance and analysis of the types of evidence available from cryptographic inventory

Table 2: Summary of taxonomy development methodology

Methodological Step	Operational Rule	Main Output
Source and theme review	Identify recurring PQC migration concerns from relevant literature and guidance	Candidate risk themes
Evidence traceability check	Retain themes that can be linked to observable inventory or technical evidence	Inventory-traceable themes
Category refinement	Consolidate related themes and resolve overlapping or governance-oriented concepts	Migration-oriented categories
Design criteria check	Check migration relevance, traceability, distinctiveness and operational neutrality	Final PQC migration risk taxonomy

artefacts. It has not yet been validated through expert-panel review, inter-rater assessment or empirical testing against operational cryptographic inventory datasets. The category definitions and indicators therefore represent the authors' structured interpretation of inventory-derived evidence based on the selected sources. Expert review, inter-rater assessment and application to real or simulated inventory datasets are identified as subsequent validation activities.

4 Proposed Inventory-Driven PQC Migration Risk Taxonomy

This section presents the proposed inventory-driven risk taxonomy for PQC migration. The taxonomy consists of five categories derived from the literature, guidance and inventory evidence considerations described in Section 3. Each category represents a migration concern that can be interpreted from Asset Inventory, SBOM, CBOM or related technical evidence.

The taxonomy does not assign scores or determine migration priority. Instead, it provides a classification structure to interpret inventory-derived findings before subsequent activities such as risk assessment, asset prioritisation or phase-based PQC migration planning.

The five proposed categories are:

1. Quantum-Vulnerable Algorithm Exposure
2. Cryptographic Protocol Exposure
3. Cryptographic Dependency Constraint
4. Crypto-Agility Constraint
5. Long-Term Confidentiality Exposure

These categories focus on cryptographic exposure, technical dependency, migration constraint and confidentiality lifetime rather than broad organisational governance weaknesses.

4.1 Overview of the Proposed Taxonomy

PQC migration concerns may arise from more than one characteristic of the same finding. A quantum-vulnerable algorithm may be embedded within a protocol, controlled by an external dependency, difficult to replace, or used to protect long-lived confidential data. The taxonomy therefore separates these concerns into distinct but non-mutually exclusive categories. Table 3 summarises the category definitions, example indicators and principal evidence sources.

Table 3: Proposed PQC migration risk taxonomy

Risk Category	Definition	Example Indicators	Main Evidence Source
Quantum-Vulnerable Algorithm Exposure	Exposure arising from the use of public-key mechanisms vulnerable to quantum attacks.	RSA, Diffie–Hellman, ECDH, ECDSA, DSA, EdDSA.	CBOM, cryptographic configuration, certificates.
Cryptographic Protocol Exposure	Exposure where a protocol or secure channel relies on quantum-vulnerable cryptographic mechanisms.	TLS, SSH, VPN, IPsec, S/MIME or DNSSEC using classical public-key mechanisms.	CBOM, protocol scan, network configuration, documentation.
Cryptographic Dependency Constraint	Constraint where migration depends on an externally controlled component, technology or service.	Vendor-managed library, HSM, third-party module, SDK, managed platform, CA service.	SBOM, CBOM, vendor documentation, architecture.
Crypto-Agility Constraint	Constraint where cryptographic mechanisms cannot be readily changed, configured or replaced.	Hard-coded algorithm, limited configuration, unsupported library, redevelopment required.	CBOM, SBOM, architecture review, technical validation.
Long-Term Confidentiality Exposure	Exposure where protected data must remain confidential long enough to be affected by future quantum capabilities.	Sensitive, regulated or classified data with long confidentiality or retention requirements.	Asset Inventory, data classification, retention requirements, CBOM.

For operational classification, the following heuristics are applied. A finding is classified as *Quantum-Vulnerable Algorithm Exposure* when the evidence identifies a quantum-vulnerable public-key mechanism. If that mechanism is instantiated within a secure communication protocol or channel, *Cryptographic Protocol Exposure* is additionally assigned. *Cryptographic Dependency Constraint* is assigned when migration depends on a component, supplier, platform or service that cannot be independently changed by the system owner. *Crypto-Agility Constraint* is assigned when the cryptographic mechanism itself cannot be readily reconfigured, upgraded or replaced without significant technical modification. *Long-Term Confidentiality Exposure* is assigned when the protected data has a confidentiality lifetime that may extend into the period of quantum-related exposure.

These rules are non-exclusive. A finding that satisfies more than one condition is assigned to each applicable category, with each assignment linked to its supporting evidence. Multiple assignments indicate different migration concerns associated with the same finding; they do not imply greater severity or determine migration priority. For example, an RSA-based certificate used in a TLS service may indicate both Quantum-Vulnerable Algorithm Exposure and Cryptographic Protocol Exposure. If the implementation is controlled by a vendor-managed platform, Cryptographic Dependency Constraint may also apply.

4.2 Quantum-Vulnerable Algorithm Exposure

Quantum-Vulnerable Algorithm Exposure refers to the use of classical public-key cryptographic mechanisms that may be compromised by cryptographically relevant quantum computers. Typical findings include RSA, Diffie–Hellman, ECDH, ECDSA, DSA and related mechanisms. This category reflects the transition from quantum-vulnerable public-key cryptography described in NIST PQC standards and transition guidance [14, 18, 15, 10], as well as the algorithmic risk layer discussed by Baseri et al. [3].

The category identifies algorithm-level exposure only; it does not by itself determine severity or migration priority.

4.3 Cryptographic Protocol Exposure

Cryptographic Protocol Exposure applies when a secure protocol or communication channel relies on quantum-vulnerable cryptographic mechanisms. Examples include TLS using RSA or ECDSA certificates, VPN or IPsec using Diffie–Hellman or ECDH, and SSH or secure messaging mechanisms relying on classical public-key cryptography.

Protocol exposure is treated separately from algorithm exposure because migration may also involve interoperability, client compatibility, certificate handling and protocol configuration. Baseri et al. identify protocol-level risk as distinct from algorithmic risk [3],

while NIST SP 1800-38B emphasises discovery of cryptographic usage and dependencies in deployed environments [12].

4.4 Cryptographic Dependency Constraint

Cryptographic Dependency Constraint applies when migration depends on a component, technology or service that is not independently controlled by the system owner. Examples include vendor-managed libraries, third-party cryptographic modules, HSMs, SDKs, managed platforms and external certificate services.

Such dependencies may delay or restrict migration even after quantum-vulnerable cryptography has been identified. PQC readiness guidance recognises supplier, hardware and implementation dependencies as important adoption challenges [20, 2]. Certificate- and key-related findings are included in this category only when they create a technical dependency rather than a broader governance concern.

4.5 Crypto-Agility Constraint

Crypto-Agility Constraint applies when cryptographic mechanisms cannot be readily changed, configured, upgraded or replaced. Examples include hard-coded algorithms, non-configurable implementations, unsupported cryptographic libraries or architectures that require substantial redevelopment.

This category is distinct from Cryptographic Dependency Constraint. Dependency concerns *what external component or party controls the required change*, whereas crypto-agility concerns *whether the implementation itself can accommodate the change*. Both may coexist in the same system. CARAF demonstrates that limited crypto-agility can make cryptographic transition slow and complex [8], while the *PQC Migration Handbook* similarly emphasises cryptographic agility as an important capability for future cryptographic transitions [2].

4.6 Long-Term Confidentiality Exposure

Long-Term Confidentiality Exposure is assigned when the protected data has a confidentiality lifetime that may extend into the period of quantum-related exposure and its confidentiality depends on a quantum-vulnerable public-key mechanism. This is particularly relevant to harvest-now-decrypt-later scenarios, where encrypted information collected today may be decrypted once quantum-vulnerable public-key mechanisms can be compromised [6, 11]. Accordingly, this category captures a present-day confidentiality exposure even though the cryptographic compromise itself may only become technically feasible in the future.

The category distinguishes data-lifetime exposure from current cryptographic weakness. Sensitive records, regulated information, classified data or long-retention archives may therefore create a migration concern even before a cryptographically relevant quantum computer exists. The *PQC Migration Handbook* similarly incorporates data confidentiality lifetime into quantum-risk and migration-urgency considerations [2].

In summary, the five categories distinguish algorithm-level exposure, protocol-level exposure, external dependency, internal changeability and confidentiality lifetime. Together, they provide a traceable classification layer for interpreting inventory-derived findings before subsequent risk assessment and migration prioritisation.

5 Mapping Inventory Evidence to Risk Categories

The proposed taxonomy provides an interpretation layer between cryptographic inventory and subsequent risk assessment. Asset Inventory, SBOM and CBOM provide different

Table 4: Mapping inventory evidence to PQC migration risk categories

Evidence Source	Example Evidence / Indicator	Related Risk Category
Asset Inventory	Sensitive, regulated or classified data with long confidentiality requirements	Long-Term Confidentiality Exposure
SBOM / CBOM	Vendor-controlled library, module, HSM, SDK, managed platform or CA service	Cryptographic Dependency Constraint
SBOM	Unsupported or end-of-life component	Cryptographic Dependency Constraint and/or Crypto-Agility Constraint
CBOM	RSA, Diffie–Hellman, ECDH, ECDSA or DSA	Quantum-Vulnerable Algorithm Exposure
CBOM / Protocol Scan	Quantum-vulnerable mechanism used within TLS, SSH, VPN, IPsec, S/MIME or DNSSEC	Quantum-Vulnerable Algorithm Exposure; Cryptographic Protocol Exposure
CBOM / Architecture Review	Hard-coded or non-configurable cryptographic mechanism	Crypto-Agility Constraint

forms of evidence and should therefore not be treated as equivalent sources. Classification is based on what a finding indicates about migration, rather than solely on the inventory artefact from which it originates.

The mapping is therefore evidence-conditioned rather than source-conditioned. For example, the presence of a third-party component in an SBOM does not automatically constitute a dependency constraint; the category applies when the required cryptographic change cannot be performed independently by the system owner. Similarly, an unsupported component may indicate a dependency constraint when remediation depends on an external supplier, a crypto-agility constraint when the implementation cannot accommodate cryptographic change, or both when these conditions coexist.

CBOM evidence provides a more direct link to algorithm and protocol exposure. A quantum-vulnerable algorithm is classified as algorithm exposure, while protocol exposure is additionally assigned only when contextual evidence shows that the mechanism is instantiated within a secure protocol or communication channel. Long-term confidentiality exposure similarly requires the relationship between protected data, its confidentiality lifetime and the cryptographic mechanism protecting it to be established.

These associations have different implications for subsequent migration activities. Algorithm and protocol exposure identify where cryptographic transition and interoperability must be assessed; dependency constraints highlight the need for supplier or platform coordination; crypto-agility constraints identify implementations that may require configuration changes or redevelopment; and long-term confidentiality exposure identifies data-lifetime considerations that should inform subsequent urgency and prioritisation assessment. The taxonomy itself does not assign severity or migration priority.

6 Discussion and Future Work

The proposed taxonomy provides a structured way to interpret findings from Asset Inventory, SBOM and CBOM from a PQC migration perspective. Its main role is to organise cryptographic visibility into meaningful migration-oriented risk categories. By grouping observable indicators into five categories, the taxonomy helps distinguish between different types of migration concern rather than treating all cryptographic findings in the same way.

A key strength of the taxonomy is that it separates algorithm exposure, protocol exposure, dependency constraint, crypto-agility constraint and long-term confidentiality exposure. These concerns may appear together in the same system, but they represent different migration meanings. This distinction can support clearer communication among system owners, security teams, technical teams and decision makers during early PQC migration planning.

The taxonomy is also intentionally neutral in tone. It does not frame findings as organisational failures or governance weaknesses. Instead, it focuses on migration exposure,

dependency and constraints that can be observed from inventory-derived evidence. This makes it suitable for early-stage assessment contexts where consistency, sensitivity and explainability are important.

However, the taxonomy has several limitations. It does not assign severity, determine migration priority or replace a detailed risk assessment method. Its effectiveness also depends on the quality and completeness of the underlying inventory evidence. In addition, the taxonomy has not yet been empirically evaluated for completeness, classification consistency or reproducibility across different assessors and operational environments.

Future validation should proceed in stages. First, the category definitions and classification heuristics should be reviewed by cryptography and cybersecurity practitioners to assess their clarity and practical relevance. Second, the taxonomy should be applied to real or simulated cryptographic inventory datasets to evaluate whether inventory-derived findings can be classified consistently. Inter-rater assessment can then be used to measure classification consistency and identify ambiguous category boundaries.

Subsequent work may refine the taxonomy for sector-specific environments and link the classified findings to severity assessment, asset prioritisation and phase-based PQC migration planning. The classification rules may also be implemented in automated or semi-automated analysis tools for structured Asset Inventory, SBOM and CBOM data. Such automation should preserve traceability between each assigned category and the underlying inventory evidence so that classification outcomes remain explainable and subject to technical validation.

7 Conclusion

This paper proposed an inventory-driven risk taxonomy for Post-Quantum Cryptography migration. The taxonomy addresses the need for structured interpretation of cryptographic visibility by classifying findings from Asset Inventory, SBOM and CBOM into five migration-oriented categories: Quantum-Vulnerable Algorithm Exposure, Cryptographic Protocol Exposure, Cryptographic Dependency Constraint, Crypto-Agility Constraint and Long-Term Confidentiality Exposure.

The proposed taxonomy provides a structured and traceable interpretation layer between cryptographic discovery and subsequent risk assessment and migration planning. Its non-exclusive classification allows individual findings to represent multiple migration concerns without directly assigning severity, migration priority or organisational governance maturity. Further empirical validation is required to assess classification consistency, refine category boundaries and support future integration with asset prioritisation, sector-specific assessment and phase-based PQC migration planning.

References

[1] Agensi Keselamatan Siber Negara. Arahan Ketua Eksekutif No. 9: Penyediaan Data dan Maklumat Pelaksanaan Migrasi Kriptografi Pasca Kuantum oleh Entiti Infrastruktur Maklumat Kritikal Negara. https://www.nacsa.gov.my/legal.php, September 2025. Accessed: 2025-09-23.

[2] A. Amadori, T. Attema, M. Bombar, J. D. Duarte, V. Dunning, S. Etinski, D. van Gent, M. Lequesne, W. van der Schoot, M. Stevens, and AIVD Cryptologists and Advisors. The PQC Migration Handbook: Guidelines for Migrating to Post-Quantum Cryptography. Technical report, AIVD, CWI, and TNO, December 2024. Accessed: 2026-05-28.

[3] Yaser Baseri, Vikas Chouhan, Ali Ghorbani, and Aaron Chow. Evaluation framework for quantum security risk assessment: A comprehensive strategy for quantum-safe transition. *Computers & Security*, 150:104272, 2025.

[4] Berat Birgin and Baris Celiktas. From policy to practice: A sector-agnostic operational framework for post-quantum cryptography transition. *IEEE Access*, 2026.

[5] Matt Campagna, Brian LaMacchia, and David Ott. Post quantum cryptography: readiness challenges and the approaching storm. *arXiv preprint arXiv:2101.01269*, 2021.

[6] Cybersecurity and Infrastructure Security Agency, National Security Agency, and National Institute of Standards and Technology. Quantum-Readiness: Migration to Post-Quantum Cryptography. Technical report, Cybersecurity and Infrastructure Security Agency, August 2023.

[7] ETSI. Migration Strategies and Recommendations to Quantum Safe Schemes. ETSI Technical Report ETSI TR 103 619, European Telecommunications Standards Institute, July 2020.

[8] Chujiao Ma, Luis Colon, Joe Dera, Bahman Rashidi, and Vaibhav Garg. Caraf: Crypto agility risk assessment framework. *Journal of Cybersecurity*, 7(1):tyab013, 2021.

[9] Louis Marinos. ENISA Threat Taxonomy: A Tool for Structuring Threat Information. Initial Version, European Union Agency for Network and Information Security (ENISA), January 2016.

[10] Dustin Moody, Ray Perlner, Andrew Regenscheid, Angela Robinson, and David Cooper. Transition to post-quantum cryptography standards. Technical report, National Institute of Standards and Technology, 2024.

[11] Michele Mosca. Cybersecurity in an era with quantum computers: Will we be ready? *IEEE Security & Privacy*, 16(5):38–41, 2018.

[12] National Cybersecurity Center of Excellence. Migration to Post-Quantum Cryptography: Quantum Readiness: Cryptographic Discovery. NIST Special Publication 1800-38B, National Institute of Standards and Technology, December 2023. Preliminary Draft.

[13] National Institute of Standards and Technology. Guide for Conducting Risk Assessments. NIST Special Publication 800-30 Revision 1, National Institute of Standards and Technology, 2012.

[14] National Institute of Standards and Technology. FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard. Technical Report FIPS 203, National Institute of Standards and Technology, August 2024.

[15] National Institute of Standards and Technology. Stateless Hash-Based Digital Signature Standard. Federal Information Processing Standards Publication FIPS 205, National Institute of Standards and Technology, August 2024.

[16] National Telecommunications and Information Administration. The Minimum Elements for a Software Bill of Materials (SBOM). Technical report, National Telecommunications and Information Administration, July 2021.

[17] Robert C Nickerson, Upkar Varshney, and Jan Muntermann. A method for taxonomy development and its application in information systems. *European journal of information systems*, 22(3):336–359, 2013.

[18] National Institute of Standards and Technology. Fips 204, module-lattice-based digital signature standard. 2024.

[19] M. Osborne. Cryptographic Bill of Materials. IBM Research Blog, August 2022. Accessed: 2026-05-28.

[20] Murugiah Souppaya, Tim Polk, and William Barker. Getting ready for post-quantum cryptography: Exploring challenges associated with adopting and using post-quantum cryptographic algorithms. 2021.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 379–393.

Governance-Driven PQC Migration for National Critical Information Infrastructure: Operationalisation Challenges and Early Readiness Insights from Malaysia

Hazlin Binti Abdul Rani[1,3], Isma Norshahila Binti Mohammad Shah[1,3], Azlina Binti Ab Aziz[2] and Afiq Muzakkir Bin Mat Ali[2]

[1] Malaysia Cryptology Technology and Management Center, c/o Universiti Putra Malaysia 43400 UPM, Serdang, Selangor Darul Ehsan, Malaysia {hazlin,isma}@upm.edu.my

[2] Agensi Keselamatan Siber Negara (NACSA), Majlis Keselamatan Negara (MKN), Jabatan Perdana Menteri (JPM), Blok Barat, Bangunan Perdana Putra, Pusat Pentadbiran Kerajaan Persekutuan, 62502 Putrajaya, Malaysia {azlina.abaziz,afiq.ali}@nacsa.gov.my

[3] CyberSecurity Malaysia, Level 7 Tower 1, Menara Cyber Axis, Jalan Impact, 63000 Cyberjaya, Selangor Darul Ehsan, Malaysia {hazlin,isma}@cybersecurity.my

Abstract. Post-Quantum Cryptography (PQC) migration is increasingly recognised as a critical national priority for strengthening cybersecurity, preserving digital trust, and enhancing cyber resilience, especially within National Critical Information Infrastructure (NCII) sectors where the security of public-key cryptographic systems is fundamental to protecting critical services and sensitive information. While many existing discussions focus on algorithm selection and technical implementation, national-scale migration also requires governance coordination, cryptographic visibility, operational readiness, and sustained institutional support. This paper outlines Malaysia's strategic governance framework for driving PQC migration across NCII entities. Using a descriptive-analytical case study method, the paper examines how Malaysia's national cybersecurity and cryptographic governance ecosystem supports early migration readiness through policy direction, institutional coordination, assurance mechanisms, and structured operational tools. The study highlights the role of cryptographic inventories, Software Bill of Materials (SBOM), Cryptographic Bill of Materials (CBOM), risk assessment, and readiness reporting in improving migration observability across complex infrastructure environments. Analysis of national policy instruments, regulatory artefacts, and early NCII readiness activities identified recurring operational challenges involving accountability for migration scope, incomplete visibility of embedded cryptographic dependencies, inconsistent readiness information, uneven organisational capability, vendor dependency, and constraints in legacy and operational technology environments. The paper contributes practical insights into how developing nations can operationalise PQC migration as a long-term governance and readiness programme rather than a purely technical cryptographic replacement exercise.

Keywords: Post-Quantum Cryptography · PQC Migration · Cybersecurity · National Critical Information Infrastructure · Cryptographic Governance · SBOM · CBOMKeyword 3

1 Introduction

1.1 Background

Advancements in quantum computing are increasingly reshaping discussions surrounding the future of cybersecurity and cryptographic protection. Public-key cryptographic algorithms such as Rivest-Shamir-Adleman (RSA) and Elliptic Curve Cryptography (ECC) currently form the foundation of secure digital communications, authentication systems, digital certificates, financial transactions, and critical online services. These algorithms were originally designed based on mathematical problems considered computationally infeasible for classical computers. However, the emergence of quantum algorithms, particularly Shor's algorithm, has raised concerns that sufficiently capable quantum computers may eventually undermine the security assumptions upon which these cryptographic systems depend [4].

The potential impact extends beyond theoretical cryptography. Government systems, financial institutions, healthcare platforms, cloud infrastructures, defence environments, and other National Critical Information Infrastructure (NCII) sectors rely heavily on cryptographic mechanisms to maintain confidentiality, integrity, authentication, and trust across interconnected digital ecosystems [7]. A successful compromise of these cryptographic foundations would therefore introduce significant operational, economic, and national security implications.

One of the most widely discussed concerns in recent years is the "Harvest Now, Decrypt Later" (HNDL) scenario [1]. Under this model, encrypted information can be collected and stored today by adversaries, even if immediate decryption is not yet possible. Once cryptographically relevant quantum computers (CRQCs) become available, previously captured information may potentially be decrypted retrospectively [21]. This creates particular concern for organisations handling long-lived sensitive information, including government agencies, financial institutions, healthcare providers, defence sectors, and NCII operators.

The challenge becomes considerably more complex within critical infrastructure ecosystems due to the operational characteristics of such environments. Many NCII systems operate over long technology lifecycles and often contain embedded, undocumented, or vendor-controlled cryptographic implementations that are difficult to identify and replace rapidly. Legacy systems, operational technology (OT), industrial control systems (ICS), and multi-vendor infrastructures further complicate migration planning and implementation activities [13]. As a result, transitioning towards PQC is increasingly viewed as a long-term operational and governance challenge, not just a simple technical upgrade.

1.2 Problem Statement

Much of the current PQC migration discussion remains strongly technology-oriented. Existing approaches primarily focus on cryptographic algorithms, protocol replacement, performance considerations, and technical implementation issues. Comparatively less attention has been given to migration governance, operational readiness coordination, and institutional oversight, particularly at national scale [13].

A major obstacle in practice is the lack of cryptographic visibility within many organisations and NCII ecosystems. In numerous cases, organisations do not possess complete awareness of where cryptography is deployed, which cryptographic libraries are being used, how dependencies are interconnected, or how vendor products implement cryptographic functions internally [5]. This significantly complicates migration planning, risk prioritisation, and readiness assessment activities.

In addition, much of the existing literature and migration guidance originates from technologically advanced economies with comparatively mature digital ecosystems and

larger resource capacities. Limited published work currently discusses governance-driven PQC migration from the perspective of developing nations, where migration efforts may be influenced by additional constraints such as fragmented infrastructure ownership, legacy operational dependencies, varying organisational maturity, and limited specialised expertise [9].

While existing governance-oriented studies have begun to discuss PQC readiness, migration maturity, and institutional policy alignment, fewer studies explain how national policy intent is operationalised into enforceable readiness reporting, standardised inventory artefacts, and cross-sector coordination mechanisms for NCII entities. This paper therefore focuses on the operationalisation layer of PQC governance, using Malaysia's early migration experience to examine how strategic direction is translated into cryptographic visibility, readiness reporting, and implementation coordination.

1.3 Contribution of the Paper

This paper contributes an early governance-operationalisation perspective for national-scale PQC migration within NCII ecosystems. Rather than focusing primarily on algorithmic or protocol-level migration, the paper examines how governance coordination, operational readiness management, and migration observability can support national transition activities.

The paper also provides practical insights into the implementation challenges associated with cryptographic inventories, Software Bill of Materials (SBOM), and Cryptographic Bill of Materials (CBOM) within large and complex infrastructure environments. In addition, the study introduces the concept of PQC readiness observability, referring to the ability to continuously assess, monitor, and report migration readiness across interconnected operational ecosystems.

Finally, the paper presents operational lessons and governance considerations that may assist developing nations undertaking large-scale PQC migration initiatives for critical infrastructure protection.

2 Global PQC Migration and Governance Landscape

2.1 International PQC Migration Initiatives

International efforts relating to PQC migration have accelerated significantly over the past decade as governments and cybersecurity agencies increasingly recognise the long-term risks posed by quantum computing. Much of the early global focus centred on identifying and standardising quantum-resistant cryptographic algorithms capable of replacing vulnerable classical cryptographic mechanisms [8].

The United States National Institute of Standards and Technology (NIST) has played a major role through its PQC standardisation programme, resulting in the publication of FIPS 203, FIPS 204, and FIPS 205 [16]. In parallel, the National Security Agency (NSA) introduced the Commercial National Security Algorithm Suite (CNSA) 2.0 to guide migration planning for national security systems and high-assurance environments. Across Europe, organisations such as ENISA, ETSI, BSI, and ANSSI have increasingly emphasised cryptographic agility, phased migration planning, operational preparedness, and long-term resilience [15].

Other countries have also begun developing national-level migration readiness initiatives [2]. The United Kingdom National Cyber Security Centre (NCSC) has focused on organisational readiness, risk-based migration planning, and cyber resilience [19]. At the same time, countries such as China continue to position PQC migration within broader

Table 1: Countries PQC migration initiatives and their primary strategic focus

Country / Region	Main Agency / Initiative	Primary Focus	Key Migration Approach
United States	NIST PQC Standardisation Program	Standardisation of PQC algorithms	Development and standardisation of quantum-resistant cryptographic algorithms
United States	NSA CNSA 2.0	National security systems	Mandatory migration guidance for high-assurance environments
European Union	ENISA	Cybersecurity coordination and readiness	Strategic migration guidance and cryptographic agility
Germany	BSI	National cybersecurity and cryptographic resilience	Hybrid cryptography, phased migration planning, cryptographic inventory
France	ANSSI	Government and critical infrastructure security	Conservative phased migration and high-assurance cryptography
United Kingdom	NCSC	National cyber resilience	Migration guidance and organisational readiness planning
Netherlands	AIVD / CWI / TNO PQC Migration Handbook	National migration readiness	Three-phase migration model: diagnosis, planning, execution
China	National PQC and Quantum Strategy Initiatives	Technological sovereignty and strategic resilience	Parallel focus on PQC, QKD, and domestic cryptographic ecosystems
Malaysia	NACSA /PTPKM National PQC Migration Programme	Governance-driven NCII migration readiness	Inventory-driven migration coordination using cryptographic inventory, SBOM and CBOM

strategies involving technological sovereignty, domestic cryptographic ecosystems, and independent quantum communication networks [1].

Although implementation approaches vary across jurisdictions, current international efforts increasingly recognise that PQC migration extends beyond cryptographic algorithm replacement alone. Migration activities involve cryptographic visibility, dependency management, interoperability, vendor readiness, and operational coordination across interconnected infrastructure ecosystems [6].

Table 1 summarises selected international PQC migration initiatives and their primary strategic focus.

The comparison suggests that most international initiatives are anchored in three dominant orientations: algorithm standardisation, sectoral migration guidance, and organisational readiness planning. Malaysia's approach does not seek to replace these technical and standards-based models. Instead, its contribution lies in translating PQC migration into a national NCII readiness governance mechanism through regulatory reporting, inventory-driven visibility, and centralised coordination. This positions the Malaysian model as an operational governance layer that complements international algorithm standards and migration guidance by addressing accountability, data collection, readiness observability, and coordination across critical infrastructure sectors.

2.2 Governance Gaps in Existing PQC Migration Approaches

Despite progress in PQC standardisation and migration planning, implementation guidance remains more mature at the technical and organisational levels than at the level of national governance operationalisation [9]. Comparatively less attention has been given to governance operationalisation and institutional coordination required to support migration at national scale, particularly across interconnected NCII ecosystems.

In practice, large-scale PQC migration extends beyond cryptographic replacement alone. Migration activities frequently involve coordination across multiple sectors, agencies, infrastructure operators, vendors, and regulatory environments, while simultaneously dealing with long technology lifecycles, legacy infrastructure, and operational dependencies [2, 3]. These challenges are particularly evident in OT and ICS environments, where many systems were not originally designed with cryptographic agility or future migration requirements in mind [1].

Another persistent challenge involves the lack of cryptographic visibility within or-

ganisational environments. Many organisations continue to face difficulties identifying deployed cryptographic assets, embedded cryptographic dependencies, hidden libraries, and third-party cryptographic implementations. Without sufficient cryptographic visibility, migration prioritisation, dependency analysis, readiness assessment, and long-term planning become difficult to perform consistently across large operational ecosystems [11].

Existing literature also provides comparatively limited discussion regarding institutional oversight mechanisms such as readiness governance, migration observability, national reporting structures, and operational coordination models capable of supporting long-term PQC migration activities. These governance gaps are often more pronounced within developing nations, where infrastructure ownership, cybersecurity maturity, operational dependencies, and specialised expertise may vary considerably across sectors [14].

Within this broader landscape, Malaysia's emerging approach introduces a governance-driven operationalisation perspective that places stronger emphasis on centralised coordination, readiness monitoring, and cryptographic visibility through mechanisms such as cryptographic inventories, Software Bill of Materials (SBOM), Cryptographic Bill of Materials (CBOM) and cryptographic risk assessment [18]. Rather than focusing solely on algorithm transition, the approach aims to strengthen migration observability and institutional readiness management across NCII ecosystems.

3 Research Methodology

3.1 Research Design

This study adopts a descriptive-analytical case study approach to examine governance and operational aspects of national-scale PQC migration within Malaysia's NCII ecosystem. The approach is appropriate given the emerging nature of PQC migration and the limited availability of established operational governance models [12]. Using Malaysia as the primary case, the study analyses how strategic migration objectives are translated into operational readiness activities across multiple critical sectors and institutional stakeholders [6]. A qualitative thematic analysis approach is employed to identify recurring governance, coordination, and implementation challenges, with emphasis on migration readiness, cryptographic visibility, and operationalisation patterns observed during early PQC migration activities [20].

3.2 Data Sources

The study utilises a triangulation of data sources to ensure analytical depth and validity. These sources are categorised into three primary layers:

i. **Strategic and Legal Foundations:** National policy instruments including the Malaysia Cyber Security Strategy (MCSS) 2025–2030, the National Cryptography Policy (MyKriptografi), and the Cyber Security Act 2024 [Act 854] [10, 17].

ii. **Regulatory and Operational Artefacts:** Documents associated with Arahan Ketua Eksekutif NACSA No. 9, which provides the mandate for NCII entities to report readiness data. This includes the National PQC Migration Workbook (Buku Kerja), which contains structured tables for SBOM, CBOM, and Risk Registers [18].

iii. **Empirical Observations:** Data derived from stakeholder engagement sessions, technical workshops, and readiness coordination activities involving cross-functional NCII teams.

3.3 Analytical Approach

The study adopts a governance and operational analysis approach to examine how national-level PQC migration coordination is operationalised across interconnected NCII environments. The analysis focuses on governance coordination, migration readiness management, cryptographic visibility, and institutional operationalisation challenges observed during early migration activities. A qualitative thematic analysis method is applied to identify recurring implementation issues, including legacy infrastructure dependencies, cryptographic visibility limitations, vendor-related constraints, and operational complexity associated with large-scale migration environments. Malaysia's migration approach is further analysed comparatively against selected international PQC migration initiatives and guidance frameworks, including those from NIST, ETSI, and the UK NCSC, particularly in areas involving migration readiness, governance coordination, standardisation, and risk management.

The thematic analysis was conducted in three stages. First, strategic, legal, and operational documents were reviewed to identify formal mandates, institutional roles, required artefacts, and reporting obligations. Second, observations from stakeholder engagements, technical workshops, and readiness coordination activities were examined to identify recurring operational issues encountered by NCII entities during inventory, SBOM, CBOM, and risk-assessment activities. Third, the documentary and observational findings were mapped against four analytical themes: governance and accountability, cryptographic visibility, organisational and vendor capability, and legacy infrastructure constraints. Only recurring issues observed across more than one source category were treated as principal findings. Entity-specific and sensitive NCII information was excluded from the analysis.

3.4 Scope and Limitations

This study primarily focuses on the early operationalisation stages of Malaysia's national PQC migration initiative, particularly activities associated with migration preparation, readiness assessment, and governance coordination. As the national migration programme remains in its early phases, quantitative migration metrics and long-term implementation data remain limited. This limitation reflects the broader global situation, where large-scale PQC migration measurement frameworks and operational benchmarks are still evolving. In addition, the study is contextualised within Malaysia's NCII ecosystem and governance environment, which may differ from implementation conditions in other jurisdictions. The findings should therefore be interpreted within the context of an evolving technological, regulatory, and threat landscape, where future developments in quantum computing, cryptanalysis, and PQC standardisation may continue to influence migration strategies and operational priorities.

4 Malaysia's National PQC Migration Ecosystem

The Malaysian model for PQC migration is structured as a centralised institutional orchestration framework, designed to manage transitions across fragmented critical infrastructure sectors. Rather than a standalone technical project, governance is distributed across three functional layers:

i. **Strategic Direction:** The National Cyber Security Committee (JKSN) provides high-level policy mandates, ensuring PQC migration is integrated into national resilience planning.

ii. **National Coordination and Oversight:** The National Cyber Security Agency (NACSA) acts as the central oversight body, responsible for sectoral synchronisation and regulatory enforcement under the Cyber Security Act 2024.

iii. **Technical Coordination:** The Malaysia Cryptology Technology and Management Centre (PTPKM) serves as the technical arm, providing cryptographic advisory services and managing national readiness data.

This hierarchy moves beyond standard IT management by positioning PQC as a national security mandate, facilitating the multi-year funding and cross-departmental alignment necessary for a transition of this scale.

Malaysia's governance-driven migration model is operationalised through a five-phase national roadmap comprising Preparation, Selection, Validation, Implementation, and Monitoring. Unlike conventional technology refresh programmes, the roadmap integrates governance, readiness assessment, dependency visibility, risk management, and implementation activities within a common framework applicable across all NCII sectors. The phased approach enables coordinated migration planning at national scale while supporting continuous readiness monitoring throughout the migration lifecycle. Figure 1 illustrates the five-phase PQC migration strategy and its role in aligning governance, readiness management, and migration execution across Malaysia's NCII ecosystem.

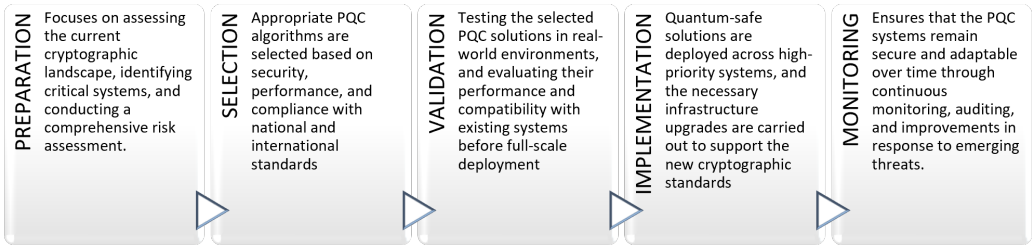


Figure 1: Five-phase PQC migration strategy.

A key differentiator in the Malaysian approach is the shift from voluntary advisories to regulatory enforcement mechanisms. Arahan Ketua Eksekutif NACSA No. 9 serves as a formal regulatory instrument that mandates NCII entities to report their cryptographic readiness. By exercising powers under Section 13 of Act 854, the government establishes a legal basis for "readiness observability," requiring entities to submit structured data via the National PQC Migration Workbook (*Buku Kerja*). This mechanism helps reduce coordination gaps often encountered in national transitions by creating a transparent compliance trail and supporting the identification of maturity gaps across the eleven NCII sectors.

This mechanism overcomes the "institutional void" often seen in national transitions by creating a transparent compliance trail and identifying maturity gaps across the eleven NCII sectors.

Complementing the five-phase migration roadmap, Malaysia has adopted a phased capability-development approach to progressively strengthen national PQC readiness. The initiative began with the establishment of the PTPKM in 2024, providing a dedicated national capability for cryptographic governance, technical coordination, and migration readiness activities. Subsequent efforts focused on building the foundational elements required for migration, including governance structures, national guidance, readiness assessment mechanisms, and supporting technical capabilities. The current phase emphasises coordinated migration preparedness across NCII sectors, while future efforts will focus on implementation, assurance, continuous monitoring, and long-term cryptographic lifecycle management. As illustrated in Figure 2, the approach reflects Malaysia's view of PQC

migration as a multi-year governance and resilience programme rather than a one-off technology replacement exercise.

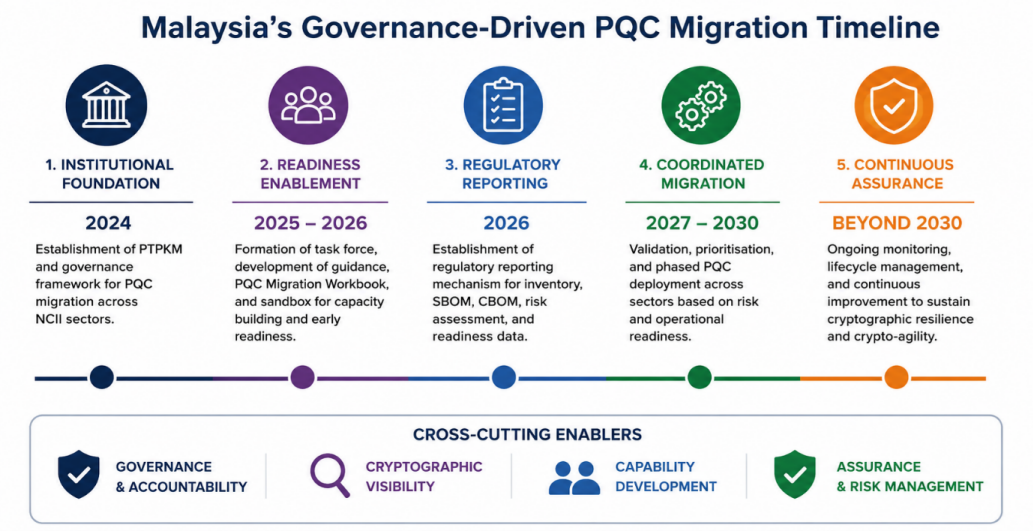


Figure 2: Malaysia's Governance-Driven PQC Migration Timeline

The ecosystem leverages an existing assurance infrastructure to support future PQC adoption. The National Trusted Cryptographic Algorithm List (MySEAL) provides a standardized reference for approved primitives, while the Malaysian Cryptography Validation (MyCV) scheme offers a framework for certifying modules according to international standards like ISO/IEC 19790. Furthermore, the Produk Kriptografi Terpercaya Negara (PKTN) program facilitates the evaluation of trusted cryptographic products for high-assurance environments. While originally algorithm-agnostic, these mechanisms are being reoriented to validate lattice-based PQC standards (e.g., ML-KEM, ML-DSA), providing an assurance baseline for the future validation and deployment of PQC-enabled cryptographic products.

Malaysia's PQC migration ecosystem can therefore be understood as a governance-driven structure supported by policy direction, institutional coordination, cryptographic assurance mechanisms, PQC-specific guidance, and the operational realities of the NCII environment. This ecosystem provides the foundation for the next section, which examines how Malaysia operationalises PQC migration through structured inventories, SBOM, CBOM, cryptographic risk assessment, and readiness reporting.

5 Operationalising PQC Migration within Malaysia NCII Environments

Malaysia's national PQC migration programme adopts a governance-driven operational approach coordinated by NACSA and the PTPKM. Unlike many international initiatives that remain focused primarily on algorithm standardisation and strategic planning, Malaysia's approach emphasises operational readiness, migration observability, and coordinated implementation across NCII sectors.

5.1 Cryptographic Inventory and Visibility

The operational baseline for migration readiness begins with cryptographic inventory and visibility activities implemented through Jadual 0 (Initial Inventory) within the national PQC Migration Workbook (*Buku Kerja*). The inventory process requires NCII entities to identify cryptographic assets, applications, protocols, certificates, libraries, firmware, operating systems, and related dependencies across operational environments.

Early readiness activities revealed significant visibility challenges within NCII ecosystems, particularly involving legacy infrastructure, ICS, SCADA environments, and vendor-managed platforms. In several cases, cryptographic implementations were found to be undocumented, hardcoded, or embedded within proprietary systems, commonly referred to as "Shadow Crypto" [5]. These conditions complicate dependency analysis, migration prioritisation, and operational readiness assessment activities.

5.2 Software Bill of Materials (SBOM)

To strengthen software supply chain visibility, Malaysia incorporates SBOM concepts through Jadual 1 of the migration workbook. The objective is to identify software components, third-party dependencies, and externally sourced libraries potentially containing quantum-vulnerable cryptographic implementations.

The CBOM records specific metadata including algorithms (e.g., RSA, ECC), key lengths, crypto libraries, and data states (rest, transit, use). This tool enables crypto-agility visibility, allowing organisations to assess whether their systems can support a modular switch to quantum-resistant standards. By utilising cryptographic dependency mapping, the CBOM traces every connection to ensure that upgrading one component does not create undocumented interoperability failures across interconnected systems.

5.3 Cryptographic Bill of Materials (CBOM)

To address these limitations, the national migration programme operationalises CBOM concepts through Jadual 2 of the PQC Migration Workbook. CBOM implementation focuses specifically on documenting cryptographic algorithms, key lengths, certificates, protocols, libraries, and dependency relationships across interconnected operational systems.

The CBOM records specific metadata including algorithms (e.g., RSA, ECC), key lengths, crypto libraries, and data states (rest, transit, use). This tool enables crypto-agility visibility, allowing organisations to assess whether their systems can support a modular switch to quantum-resistant standards. By utilising cryptographic dependency mapping, the CBOM traces every connection to ensure that upgrading one component does not cause "silent breakages" in interconnected systems.

5.4 Cryptographic Risk Register and Risk Assessment

Structured cryptographic risk assessment activities are conducted through Cryptographic Risk Registers and Risk & Dependency Assessment activities implemented under Jadual 3 and Jadual 4 of the migration workbook. These activities focus on identifying and prioritising vulnerable cryptographic implementations, particularly RSA- and ECC-based systems expected to be affected by Shor's algorithm.

Migration prioritisation is aligned with operational criticality, confidentiality requirements, system lifecycle considerations, and dependency exposure. Malaysia's approach also incorporates Mosca's Theorem ($X + Y > Z$) to support migration urgency assessment, particularly for systems handling long-term sensitive information. Operational assessments further evaluate interoperability constraints, vendor readiness, computational overhead, and service continuity risks associated with PQC implementation.

5.5 Readiness Reporting and National Migration Coordination

At the national level, PQC migration readiness is coordinated through a structured reporting mechanism established under NACSA Chief Executive Directive No. 9. Under this directive, NCII entities are required to submit migration-related information to NACSA and the PTPKM, including cryptographic inventories, SBOM, CBOM, cryptographic risk registers, and readiness assessment findings.

The reporting mechanism provides NACSA and PTPKM with visibility into the current state of migration readiness across NCII sectors. It enables the identification of cryptographic dependencies, implementation challenges, capability gaps, and areas requiring further intervention or support. The reporting process is aligned with Malaysia's national PQC migration roadmap, covering the Preparation, Selection, Validation, Implementation, and Monitoring phases.

Beyond compliance reporting, the process serves as an important coordination mechanism for tracking readiness progress across interconnected critical infrastructure sectors. By consolidating migration-related information from multiple entities, NACSA and PTPKM are able to develop a clearer understanding of sectoral readiness, emerging risks, and overall migration progress. This reflects Malaysia's gradual shift from strategic planning towards the practical coordination and management of PQC migration at national scale.

6 Initial Findings and Operational Challenges

The early stages of Malaysia's national PQC migration programme have provided valuable insights into the practical challenges associated with preparing organisations for a large-scale cryptographic transition. These observations emerged from migration workshops, readiness assessments, stakeholder engagements, and coordination activities involving NCII entities. While the findings do not represent a complete assessment of national migration maturity, they highlight several recurring issues that may influence future migration efforts.

6.1 Governance and Scope Definition

One of the recurring challenges involves defining migration scope and accountability within complex operational environments. Many critical systems are developed, maintained, or supported by multiple stakeholders, including vendors, service providers, and system integrators. This issue was particularly visible where system ownership, operational support, application development, and cryptographic implementation knowledge were distributed among different parties, resulting in uncertainty over who should provide inventory data, validate cryptographic dependencies, and approve migration priorities.

6.2 Cryptographic Visibility and Readiness Information

Limited visibility over deployed cryptographic implementations remains a significant challenge. During readiness activities, participants were generally able to identify major systems and applications, but additional effort was often required to determine embedded libraries, certificate dependencies, key-management mechanisms, and vendor-controlled cryptographic functions. While some organisations maintain comprehensive system documentation, others have cryptographic functions embedded within applications, middleware, hardware appliances, or vendor-controlled platforms that are difficult to identify and assess. Consequently, many organisations must first establish baseline cryptographic inventories before meaningful migration planning can take place.

The quality and consistency of readiness information also vary considerably across organisations. Differences in documentation practices, terminology, and reporting granularity can affect dependency analysis, risk prioritisation, and national-level readiness assessment.

These findings reinforce the importance of standardised reporting structures and common assessment methodologies.

6.3 Capability and Vendor Dependency Challenges

The study also identified varying levels of technical readiness across participating organisations. These observations indicate that PQC migration requires not only cybersecurity expertise, but also coordinated input from application owners, procurement teams, enterprise architects, vendors, and senior management. However, familiarity with concepts such as cryptographic inventories, SBOM, CBOM, and crypto-agility is not always consistent across these groups.

This challenge is compounded by heavy reliance on vendor-managed technologies. In many cases, organisations have limited visibility into the cryptographic components embedded within proprietary products and depend on vendors for information regarding algorithm support, upgrade paths, and future migration plans. Such dependencies may affect migration timelines and complicate readiness assessment activities.

6.4 Legacy Infrastructure Considerations

Legacy systems and operational technology environments present additional migration challenges. Many of these systems operate over long lifecycles, have limited upgrade options, and support critical services with strict availability requirements. Consequently, migration planning cannot rely solely on conventional ICT replacement approaches and must account for operational continuity, interoperability, testing requirements, and service disruption risks.

For systems that cannot be migrated immediately, interim risk treatment may include segmentation, restricted exposure, enhanced monitoring, accelerated replacement planning, vendor assurance requirements, and, where technically feasible, validated transitional or hybrid cryptographic mechanisms. These measures should be treated as time-bound compensating controls rather than substitutes for eventual migration.

Overall, the findings suggest that successful PQC migration depends not only on the availability of quantum-resistant algorithms but also on governance coordination, cryptographic visibility, organisational capability, vendor engagement, and effective management of legacy infrastructure dependencies. These factors are likely to remain important considerations as Malaysia progresses towards later phases of its national migration roadmap.

7 Discussion

The findings suggest that PQC migration should be viewed as more than a cryptographic modernisation effort. While much of the global discourse remains centred on algorithm selection and technical implementation, the Malaysian experience indicates that the more difficult challenge lies in coordinating migration across complex organisational, sectoral, and infrastructure environments. In practice, the transition affects governance arrangements, operational processes, supply chain dependencies, and long-term resilience planning as much as it affects cryptographic technologies themselves.

7.1 PQC Migration as a Governance and National Resilience Challenge

The transition to PQC represents a fundamental shift that extends beyond routine software upgrades or isolated cryptographic replacement activities. Unlike emerging technologies that may operate as standalone systems, cryptography is deeply embedded across digital infrastructures, operational technologies, communication protocols, and trust architectures

that support modern national ecosystems. As a result, PQC migration requires coordinated transformation across multiple operational layers rather than direct algorithm substitution alone.

The Malaysian experience underscores that successful transition depends on governance coordination that aligns high-level policy mandates with cross-sectoral execution. A failure to synchronise this effort across interconnected sectors creates systemic vulnerabilities where a compromise in one sector could trigger cascading failures across the national economy. Consequently, PQC migration must be approached as a continuous national resilience programme rather than a finite technical project.

7.2 Towards Risk-Adaptive PQC Governance

A static migration approach is unlikely to remain effective within the rapidly evolving quantum threat landscape. Instead, organisations should adopt risk-adaptive governance models that support continuous readiness assessment and ongoing migration planning. In practice, PQC readiness should be treated as a recurring operational responsibility rather than a one-time implementation exercise.

Given the scale and complexity of NCII environments, migration prioritisation must also remain risk-driven. Concepts such as Mosca's Theorem ($X + Y > Z$) provide a practical basis for identifying systems requiring urgent migration, particularly where confidentiality lifecycles and migration timelines may already exceed projected quantum threat timelines.

Equally important is the development of crypto-agility as a long-term operational capability. Rather than focusing solely on replacing vulnerable algorithms, organisations must design operational environments capable of supporting future cryptographic transitions with minimal architectural disruption. This becomes increasingly important as PQC standards, implementation guidance, and quantum threat projections continue to evolve.

7.3 Implications for Developing Nations

Malaysia's operational experience provides several observations that may be relevant for other developing nations undertaking PQC migration initiatives. One notable finding is the importance of centralised governance coordination through dedicated national institutions such as NACSA and PTPKM. This approach helps address common challenges within developing economies, including fragmented infrastructure governance, inconsistent organisational maturity, and limited specialised cryptographic expertise.

The study also highlights the importance of establishing migration observability mechanisms through structured readiness reporting, cryptographic inventories, SBOM, and CBOM activities. Without baseline visibility over cryptographic dependencies and operational exposure, large-scale migration planning becomes difficult to coordinate effectively at national level.

At the same time, developing nations may face additional constraints involving funding limitations, legacy infrastructure dependencies, and reliance on foreign technology ecosystems. Early establishment of national migration frameworks, regulatory mandates, and institutional coordination mechanisms therefore becomes increasingly important to strengthen long-term technological resilience and national cryptographic sovereignty.

7.4 Practical Implications for Policymakers and NCII Operators

The findings have practical implications for both policymakers and NCII operators. For policymakers, PQC migration requires clear accountability structures, standardised readiness artefacts, secure handling of cryptographic inventory data, and mechanisms for monitoring sectoral progress without exposing sensitive operational information. National coordination bodies should also provide common terminology, minimum reporting requirements,

escalation channels, and capability-building support for sectors with lower cryptographic maturity.

For NCII operators, the immediate priority is to establish accountable ownership for cryptographic assets, maintain updated inventory, SBOM and CBOM records, identify quantum-vulnerable dependencies, and integrate PQC readiness into procurement, vendor management, technology refresh, and risk management processes. Vendor contracts should require disclosure of cryptographic dependencies, PQC support plans, and upgrade timelines. These measures would help transform PQC migration from a one-off compliance exercise into a sustained cryptographic lifecycle management practice.

7.5 Limitations of the Study

This study reflects the early stages of PQC migration, with the analysis primarily focused on readiness assessment, governance coordination, and migration planning activities. As large-scale deployment efforts are still ongoing across many sectors, empirical implementation data and quantitative migration metrics remain limited.

In addition, the findings are based on Malaysia's regulatory and operational environment and may not be directly transferable to jurisdictions with different governance structures or critical infrastructure ecosystems. Finally, the PQC landscape continues to evolve alongside developments in quantum computing capabilities and international standardisation efforts. Consequently, migration strategies and governance approaches may require further refinement as technologies, threats, and implementation practices mature.

8 Conclusion

Successful PQC migration requires more than the replacement of quantum-vulnerable algorithms. This study demonstrates that governance, coordination mechanisms, operational readiness assessment, and risk-informed decision-making are critical components of a sustainable national-scale migration strategy.

The paper presented a governance-driven operationalisation model for PQC migration within NCII, integrating Inventory, SBOM, CBOM, risk assessment, and readiness reporting into a structured pathway from policy intent to operational implementation. The findings further show that inventory activities alone do not provide sufficient visibility for migration planning without systematic interpretation of cryptographic dependencies, operational risks, and organisational readiness.

The Malaysian experience illustrates how existing national cybersecurity and cryptography governance structures can support coordinated PQC migration across multiple critical sectors. Future work should focus on developing quantitative PQC readiness indicators, sector-specific maturity assessment models, and data-quality metrics for inventory, SBOM, CBOM, and risk-register submissions. Further research may also examine how vendor dependency, legacy infrastructure, and regulatory reporting affect progression from readiness assessment to validation, deployment, and continuous monitoring across NCII sectors.

References

[1] Aziz Alghamdi. Post-quantum cryptography implementation challenges: Security implications for critical infrastructure. 2025.

[2] Robert Campbell. Estimating migration timelines for enterprises transitioning to post-quantum cryptography: A real-world analysis of dependencies and interconnectedness. 2025.

[3] P. Chammas, K. Hariss, C. Bassil, and M. Chamoun. Towards a Holistic Risk Framework for PQC Migration in Legacy Systems. https://www.qurisk.fr/s/Towards_a_Holistic_Risk_Framework_For_PQC_Migration_in_Legacy_Systems.pdf. Accessed: 2026-05-25.

[4] Lily Chen, Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray A Perlner, and Daniel Smith-Tone. *Report on post-quantum cryptography*, volume 12. US Department of Commerce, National Institute of Standards and Technology . . . , 2016.

[5] Brian Couzens. Quantum risk: A strategic framework for pqc migration and board accountability. 2025.

[6] CyberSecurity Malaysia. Post-Quantum Cryptography Migration Framework, Malaysia. https://mykripto.cybersecurity.my/index.php/services/post-quantum-cryptography-initiatives/documents. Accessed: 2026-05-25.

[7] DigiCert. *Post-Quantum Cryptography For Dummies, DigiCert Special Edition*. John Wiley & Sons, Inc. / DigiCert, Hoboken, NJ, 2025. Available at https://www.digicert.com/campaigns/pqc-for-dummies-ebook.

[8] Mustafa El Bizri, Ahmad M El-Hajj, Layth Sliman, and Ali Massoud Haidar. Institutional approaches to post-quantum cryptography: A comparative analysis of migration frameworks. *IEEE Access*, 14:3259–3283, 2026.

[9] Volkan Erol. Quantum readiness in cryptography: A maturity-based framework for post-quantum transition. 2025.

[10] Government of Malaysia. Cyber Security Act 2024 [Act 854]. https://lom.agc.gov.my/act-detail.php?language=BI&act=854, 2024. Accessed: 2026-05-03.

[11] Khondokar Fida Hasan, Leonie Simpson, Mir Ali Rezazadeh Baee, Chadni Islam, Ziaur Rahman, Warren Armstrong, Praveen Gauravaram, and Matthew McKague. A Framework for Migrating to Post-Quantum Cryptography: Security Dependency Analysis and Case Studies. *IEEE Access*, 12:23427–23450, 2024.

[12] Ini Kong. Transitioning Towards Quantum-Safe Government: Examining Stages of Growth Models for Quantum-Safe Public Key Infrastructure Systems. In *Proceedings of the 15th International Conference on Theory and Practice of Electronic Governance*, pages 499–503, 2022.

[13] Ini Kong, Marijn Janssen, and Nitesh Bharosa. Challenges in the transition towards a quantum-safe government. In *Proceedings of the 23rd Annual International Conference on Digital Government Research*, pages 282–292, 2022.

[14] Silong Li, Yuxiang Chen, Lin Chen, Jing Liao, Chanchan Kuang, Kuanching Li, Wei Liang, and Naixue Xiong. Post-Quantum Security: Opportunities and Challenges. *Sensors*, 23(21):8744, 2023.

[15] Lukas Malina, Patrik Dobias, Jan Hajny, and Kim-Kwang Raymond Choo. On deploying quantum-resistant cybersecurity in intelligent infrastructures. In *Proceedings of the 18th International Conference on Availability, Reliability and Security*, pages 1–10, 2023.

[16] Dustin Moody, Ray Perlner, Andrew Regenscheid, Angela Robinson, and David Cooper. Transition to post-quantum cryptography standards. Technical report, National Institute of Standards and Technology, 2024.

[17] National Cyber Security Agency (NACSA), Malaysia. MyKriptografi: Dasar Kriptografi Negara. https://www.nacsa.gov.my/kriptografi.php. Accessed: 2026-03-16.

[18] National Cyber Security Agency (NACSA), Malaysia. Arahan KE NACSA No. 9. https://www.nacsa.gov.my/legal.php, September 2025. Accessed: 2025-09-23.

[19] National Cyber Security Centre. Timelines for migration to post-quantum cryptography. https://www.ncsc.gov.uk/sites/default/files/pdfs/publication/pqc-migration-timelines.pdf, March 2025. Accessed: 2026-5-25.

[20] Nurhaliza Nurhaliza and Adichandra Febryana Yuscatama Darmawan. Zero-Trust Architecture Adaptation for Post-Quantum Cryptography: Implementation Roadmap for Critical Infrastructure. *Jurnal Ekonomi Teknologi dan Bisnis (JETBIS)*, 4(5):225–242, 2025.

[21] Jeffrey Witty. Securing the future: Mandating quantum-resistant encryption for us critical infrastructure. *Available at SSRN 5321016*, 2025.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 394–412.

QRAF: A Quantum Risk Assessment Framework for Post-Quantum Cryptographic Readiness, Migration Prioritization and National Cyber Resilience

Solahuddin Shamsuddin, Noorul Halimin Mansol, Muhammad Sufyan Basri and Nor Effah Aina Safari

Malaysia Cryptology Technology and Management Centre, Universiti Putra Malaysia, 43400 Serdang Selangor, Malaysia
{solahuddin,noorulhalimin,sufyanbasri,effahaina}@upm.edu.my

Abstract. The use of post-quantum cryptography (PQC) has become increasingly relevant to cyber-defence policy because quantum computers threaten traditional asymmetric cryptography. Current PQC guidance focuses on standardisation and algorithm deployment, but insufficient attention has been given to quantum-risk assessment and migration prioritisation. This paper presents the Quantum Risk Assessment Framework (QRAF), which integrates cryptographic-asset discovery, quantum-risk assessment, PQC migration priorities and governance. QRAF introduces the Quantum Risk Score (QRS), a multidimensional method considering vulnerability, exploitability, quantum relevance, data longevity, criticality, migration readiness and sovereignty sensitivity. Illustrative migration scenarios show that QRAF produces reasonable, guidance-consistent priorities while elevating long-lived assets and sovereignty-sensitive infrastructure. Entropy Sovereignty is introduced as a governance concept placing sovereign control of cryptographic assets, trust anchors and migration at the centre of national cyber resilience.

Keywords: Post-Quantum Cryptography · Quantum Risk Assessment · PQC Migration · Crypto-Agility · Entropy Governance · Quantum Governance

1 Introduction

Quantum computing represents a very obvious long-term risk to public key cryptography and digital trust infrastructure [11][17]. Cryptographic algorithm which has been used widely like RSA, ECC, DH, ECDH and ECDSA currently to ensure secure communications, digital identity, financial transaction, cloud services, defence systems and the security of National Critical Information Infrastructure (NCII) as a whole. Nonetheless, a powerful quantum computer which is known as 'Cryptographically Relevant Quantum Computer (CRQC)' could break current classical cryptographic algorithms in the near future using Shor's algorithm, while

Grover's algorithm can reduce the strength of the security of symmetric encryption [17][8]. Even though currently there is no real usage of CRQC, the usage of 'Harvest Now, Decrypt Later (HNDL)' technique by perpetrators demonstrates the quantum risks have now began. All encrypted information today can be harvested and decrypted in the future once CRQC computers become available. To address this issue, a lot of organizations dan governments all over the world hasten their efforts to standardize the usage of PQC algorithms. For example, National Institute of Standards and Technology (NIST) have

produced their first PQC algorithm standards known as Federal Information Processing Standards (FIPS) 203, 204 and 205. Apart from that, Commercial National Security Algorithm (CNSA) 2.0, National Security Agency (NSA) have produced PQC migration guidelines for Quantum-resistant systems. Other organizations such as such as the European Telecommunications Standards Institute (ETSI), the European Union Agency for Network and Information Security (ENISA), the International Organization for Standardization (ISO), and the UK's National Cyber Security Centre (NCSC), have also issued guidelines which focuses on crypto-agility, interoperability and phased transition approach to ensure smooth migration to PQC-ready system. These efforts are for mitigating potential risks related to quantum computers and to ensure long term security for confidential information. The problem is: most of the efforts are about to acquire the right technology and adherence to guidelines not to think thoroughly on the potential big risks and how to prepare the nation to be quantum safe. This paper will address the following questions:

i. How governments and industries can prioritize PQC migration systematically when crypto inventory have not been identified and a vague PQC migration timeline?.

ii. How multi-dimensional quantum risk score can be used in a PQC migration-ready model? This paper contributes:

iii. A structured quantum-risk assessment methodology.

iv. A multi-dimensional Quantum Risk Score (QRS) model.

v. A governance-based national PQC readiness framework.

vi. A prioritisation methodology for critical sectors.

vii. Hybrid migration strategy and crypto-agility.

viii. The idea of Entropy Sovereignty as a governance concept.

ix. ASEAN-oriented interoperability and readiness model. This is how the sequence of this paper is organized. First, we will look closely quantum computer threats, and how we can move forward in chapter 2. Next, in chapter 3, we will study what have been done in this area and see if there is still gap in our understanding. Next, we will introduce QRAF methodology and its architecture in chapter 4. Next, in Chapters 4 and 5, we will explain how to discover crypto assets and their vulnerabilities. Next in chapter 6, we will deep dive QRS marking model and how it is being used for validation of different scenarios. Chapter 7 will comprise of the importance of governance, PQC ready assessment and how to become crypto-agile. We will also discuss how ASEAN nations can work together and share information across borders in chapter 8. Next, in chapter 9, we will give some suggestions to develop strategies to address these quantum threats. Finally, we will end this paper in chapter 10, summarizing what have been discussed and propose some ideas for future work.

2 Quantum Threat Landscape

2.1 Quantum Computing and Cryptographic Disruption

Quantum computing represents paradigm shifts in a calculation model using quantum bit (qubit) which can operate in a superposition state and exploiting entanglement to do parallel processing. From cyber security perspective, our main concerns are from two quantum algorithms which are very well known. Shor's algorithm can solve big integer factorization and discrete problem effectively while Grover's algorithm will reduce attack

difficulty towards symmetric algorithm. [8] Shor's algorithm will also raise an alarm to other public key cryptography schemes like RSA, ECC, DH, ECDH and ECDSA. These algorithms are the foundation of secure communications, digital identity management, trusted infrastructure used by the government, financial institution, cloud provider and National Critical Information Infrastructure (NCII). As soon as

CRQC arrives, those systems using classical cryptographic algorithms will not be able to give enough security assurance. Even though large scale CRQC has not matured, the prospect of it to disrupt future cryptography protocols have started to affect the long-term security strategy and PQC migration plan.

2.2 Harvest Now, Decrypt Later (HNDL)

The immediate quantum threat is "Harvest Now, Decrypt Later, (HNDL)". The HNDL would intercept and harvest encrypted communications and encrypted confidential information with the intention to decrypt those encrypted information as soon as CRQC facilities are available in the future. These concerns are phenomenal for data which requires long term secrecy. Example will include confidential government communication, defense intelligent data, health records, financial archive, law related document, diplomatic documents, strategic intelligent property documents and any other sensitive information which must be protected securely in the range of 5 to 10 years. For these kinds of data, the issue is not the current inability to break the encrypted data, but our concern is to encrypt the data such that it will not be broken by CRQC computers in the near future. As a result, organizations with long term confidential data need to plan PQC migration earlier before CRQC becomes a reality.

2.3 Global Standardization Momentum

A few international organizations and cyber security agencies have increased the effort for standardization and migration of their respective PQC. The trend is now shifting from Research and Development to practical applications. Table 1 shows an overall picture on main key initiatives globally on PQC migration and governance.

Table 1: Global standardisation key initiatives.

Ser.	Organisation	Key initiative
1	NIST	FIPS 203, FIPS 204 and FIPS 205 PQC standards [14, 13, 15]
2	NSA	CNSA 2.0 migration guidance [16]
3	ETSI	PQC interoperability and migration studies [5, 6]
4	ISO	Cryptographic module security requirements [9]
5	UK NCSC	National migration guidance [18]
6	ENISA	PQC risk and migration analysis [4]
7	CSA Singapore	Quantum-safe cryptography advisory [2]

This development shows that PQC migration not only be seen as a concern of future theories, as a lot of stakeholders have understood that PQC readiness is mandatory for practical operational and strategic requirements.

3 Related Work

A lot of standards body are improving their efforts in addressing PQC migration issues and quantum resilience security, but most of their efforts are focusing on algorithm selection, execution agility and operational interoperability and not on structured risk assessment

based on governance and national readiness measurement. Standards organizations like NIST has published PQC standards which include FIPS 203, FIPS 204 dan FIPS 205 [14], [13], [15], while CNSA 2.0 NSA program provides roadmap based on crypto suite and phases for national security system [16]. International and European actors including ETSI, ENISA, ISO and NCSC UK, have published PQC migration reports, interoperability study and proposals based on policy which emphasize on crypto-agility, hybrid usage strategy and risk tolerance on special sectors [5], [18], [6], [4], [9]. NIST SP 800-53 also provides security and privacy controls for information system which can be mapped to the PQC migration capabilities [12]. Nonetheless, these frameworks tend to consider cryptography migration as a technical or compliance monitoring exercises only. They offer limited support for systematic quantum exposure scoring system, dependency mapping across complex national infrastructure and prioritizing migration activities based on data lifespan, critical operations and geopolitical

sovereignty concerns. Existing frameworks have limited discussions about integrated governance structure, operational resilience, readiness measurement and national strategic alignment into one integrated migration framework. Baseri et al. proposed assessment framework for quantum risk assessment which describes PQC based risks across vulnerabilities at algorithm level, certificate and current level premigration, during migration and post-migration protocols [1]; Nonetheless, even though the framework gives a structured opinion on technical risk exposure and mitigation choices especially based on technical risk modelling but nonetheless it does not explicitly integrate sensitive sovereignty indicators, multi-dimensional risk scoring for national level prioritization, or development of regional based interoperability.[12]. Recently, a graph-based framework for cryptography risk modelling in a post-quantum era, Systematic Quantum Risk Scoring (S-QRS) which aggregates primitive level vulnerabilities into single risk matrix scale using dependency modelling based on graph and Shapley style attribution [10]; this approach can be used to indicate cascading failures and technical dependency structure, but the resulting S-QRS which specifically designed for detail technical analysis and not driven by high-level policy prioritization [10]. In parallel, some of the Quantum Risk Analysis (QRA) and industrial based readiness tools have emerged, including Quantum Readiness Assessment & Migration Methodology (QRAMM) and SISA's Quantum Risk Assessment for financial institutions, as well as commercial products like PQC risk assessment framework Acubed.IT, this model usually focuses on cryptography asset discovery, risk classification and migration roadmap planning at enterprises, which usually use QRA based score and CBOM/CBOM based inventory. Meanwhile, CyberSecurity Malaysia has produced National PQC Migration Framework tailored especially for Malaysian NCII, with technical usage guidelines and post quantum cryptography implementation solution by phases [3]; even though this framework provides regional context and important policy alignment, its existence remain especially national migration roadmap plan and not as a model for multi-dimensional quantum risk scoring or ASEAN level governance framework for regional resilience scale [3]. On the other hand, QRAF goes beyond algorithm and governance-oriented guidance with the introduction of multi-dimensional quantum risk scoring model, structured ecosystem for national level interoperability assessment and Entropy Sovereignty as governance

architecture explicitly for sovereign control on cryptography and migration policy infrastructures. QRAF also integrates crypto asset discovery, migration prioritization, sectoral readiness assessment and national cyber resilience measurement into one integrated framework with the intention to support both decision makers at technical and policy levels. Such that, most of the existing frameworks give attention to X – either enterprise level technical risk assessment across algorithm, protocol or readiness with QRA style scoring – meanwhile QRAF adds Y and Z with the introduction of: (1) multi-dimensional Quantum Risk Score (QRS) which integrate technical, operational, and sensitive sovereignty into one integrated governance-based model and (2) the development of national and ASEAN

scale resilience like Entropy Sovereignty and ASEAN PQC readiness index which explicitly relates PQC migration to the national risk management and cross border interoperability narratives.

4 QRAF Methodology and Architecture

4.1 Framework Objectives

QRAF aims to assist government and organizations to develop a systematic approach towards the transformation of their IT infrastructure to reduce the potential threats evolved as a result of quantum technology to their cyber security. This framework is developed based on existing migration guidelines and governance principles developed by NIST, ETSI, ENISA and other national security stakeholders [14][5][18][4]. It is more than considering especially PQC migration as a cryptography upgrade, QRAF sees this migration as a challenge related to organization's resilient and governance. This framework is designed to:

i. Discover cryptographic dependencies in IT systems and infrastructure.

ii. Evaluate quantum-related exposure and impact on operations.

iii. Prioritize migration efforts based on risk and criticality assessments.

iv. Enhance organisational and national cyber-resilience.

v. Support PQC migration planning through governance.

vi. Ensure operational continuity throughout the process of phased migration All of these objectives will ensure migration process and cyber security planning can be connected effectively.

4.2 Core Framework Lifecycle

QRAF uses a six-phase operational lifecycle: i. Identify, ii. Inventory, iii. Assess., iv. Priorities,

i. Mitigate and vi. Monitor. It is very important to observe that this life cycle not necessarily be linear but on the other hand it is cyclic and repeating. Cryptographic environment is constantly changing as a result of updates in system configuration, changes in dependencies and receiving new cyber threat intelligence reports. Apart from that, PQC standards and guidelines will evolve in coming years. As a result, organizations must always be ready to repeat some processes in the relevant phase as and when required. For example, increased visibility in the cryptographic inventory may expose new dependency, while changes in the operational requirement or geopolitics may influence priority level given to a particular system. QRAF emphasizes the importance of continuous reassessment compared to doing assessment once and only driven by compliance.

4.3 Layers of QRAF Architecture

QRAF uses layered architecture to connect technical challenges related to cryptographic exposure to higher strategic governance aspects, organization's resilient and decision at the national level. QRAF frameworks consist of 6 layers. Governance layer sets strategic direction, accountability, policy compliance and migration governance responsibility. It is responsible to ensure issues with regard to national level PQC migration, alignment

with guidelines and the management of trust anchor to be addressed effectively. Risk layer deals with identifying and assessment of related quantum risks. It executes risk score, prioritization, reports and required decision support function for migration planning. Technical layer will include cryptography discovery, protocol analysis, dependency analysis and migration planning. It may also include certificate discovery, CBOM analysis, crypto-agility assessment and hybrid usage analysis.

Compliance layer ensures compliant to the standards, certificate requirement and auditability. This includes compliant to the newly published PQC migration guidelines by NIST, ETSI and ISO. Operation layer handles technical aspects which include trust anchor rotation, operational continuity, migration alignment and service resilient. Monitoring layer provides continuous visibility, readiness tracking and migration monitoring.

5 Crypto-Asset Discovery and Exposure Mapping

5.1 Cryptographic Asset Discovery

One of the main obstacles to PQC migration is the limited transparency of cryptographic sources. Organizations usually have a complex ICT infrastructure where cryptography components have been added from time to time to the legacy system, cloud technology, embedded devices and third-party tools. As a result, a lot of international studies have highlighted the need for effective cryptographic inventory for the PQC transition plan [5][4][12]. QRAF offers new detailed discovery methodology which has been designed to improve the visibility of both operational and strategic cryptographic assets. Cryptographic management involves a lot of aspects like environmental PKI, TLS certificates, VPN network, embedded system, IoT infrastructure, cloud solutions, HSM, mobile applications and satellite communications. QRAF integrates the concept like Cryptographic Bill of Materials (CBOM), Software Bill of Material (SBOM), certificate discovery and dependency assessment. This approach enables organizations to detect hidden cryptographic dependencies which might be ignored during migration process. A lot of operational risks do not come from their main cryptographic system itself, but from hidden dependency in applications, hard coded software, middleware or software supply chain. With better inventory management, organizations can make a better estimation on the complexity of migration process, identify which priority area for recovery and assess long term operational exposures.

5.2 Quantum Exposure Mapping

As a way forward, QRAF assess exposures based on two main criteria – exposure level and critical operation. The aim of this assessment is to differentiate between systems which are exposed to quantum risk and the failure of a system which may affect operational matters or cause severe disruption at the national level. Assets are categorized as having small exposure, medium, high or hierarchical depending on accessibility, dependency to exposed primitive cryptography, network continuity and operational importance. High priority assets may include NCII, military communication system, financial transaction system, tiered government environment and healthcare storage facilities. All of these assets have high cryptographic exposure and high operational importance which make their migration very relevant. Exposure mapping may enable relevant environmental dependency analysis. Usually, vulnerabilities in one system may negatively affect trust relationships, authentication process or operational continuity in other systems. As a result, QRAF considers exposure in the context of the whole ecosystem instead of in an individual system.

5.3 Data Sensitivity and Longevity Model

To ease the migration process, QRAF proposes sensitive period model which considers HNDL risks. QRAF is aware of the urgent requirements of migration depend not only on current operational requirements of the system but also on the secrecy duration of the information. Table 2 shows a sample of data classification with its corresponding secrecy duration.

Table 2: Data classification and required secrecy duration.

Ser.	Classification	Example	Secrecy duration
1	PUBLIC	Public websites	Minimal
2	RESTRICTED	Internal operations	1–5 years
3	CONFIDENTIAL	Commercial records	5–10 years
4	SECRET	National security data	10–25 years
5	TOP SECRET	Strategic intelligence	25+ years

This approach will be so relevant when dealing with data interception which may become important strategically for a long-term period. Some examples of this data are healthcare information archive, diplomatic communication and long-term industrial research archive. Thus, migration prioritization with consideration of data life span may be possible with QRAF.

6 Quantum Risk Scoring and Prioritization

6.1 Quantum Risk Score (QRS) Methodology

To support a more structured migration and transparent, QRAF introduces QRS, a new multidimensional scoring model used to evaluate quantum risk exposure across technical, operation and governance domains. This objective was never meant to forecast accurate timeline on cryptographic failure but to assist organization prioritizing migration activities in an unpredictable environment. QRS model evaluates 7 dimensions which collectively influence the urgency of PQC migration.

Table 3: Quantum risk dimensions.

Ser.	Dimension	Description	Scale
1	Vulnerability (V)	Weakness of deployed cryptographic algorithms	0–5
2	Exploitability (E)	Exposure and accessibility of the asset	0–5
3	Quantum Relevance (Q)	Dependence on quantum-vulnerable cryptography	0–5
4	Data Longevity (D)	Required confidentiality duration	0–5
5	Criticality (C)	Operational and national impact	0–5
6	Migration Readiness (M)	Preparedness for migration	0–5
7	Sovereignty Sensitivity (S)	Strategic importance to national sovereignty	0–5

The usage of 0-5 scale enables the framework represents both situations – low risks and high risks clearer. Score 0 represents exposure which can be discarded while score 5 represents the highest concern in that dimension which must be prioritized.

6.1.1 Weighted Scoring Model

As some of the dimensions have higher long-term strategic values compared to other dimensions, QRAF introduces weighting coefficient for each dimension. The weighted scoring model is defined as follows:

$$QRS_w = w_1V + w_2E + w_3Q + w_4D + w_5C + w_6(5 - M) + w_7S. \quad (1)$$

Here, V is vulnerability, E exploitability, Q quantum relevance, D data longevity, C criticality, M migration readiness, S sovereignty sensitivity, and $w_1, \ldots, w_7$ are weighting coefficients. Migration readiness uses $(5 - M)$ because lower readiness increases risk. Table 4 shows the representative weighting coefficient for each dimension.

Table 4: Representative weighting coefficients.

Ser.	Dimension	Weight
1	Vulnerability (V)	1.0
2	Exploitability (E)	1.0
3	Quantum Relevance (Q)	1.2
4	Data Longevity (D)	1.3
5	Criticality (C)	1.3
6	Migration Readiness $(5 - M)$	1.0
7	Sovereignty Sensitivity (S)	1.2

Higher weighting coefficient value is assigned to Quantum Relevance (Q), Data Longevity (D), Criticality (C) and Sovereignty Sensitivity (S) because these dimensions can have great influence to national quantum risk exposures and long term operations under HNDL's assumption. Maximum Weighted Score and Normalisation The maximum possible weighted score across all dimensions is:

$$QRS_{\max} = (5 \times 1.0) + (5 \times 1.0) + (5 \times 1.2) + (5 \times 1.3) \quad (2)$$

$$+ (5 \times 1.3) + (5 \times 1.0) + (5 \times 1.2) = 40.0. \quad (3)$$

This value represents the maximum weighted quantum-risk exposure in the framework.

6.1.2 Normalisation

To improve interpretability across dimensions, the weighted score is normalised to a 0–100 scale:

$$QRS_{\text{norm}} = \frac{QRS_w}{QRS_{\max}} \times 100. \quad (4)$$

This ensures that QRS remains between 0 (minimum risk) and 100 (maximum risk). The range determines migration priority as shown in Table 5.

Table 5: QRS range and priority.

Ser.	QRS range	Priority
1	80–100	Immediate
2	60–79	High
3	40–59	Medium
4	0–39	Low

This category intends to support migration prioritization based on governance across a wide scale of infrastructure environment. The additive scoring structure was chosen based on operational transparency and usability. Policy maker and governance stakeholders often require easy to understand model which can be deployed for prioritization exercise immediately. Even though a more advanced model like dependency graph or statistical

probability may increase the statistical accuracy, it may also reduce the interpretability and complicate the operationalization of the framework across a wide range of environments. As a result, this QRS model which represents a baseline based on a practical governance which can still be detailed out using expert's evaluation, sector's weighting coefficients or extended probability in future works.

6.2 Worked Example and Scenario Validation

To evaluate the practical usability of QRAF, a few infrastructure scenarios have been developed using synthetic assumption but realistic in term of operation which have been documented in existing PQC migration guidance is general, cyber security governance best practices and NCII model. The aim of this scenario is not to validate the empirical usage in the real world, but to demonstrate how QRS methodology can be used to support PQC migration priority in a different operational situation. The first scenario considers National PKI environment hypothesis. National PKI infrastructure normally represents one of the critical digital trust infrastructures as it supports certificate issuance, validation, secure communication, identity verification and trusted national ecosystem in a large scale. For this scenario, assigned score assumptions as shown in Table 6 is used.

Table 6: Assigned score assumptions.

Ser.	Dimension	Assigned score
1	Vulnerability (V)	5.0
2	Exploitability (E)	5.0
3	Quantum Relevance (Q)	5.0
4	Data Longevity (D)	5.0
5	Criticality (C)	5.0
6	Migration Readiness $(5 - M)$	1.0
7	Sovereignty Sensitivity (S)	5.0

Table 7 shows the detailed QRS calculation for National PKI environment hypothesis.

Table 7: Sample QRS calculation—National PKI.

Ser.	Dimension	Score	Weight	Contribution
1	Vulnerability	5	1.0	5.0
2	Exploitability	5	1.0	5.0
3	Quantum Relevance	5	1.2	6.0
4	Data Longevity	5	1.3	6.5
5	Criticality	5	1.3	6.5
6	Migration Readiness	1	1.0	4.0
7	Sovereignty Sensitivity	5	1.2	6.0
		Total weighted score		39.0

Using the weighting configuration defined earlier:

$$QRS_w = 39.0, \tag{5}$$

$$QRS_{\text{norm}} = \frac{39.0}{40.0} \times 100 = 97.5\%. \tag{6}$$

This positions the asset in the "Immediate" migration prioritization category. The national PKI scenario receives a high score because it widely integrates high dependency to a high

operational quantum exposed cryptography, trusted long terms requirement. To analyse further the behaviour of the prioritization, 4 representative infrastructure scenarios as shown in Table 8 have been evaluated using a complete QRS methodology.

Table 8: Representative infrastructure scenarios.

Asset	V	E	Q	D	C	M	S	QRS	Priority
National PKI	5	5	5	5	5	2	5	95.00	Immediate
Banking VPN	4	5	5	4	5	3	3	80.75	Immediate
Healthcare archive	4	3	4	5	4	3	3	72.75	High
Legacy IoT fleet	3	3	3	2	3	4	2	48.75	Medium

The prioritization which has been formed aligned with existing PQC migration guidance which have been produced by standards organizations like NIST, NAS CNSA. Nonetheless, the addition of age, data longevity and sovereignty sensitivity will cause certain systems especially healthcare archive and trusted digital infrastructure to receive a higher PQC migration prioritization compared to if they were evaluated using technical evaluation only. The difference is important because conventional cyber security evaluation usually focus on the current operational ability and vulnerability only. Nonetheless, under the HNDL's assumption, systems which have long term sensitive secret data may have been exposed to quantum risks now even though the current categorization is 'medium' only.

Work example too shows how QRAF tries to balance the technical exposure, affected operation, migration readiness and governance scale in one prioritization framework. Even though the scenario presented here is only illustrative, it shows how the scoring model base on governance can support a more structured migration decision makers in a complex post-quantum environment.

6.3 Sensitivity Analysis

Sensitivity analysis is done by diversifying the weighting coefficient across different representatives. The objective is to see if medium changes in weighted coefficient will affect migration prioritization significantly. The result shows that critical systems, including National PKI infrastructure, defence environment and healthcare archive consistently remain in the high priority category across different weighted configurations which shows a stable resultant prioritization suitable to be used for governance-based planning. This analysis also shows that certain dimensions especially data longevity and sovereignty sensitivity can influence the position of migration greatly. The effect of this analysis is clearer in a system where conventional operational exposure only does not represent strategic long term quantum risks. This discovery shows that governance-based prioritization model may produce different migration outcomes compared to the traditional technical risk assessment especially if longterm secrecy environment, digital trust dependency or sovereign infrastructures are very important aspects to be taken into account in the process.

6.4 Governance Metrics and KPIs

QRAF introduces a set of measurable indicators with the intention to support migration monitoring, readiness tracking and governance reporting. The objective of these indicators are to move across qualitative evaluation only by establishing operational readiness matrix and observable migration maturity.

Rather than considering measurable PQC migration readiness as binary, QRAF approaches PQC migration as a continuous development process involving technical, operational and governance dimensions.

6.4.1 Governance KPIs

Governance based indicators give attention to visibility, operational readiness and migration alignment. The KPI will include:

i. Percentage of mission-critical systems covered by CBOM or SBOM inventories.

ii. Percentage of TLS endpoints supporting crypto-agility mechanisms.

iii. Percentage of trust anchors with defined rotation and replacement timelines.

iv. Percentage of cryptographic modules assessed as PQC-ready.

v. Percentage of high-priority assets assigned formal migration plans. These indicators enable organization to monitor PQC migration readiness from time to time while identifying areas where operational planning and cryptographic visibility may still not be sufficient.

6.4.2 Entropy Sovereignty Indicators

QRAF also introduces indicators related to Entropy Sovereignty. This matrix evaluates how far control on sovereignty remain on cryptographic infrastructure, digital trust management and migration governance. Sample of representatives include:

i. Percentage of root certification authorities hosted within sovereign infrastructure.

ii. Percentage of cryptographic modules sourced through trusted or validated supply chains.

iii. Percentage of national cryptographic policies governed by domestic regulatory authorities.

iv. Availability of domestic PQC certification or validation capabilities.

v. Percentage of critical trust infrastructure subject to sovereign operational oversight. These indicators are not meant to create technology isolation. On the other hand, they serve as practical ways to evaluate strategic exposure dependency and national security assurance in an interconnected trusted digital ecosystem.

6.4.3 ASEAN Collaboration Indicators

Considering the nature of cross-sector digital infrastructure, QRAF will also integrate related operational readiness and regional cooperation indicators with ASEAN migration planning. Examples include:

i. Number of regional PQC interoperability pilot programmes.

ii. Number of shared ASEAN PQC interoperability profiles.

iii. Percentage of cross-border services tested using hybrid deployment configurations.

iv. Participation in regional cyber exercises involving PQC transition scenarios.

v. Number of bilateral or multilateral trust-anchor coordination initiatives. The objective of Indicators is to increase regional visibility into operational readiness while supporting interconnected cross sector migration collaboration initiatives.

Table 9: Sample quantum risk register.

Ser.	Asset	Dependency	Exposure	Score	Priority	Mitigation
1	National PKI	RSA-2048	Critical	95	Immediate	Hybrid PQC migration
2	Banking VPN	ECC	High	84	Immediate	Crypto-agility upgrade
3	Healthcare archive	RSA	High	82	Immediate	Long-term encryption transition
4	Legacy IoT devices	ECC	Medium	56	Medium	Device lifecycle replacement

6.5 Sample Quantum Risk Register

Table 9 shows how quantum risk register which represents QRAF can be used operationally across different infrastructure categories.

This register shows how technical exposure, operational requirements and migration readiness can be integrated into one prioritization framework. In operational settings, additional field ownership, dependency mapping, operational status, financial budget and recovery timeline can also be included.

6.6 Migration Prioritization

QRAF prioritizes migration activities using combination of factors which include impact to the nation, operational dependencies, HNDL exposure, economic importance and interoperability risk. Even though migration prioritization may be different across jurisdiction, migration sequence may include:

i. Government systems.

ii. Defence infrastructure.

iii. Financial services.

iv. Telecommunications.

v. Healthcare systems.

vi. Cloud and digital infrastructure.

vii. Industrial and operational technology environments. The prioritization order cannot be interpreted as fixed universally. Different nation may adjust the migration prioritization to be suited with local threat environments, governance requirements, infrastructure dependencies and national security considerations. This framework also recognizes that migration alignment always subjected to practical operational limit. In some environments, legacy system may not support the immediate usage of PQC because of the limitation of the hardware, vendor dependency or interoperability concerns. In that case, organizations can depend temporarily on compensating controls, hybrid usage strategies or accelerate infrastructure replacement program until migration capabilities reached recommended maturity level threshold.

7 Governance, Readiness and Crypto-Agility

7.1 Crypto-Agility

Crypto-agility is the capability of a system to replace, update or reconfigure cryptographic algorithms without the need to modify the architecture. Crypto-agility is recognized as operational requirements and not as technical improvements. Its usage has been very important as a prerequisite for PQC migration [16][5][4]. For best practice, this requirement evolves from the fact that most of today's infrastructures are dependent on cryptographic assumptions for long term continuity. The main issue is that it is not just about finding alternative algorithms, but it must be able to replace it and continue operation without any disruptions. As a result, QRAF assumes crypto-agility as a basic sustainable PQC migration principle. The main concept includes modular cryptographic architecture, abstraction layers for selection of algorithms, automated trust management, cryptographic configuration based on policy and continuous operability tests. Organization with advanced cryptographic-agility ability tend to have higher resilient not just against quantum computer threats but also against other cryptographic vulnerabilities, governance changes and interoperability requirements.

7.2 Hybrid Cryptography Strategy

Since the transition to PQC may take years to complete, it is worthwhile to forecast that hybrid cryptographic strategy plays an important role at least in the beginning of the migration plan [5][6][7]. Hybrid cryptography strategy involves the usage of the combination of classical cryptography algorithms and PQC algorithms to sustain interoperability while reducing the related risks before using new approach. This is important because most operational ecosystems, like browsers, embedded devices, cloud services and PKI environment, are still dependent to mixed cryptographic solutions for quite some time in the future. As a result, it is not viable to replace trusted infrastructures immediately if the action may inadvertently disrupt services. In line with guidelines still under development by standards community such as NIST, ETSI and IETF, QRAF provides support for phased migration profile and not total replacement. Some of the hybrid strategies may include:

i. TLS 1.3 hybrid key establishment using ML-KEM alongside classical ECC mechanisms.

ii. Hybrid PKI trust-anchor migration combining classical and PQC signatures.

iii. Hybrid code-signing deployments using ML-DSA together with classical digitalsignature schemes.

iv. Long-term archival signature environments using SLH-DSA for high-assurance use cases. While supporting the changing standards, the usage of hybrid strategy ensures the continuity of operations during the transition period. At the same time, hybrid cryptography has its own complexity. Mutual support for different cryptographic approaches can complicate further interoperability, increase operational cost and require additional configuration management overhead. This is why QRAF considers hybrid migration as a temporary resilient solution and not as a final architectural approach.

7.3 Trust Anchor Rotation

One of the most sensitive aspects of PQC migration from operational point of view is the trust anchor rotation. The Root Certificate Authority, Intermediate Certificate Authority

and identity infrastructure which are well established form the core of nation's digital ecosystem. As a result, any changes to the Trust Anchor Rotation will have a greater impact than a simple certificate replacement. In QRAF context, Trust Anchor Rotation can be seen as an operational process which includes the following phases:

i. Assessment.

ii. Preparation.

iii. Distribution.

iv. Activation.

v. Retirement. Uncoordinated Trust Anchor Rotation may cause service disruptions, authentication failure, PKI failure and international scale interoperability issues. The effect of operational disruptions is very severe in the industrial environment which very much dependent on digital identity services like financial services, government services and telecommunications. Looking at this situation, QRAF considers Trust Anchor Rotation management not only as a technical issue, but it is also a governance issue. Migration requires coordination across various stakeholders, from infrastructure caretakers to software vendors and cross border trust ecosystems.

7.4 Entropy Sovereignty

This paper proposes a new concept which is known as Entropy Sovereignty, which can be viewed as development of governance to evaluate entropy sovereignty on cryptographic infrastructure and trust management. In principle, entropy sovereignty enlightens nation's capability to retain meaningful control over cryptographic key generation, trust anchor, migration governance, digital certificate process and in general, cryptographic infrastructure.

This concept shows paradigm shift in cyber security school of thoughts where cryptographic dependencies now have been understood not only from technical apprehension angle, but also from strategic and geopolitical angles. Some of the operational manifestation on Entropy Sovereignty may include:

i. Sovereign hosting of root certification authorities.

ii. Domestic governance of trust-anchor infrastructure.

iii. Trusted or nationally validated HSM supply chains.

iv. National PQC certification and validation programmes.

v. Regulatory authority over approved cryptographic standards and migration policies. This framework does not assume that technology viability is possible or necessarily desirable. Even though strong sovereignty controls will increase assurance and strategic autonomy, it comes with trade-offs in terms of interoperability and procurement flexibility, among others. On the other hand, heavy reliant on trusted infrastructure controlled by foreigners will increase interoperability and reduce operational cost, but it also adds strategic dependency element. As such, QRAF considers Entropy Sovereignty as a measurable governance dimension. Potential evaluation approaches may include:

vi. supply-chain dependency scoring,

vii. trust-anchor governance analysis,

viii. interoperability-cost evaluation,

ix. migration-governance maturity assessment,

x. sovereign certification capability measurement. By integrating entropy sovereignty consideration directly into migration prioritization, this framework intends to close strategic technical cyber planning with nation's sovereign resilience and wider strategic governance objectives.

8 ASEAN Interoperability and Cross-Border Collaboration

8.1 Cross-Border Challenges

Migration process to quantum resistant cryptography creates several cross-border challenges beyond operational technical issues. Since digital economy has become more interdependent, the decision on PQC migration which is produced by stakeholder may affect the operational capability, trust and continuity in other places. As such, international alignment issues, certificate alignment and interoperability planning have been highlighted in the documents produced by ETSI, ENISA dan other cybersecurity authorities [5][18][6][4]. One of the national risk challenges is that vendor and infrastructure developer may accept PQC standards, migration plan or different migration plan. Differences in approaches may lead to a more complex cross border validation, digital commerce and trust relationship between governments and industries which depend on share PKI infrastructure and digital identity which can be handled by the infrastructure. One more complicated factor is a supply chain exposure. Hardware, software in a cloud computing environment always filled with vague embedded cryptographic to the risk owners and stakeholders. In some of the cases, organizations may depend on third party infrastructure where the PQC migration plan is not known or totally externally controlled. Legacy system adds another layer of complexity. Some environment may have embedded technology or operation which does not allow PQC installation directly because of hardware limitation, firmware restriction or vendor support termination. In that situation, QRAF recommends layered approach to mitigate network segmentation, cryptographic gateway, accelerated replacement scheme and balancing operational control. As a whole, this consideration shows that coordinated effort across technical, operational and governance domains are required to have an effective PQC migration.

8.2 ASEAN Collaboration Opportunities

ASEAN plays a very important environment for PQC readiness because of increasing digital interoperability in this region, cross border services and growing cyber security issues.

Numerous ASEAN member countries have explored quantum-safe cryptography, cyber resilience and trusted digital governance. Collaboration among the ASEAN nations can assist in reducing fragmentation problems through a well-planned interoperability, knowledge sharing and coordinated PQC migration. Some of potential collaborative areas are as follows:

i. shared interoperability testing initiatives,

ii. regional cyber exercises involving PQC transition scenarios,

iii. joint certification and validation programmes,

iv. knowledge-sharing platforms,

v. coordinated workforce development initiatives,

vi. collaborative migration pilot projects. By engaging in collaboration among member nations, ASEAN may be able to reduce the readiness levels between countries and eventually will strengthen regional resilience. QRAF does not assume all nations that take part need to follow the same PQC migration architecture or governance. Emphasize is given more towards achieving basic capability which will enable cross border trust relationships but retain each nation's autonomy.

8.3 ASEAN PQC Readiness Index

This paper suggests ASEAN PQC readiness index as regional measurement which has been designed to assist in collaborative readiness assessment and migration planning. The proposed index operationalizes entropy sovereignty elements across various jurisdictions without releasing controls over migration policy and trusted infrastructure. Instead of being compliance score card, the index aims to function as a tool to assess regional compliance which measure capability gaps, interoperability problems and collaboration prospects. Some of the sample indicators may include:

i. Percentage of critical systems with completed cryptographic inventories.

ii. Percentage of systems assigned QRAF-based migration priorities.

iii. Percentage of cross-border services tested using PQC interoperability profiles.

iv. Percentage of national trust anchors with defined migration roadmaps.

v. Number of completed regional PQC interoperability pilot initiatives. ASEAN PQC readiness index can function as a benchmark among ASEAN member nations and encourage coordinated readiness planning without mandating standardize national architecture.

8.4 ASEAN Interoperability Profile

ASEAN interoperability profile which can be operationalized can be initiated by focusing on the number of cryptographic primitives and limited use cases rather than imposing abrupt comprehensive regional standardization. For example:

i. ML-KEM may be mapped to hybrid TLS 1.3 key-establishment profiles.

ii. ML-DSA may support digital signatures and authentication workflows.

iii. SLH-DSA may be considered for long-term archival assurance and high-assurance signature environments. The aim here is not to impose regional cryptographic architecture. On the other hand, the aim is to define common baselines which can be used to support cross-border digital services, government communications and critical infrastructure coordination. From time to time, this readiness profile may evolve when operational experience has matured, standards have stabilized and regional requirements have become clearer.

9 Strategic Recommendations

PQC migration needs to be considered as a continuous program which focuses on resilience and not just as technology update activity. Based on QRAF model, there are a few strategic proposals which can be considered by the governments, stakeholders and critical infrastructure owners.

i. Establish national PQC governance programs. Government shall develop coordinated governance which defines migration responsibilities, nation prioritization and monitoring mechanism. An effective migration planning may require collaboration across cyber security agencies, regulators, critical infrastructure owners and standards organizations.

ii. Conduct comprehensive cryptographic asset inventories. Organizations need to prioritize cryptographic discovery activities to enhance visibility into exposed dependency across infrastructures, application and supply chain. Incomplete inventory will remain as a big hindrance to effective migration planning.

iii. Develop national quantum risk registers. Structured risk register can assist organizations and governments to identify high priority systems, monitoring migration status and coordinate cross sector recovery activities.

iv. Prioritize migration for critical infrastructure. Migration planning shall focus on system where once compromised can affect severely operational capabilities, national economic or security. Special attention shall be given to environments exposed to long term HNDL risk.

v. Invest in Crypto-agility capabilities. Organizations shall modernize cryptographic architecture to support algorithmic replacement, hybrid deployment and future interoperability requirements. Systems with less crypto-agility capabilities may incur higher migration cost and operational disruption.

vi. Establish trusted certification ecosystems. National certification and validation can assist to increase trust assurance, supply chain transparency and operational consistency during the transition to PQC migration.

vii. Strengthening sovereign cryptographic capabilities Government shall evaluate strategic dependencies related to trust anchor infrastructure, cryptographic supply chain and external certification ecosystem. The objective is not to isolate the technology but to have better resilience and governance visibility.

viii. Strengthening interoperability and regional collaboration. Cross border interoperability planning will be more important when digital services and trust infrastructure interoperability across jurisdictions. Regional cooperation initiatives can assist to reduce segmentation and increase operational coordination.

ix. Develop PQC workforce and talent pipeline. The success of long-term migration depends on competent workforce in cryptography, governance, cyber security, PKI operations, interoperability test and critical infrastructure security.

x. Continuously monitor quantum and standards development. PQC standards, operational guidelines and quantum capabilities will always evolve. As a result, migration plan must be adaptive. Collectively, this recommendation will highlight the importance of PQC migration as a resilience and governance issue as much as technical problems.

10 Limitations

A number of limitations must be stated about our current research. First, the example of validation in this paper is an illustration only and data set is not from operational critical infrastructure. Even though the examples of the illustrations based on existing migration guidelines and cyber security governance best practices, it cannot be taken

as empirical evidence of real national infrastructure migration. Second, the weighting scheme considered in the QRS framework still has not been validated by a wide group of experts and still subjected to rigorous statistical validation. Various jurisdiction and sectors may use different weighing coefficients to take into account sovereignty sensitivity, criticality of operations or data life span based on their varying risk appetite. Third, interoperability mapping explained in this paper is also illustrative with the purpose to supplement the number of evolving NIST, ETSI and IETF community guidance. Finally, current governance-based framework emphasizes on the ease of understanding and practical interoperability compared to mathematical solution. Even though more complex

dependency modelling or probabilistic approach can probably bring the model to a higher degree of accuracy, this aspect is still out of our current research scope.

11 Conclusion and Future Work

Quantum computing raises long term threat to modern cryptographic frameworks and trusted digital infrastructure. Even though CRQC does not exist yet in current practices, the evolution of HNDL shows that PQC migration cannot be delayed indefinitely [11][17]. This paper has outlined QRAF based on governance with the aim to support existing PQC readiness, migration prioritization and cyber resilience planning. This framework is using cryptographic asset discovery, exposure mapping, risk scoring, governance alignment and operational resilience considerations in its assessment approaches. One of QRAF main contributions is the development of QRS which integrate technical exposure aspect, migration-ready, data lifespan and entropy sovereignty concern in one single matrix. This paper also suggests 'Entropy Sovereignty' as governance matrix to be used for sovereign cryptographic asset management, trust anchors and migration. Finally, this research proposes that PQC migration cannot be thought from algorithm replacement angle only. Migration involves governance related issues, cyber resilience, operational continuity, supply chain integrity and regional interoperability. Such that, future migration planning must be driven by continuous cross sector collaboration. Further research may include quantitative frameworks validations, expert weighted scoring criteria, industrial based migration simulations and using AI based risk analysis. It may also include interoperability test methodology development, sovereignty assurance matrix and ASEAN readiness matrix as a whole. QRS model also can be extended to include symmetric key exposure through the introduction of parameters like key length, key re-keying frequency and Grover threat assumptions.

References

[1] Y. Baseri, V. Chouhan, A. Ghorbani, and A. Chow. Evaluation framework for quantum security risk assessment: A comprehensive strategy for quantum-safe transition. *Computers & Security*, 149:104272, 2024.

[2] Cyber Security Agency of Singapore. Advisory on quantum-safe cryptography. Technical report, CSA Singapore, 2023.

[3] CyberSecurity Malaysia. Post-quantum cryptography migration framework. Technical report, CyberSecurity Malaysia, 2024.

[4] ENISA. Post-quantum cryptography: Current state and quantum mitigation. Technical report, European Union Agency for Cybersecurity, 2021.

[5] ETSI. Migration strategies and recommendations to quantum safe schemes. Technical Report ETSI TR 103 619 V1.1.1, European Telecommunications Standards Institute, 2020.

[6] ETSI. Quantum-safe cryptography and security. Technical Report White Paper 36, 2nd ed., European Telecommunications Standards Institute, 2020.

[7] ETSI. Quantum-safe cryptography toolkit. Technical Report ETSI GS QSC 014 V1.1.1, European Telecommunications Standards Institute, 2020.

[8] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 212–219, 1996.

[9] ISO/IEC. Security requirements for cryptographic modules, 2012.

[10] M. A. Khan and S. R. Mohanty. A graph-based framework for cryptographic risk modeling in the post-quantum era. *Journal of Computer Security*, 32(3):301–325, 2025.

[11] Michele Mosca. Cybersecurity in an era with quantum computers. *IEEE Security & Privacy*, 16(5):38–41, 2018.

[12] NIST. Security and privacy controls for information systems and organizations. Technical Report SP 800-53, National Institute of Standards and Technology, 2020.

[13] NIST. Module-lattice-based digital signature standard (ml-dsa). Technical Report FIPS PUB 204, National Institute of Standards and Technology, 2024.

[14] NIST. Module-lattice-based key-encapsulation mechanism standard (ml-kem). Technical Report FIPS PUB 203, National Institute of Standards and Technology, 2024.

[15] NIST. Stateless hash-based digital signature standard (slh-dsa). Technical Report FIPS PUB 205, National Institute of Standards and Technology, 2024.

[16] NSA. Commercial national security algorithm suite 2.0 (cnsa 2.0). Technical report, National Security Agency, 2022.

[17] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[18] UK NCSC. Next steps in preparing for post-quantum cryptography. Technical report, National Cyber Security Centre, 2023.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 413–426.

A Comparative Evaluation of Wiener's Continued Fraction and Boneh–Durfee Lattice-Based Attacks on RSA under Private Exponent MSB Leakage

Siti Nor Taqiyah Zuhari[1] and Khairun Nisak Muhammad[1,2]

[1] Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia {216930@student,nisak_muhammad}.upm.edu.my
[2] Centre for Foundation Studies in Science of Universiti Putra Malaysia, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia nisak_muhammad@upm.edu.my

Abstract. RSA security can be weakened when secret information, such as the private exponent is leaked. This paper studies the effect of most significant bit (MSB) leakage on two small private exponent attacks, which are the Wiener's continued fraction attack and the Boneh–Durfee lattice-based attack. RSA keys of 512, 1024 and 2048 bits were tested with different levels of MSB leakage. The results show that Wiener's attack was not successful, even with MSB leakage. However, the Boneh–Durfee attack became successful when enough MSBs were leaked, especially at about one quarter leakage for larger keys. This shows that lattice-based attacks are more effective than continued fraction attacks when part of the RSA private exponent is exposed.

Keywords: RSA · MSB leakage · Wiener's continued fraction attack · Boneh–Durfee lattice-based attack · Private exponent

1 Introduction

The Rivest–Shamir–Adleman (RSA) cryptosystem [1] is one of the most widely studied public-key cryptosystems and remains important in digital signatures, encryption and authentication. Its security is commonly associated with the computational difficulty of factoring a large composite modulus $N = pq$, where p and q are large prime numbers. In a correctly implemented RSA system, the private exponent d must remain secret because knowledge of d allows an adversary to decrypt ciphertexts or generate valid digital signatures.

Although RSA is mathematically secure when its parameters are properly selected, practical implementations may still be vulnerable to side-channel attacks. These attacks exploit information leaked through timing behaviour, power consumption, electromagnetic emission, memory exposure or fault behaviour. One possible consequence of such leakage is partial exposure of the private exponent. In particular, exposure of the most significant bits (MSBs) of d may reduce the uncertainty of the secret exponent and assist cryptanalytic recovery.

Several attacks have been proposed against RSA when the private exponent is weak or partially exposed. Wiener's attack is a classical continued fraction attack that can recover d when the private exponent is sufficiently small [2]. Boneh and Durfee later introduced a stronger lattice-based attack that extends the vulnerable range of small private exponents [3]. In partial key exposure settings, lattice-based approaches are especially relevant because known bits of secret parameters can be incorporated into polynomial equations

and solved using small-root techniques [4].

Contribution.
This paper evaluates the impact of the private exponent MSB leakage on Wiener's continued fraction attack and the Boneh–Durfee lattice-based attack. RSA key sizes of 512, 1024 and 2048 bits are considered. The attacks are compared on the basis of success rate and effectiveness under partial key exposure. The main contribution of this paper is an experimental comparison of continued fraction and lattice-based private exponent attacks under controlled MSB leakage conditions.

Paper Organisation.
The remainder of this paper is organised as follows. Section 2 presents the background of RSA, private exponent leakage, Wiener's continued fraction attack and Boneh–Durfee's lattice-based attack. Section 3 discusses related work on small private exponent attacks, lattice-based RSA cryptanalysis and partial key exposure attacks. Section 4 describes the experimental setup, including RSA key generation, MSB leakage simulation and attack implementation. Section 5 presents the experimental results and discussion. Finally, Section 6 concludes the paper and outlines possible future work.

2 Mathematical Preliminaries

2.1 RSA Cryptosystem

In RSA, key generation begins by selecting two large prime numbers p and q and multiplied to form the modulus

$$N = pq. \tag{1}$$

The Euler totient value is given by

$$\phi(N) = (p-1)(q-1). \tag{2}$$

A public exponent e is chosen such that $\gcd(e, \phi(N)) = 1$. The private exponent d is then computed as the modular inverse of e modulo $\phi(N)$, satisfying

$$ed \equiv 1 (\mathrm{mod} \phi(N)). \tag{3}$$

The public key consists of (N, e), while the private key includes d and, in most implementations, the prime factors p and q. To encrypt a message m, the sender uses the public key (N, e). The ciphertext c is computed as

$$c \equiv m^e (\mathrm{mod} N). \tag{4}$$

Decryption is performed using the private exponent d. Given a ciphertext c, the original message m is recovered by computing

$$m \equiv c^d (\mathrm{mod} N). \tag{5}$$

The security of RSA depends on the assumed hardness of factoring the modulus N, which prevents an attacker from computing the private exponent d from the public parameters e and N. In standard implementations, the private exponent d must be chosen sufficiently large to prevent known small private exponent attacks, including the lattice-based attack introduced by Boneh and Durfee [3].

2.2 Private Exponent MSB Leakage

Private exponent leakage occurs when an adversary obtains partial information about d. In practice, MSB leakage may occur through side-channel attacks such as timing analysis, power analysis, electromagnetic emanations, or faulty hardware behavior as demonstrated by Kocher [18]. More recent work by Lü et al. further confirms that real-world devices may unintentionally leak high or low-order bits through timing, electromagnetic, or physical side-channel analysis [16]. Heninger and Shacham demonstrated that implementation flaws and side-channel effects can leak partial key information at scale [17], while Genkin, Shamir, and Tromer showed that side-channel leakage can reveal partial RSA secrets in practical settings [15].

If the private exponent d has m bits and the first k MSBs are known, it can be expressed as

$$d = 2^{m-k} \cdot d_{\text{high}} + d_{\text{low}},$$

where d_{low} remains unknown. When part of the private exponent is leaked, the attacker's search space becomes smaller. This weakens RSA security and increases the possibility of attacks that are normally impractical.

2.3 Wiener's Continued Fraction Attack

Wiener's continued fraction attack is a classical cryptanalytic method that targets RSA instances with an unusually small private exponent. The attack exploits the relationship between the public exponent e, the modulus N, and the private exponent d. It is based on the RSA key equation

$$ed - k\phi(N) = 1, \tag{6}$$

where k is an unknown integer. From this relation, the fraction e/N can provide a good rational approximation to k/d when d is sufficiently small.

A continued fraction expansion of a rational number can be written as

Definition 1 (Continued Fraction)**.** [6] The continued fraction expansion of a rational number $\frac{p}{q}$ is given by

$$\frac{p}{q} = a_0 + \cfrac{1}{a_1 + \cfrac{1}{\ddots + \cfrac{1}{a_n}}}$$

where $a_0 \in \mathbb{Z}$ and $a_1, \ldots, a_n \in \mathbb{Z}^+$. The convergents of this continued fraction are the rational numbers

$$\frac{p_i}{q_i} = a_0 + \cfrac{1}{a_1 + \cfrac{1}{\ddots + \cfrac{1}{a_i}}}, \qquad i = 0, 1, \ldots, n.$$

Each convergent $\frac{p_i}{q_i}$ provides the best rational approximation to $\frac{p}{q}$ among all rational numbers with a denominator at most q_i.

In Wiener's attack, the convergents of e/N are tested as possible candidates for k/d. The denominator of a valid convergent may reveal the private exponent d.

Theorem 1 (Wiener)**.** *[2] Let $N = pq$ be an RSA modulus with balanced primes, and let d be the private exponent. If*

$$d < \frac{1}{3} N^{1/4}, \tag{7}$$

then d can be recovered in polynomial time using the continued fraction expansion of e/N.

This theorem shows that RSA becomes insecure when the private exponent is too small. In this study, Wiener's attack is used as the representative continued fraction-based attack.

Algorithm Summary. The general procedure of Wiener's continued fraction attack is summarised as follows:

Algorithm 1 Wiener's Continued Fraction Attack

Input: RSA public key (N, e)
Output: Private exponent d (if recovered)
1: Compute the continued fraction expansion of e/N
2: Generate convergents (k_i/d_i)
3: **for** each convergent (k_i/d_i) **do**
4: Treat d_i as a candidate private exponent
5: Compute a candidate value of $\varphi(N)$ using

$$\varphi(N) \leftarrow \frac{ed_i - 1}{k_i}$$

6: **if** $\varphi(N)$ is an integer **then**
7: Compute $S \leftarrow N - \varphi(N) + 1$
8: Solve the quadratic equation $x^2 - Sx + N = 0$
9: **if** the equation has integer roots p, q **then**
10: **return** $d \leftarrow d_i$
11: **end if**
12: **end if**
13: **end for**
14: **return** failure

Wiener's attack is efficient and runs in polynomial time, but it is only effective when the RSA private exponent satisfies a strict smallness condition.

2.4 Boneh–Durfee Lattice-Based Attack

The Boneh–Durfee attack is a lattice-based attack that improves upon Wiener's continued fraction approach. While Wiener's attack is limited to very small private exponents, Boneh and Durfee showed that RSA can also be vulnerable for a wider range of private exponent sizes using lattice reduction techniques [3]. The attack is based on the RSA key equation

$$ed - k\phi(N) = 1, \tag{8}$$

where (k) is an unknown integer.

By expressing Euler's totient function as

$$\phi(N) = N - (p + q) + 1, \tag{9}$$

the RSA key equation can be transformed into a modular polynomial involving small unknown values. These small values can then be recovered using lattice reduction techniques, particularly the LLL algorithm.

Theorem 2 (Boneh–Durfee)**.** *[3] Let (N = pq) be an RSA modulus with balanced primes. Under certain heuristic assumptions, if the private exponent satisfies*

$$d < N^{0.292}, \tag{10}$$

then the RSA private exponent d can be recovered in polynomial time using lattice-based techniques.

This theorem shows that the Boneh–Durfee attack extends the vulnerable range of RSA private exponents beyond the classical Wiener bound of approximately ($N^{1/4}$). In the context of private exponent MSB leakage, the known most significant bits reduce the size of the unknown part of d, making lattice-based recovery more practical.

Algorithm Summary. The general procedure of the Boneh–Durfee lattice-based attack is summarised as follows:

Algorithm 2 Boneh-Durfee Lattice Attack

Input: RSA public key (N, e)
Output: Private exponent d (if recovered)
1: Construct the modular polynomial $f(x, y)$ derived from the RSA equation
2: Generate shifted multiples of $f(x, y)$ to encode small-root conditions
3: Embed the shifted polynomials into a lattice basis
4: Apply the LLL algorithm to reduce the lattice basis
5: Extract the small integer roots (x, y) from the reduced basis
6: Recover p and q from $s = p + q$
7: Compute $\varphi(N) \leftarrow (p - 1)(q - 1)$
8: Compute $d \leftarrow e^{-1} \bmod \varphi(N)$
9: **return** d

Compared with Wiener's attack, the Boneh–Durfee attack is more powerful under partial key exposure conditions. However, it is also more computationally demanding and depends strongly on lattice parameters and the amount of available MSB leakage.

3 Related Work

The RSA cryptosystem was introduced by Rivest, Shamir and Adleman as a public-key cryptosystem based on the difficulty of factoring a large composite modulus [1]. In RSA, the public key consists of the modulus N and public exponent e, while the private exponent d must remain secret. Once d is recovered, the RSA system can no longer be considered secure. Therefore, RSA security depends not only on the difficulty of factoring N, but also on the correct selection and protection of its private parameters.

One important vulnerability in RSA occurs when the private exponent is chosen too small. Wiener introduced an efficient continued fraction attack that can recover the private exponent when $d < N^{0.25}$ [2]. This attack uses the continued fraction expansion of e/N to identify possible values of d. Later, Verheul and van Tilborg extended Wiener's approach by combining continued fractions with additional search techniques [7], while Dujella further refined the analysis of continued fraction-based attacks [8]. However, these methods mainly rely on the smallness of d and do not naturally exploit leaked most significant bits (MSBs) of the private exponent.

A more powerful direction in RSA cryptanalysis is based on lattice reduction. The Lenstra-Lenstra-Lovasz (LLL) algorithm introduced a polynomial-time method for lattice basis reduction [5], and Coppersmith later showed that lattice techniques can be used to find small roots of polynomial equations [9]. Building on these methods, Boneh and Durfee developed a lattice-based attack that can recover RSA private exponents satisfying approximately $d < N^{0.292}$ [3]. This result improves the classical Wiener bound and shows that lattice-based methods can attack a wider range of weak RSA private exponents.

More recently, X. Feng, Nitaj, and Pan demonstrated that small private exponent attacks become significantly more effective when auxiliary information, such as leaked MSBs of d, is available [12]. Their results indicate that the classical Wiener bound is no

longer a reliable security threshold in the presence of partial leakage. Q. Feng, Liu, Nitaj, and Pan revisited small private exponent attacks under realistic leakage conditions and demonstrated that limited MSB exposure can substantially reduce attack complexity [13]. Jochemsz and May revisited multivariate Coppersmith techniques and showed improved bounds under structured assumptions [14].

Partial key exposure attacks further show that RSA may be compromised even when only part of the private key is leaked. Boneh, Durfee, and Frankel showed that knowing a small fraction of the private key bits can be sufficient for key recovery under certain conditions [4]. Later, Blömer and May studied new partial key exposure attacks on RSA using lattice-based techniques [10], while Ernst et al. extended the analysis to cases involving most significant bits and least significant bits of the private exponent [11].

In practical side-channel settings, Genkin, Shamir, and Tromer showed that side-channel leakage can reveal partial RSA secrets in practical settings [15], while Lü et al. analyzed partial exponent leakage in embedded devices and confirmed that MSB leakage significantly weakens RSA security [16]. Heninger and Shacham demonstrated that implementation flaws and side-channel effects can leak partial key information at scale [17]. These scenarios motivate the investigation of private exponent attacks under MSB leakage conditions.

However, limited work has directly compared Wiener's continued fraction attack and Boneh-Durfee's lattice-based attack under the same private exponent MSB leakage conditions. Therefore, this paper evaluates both attacks using identical RSA key sizes and leakage levels to determine which approach remains effective under partial key exposure.

4 Experimental Setup

This study uses an experimental simulation approach. Experiments were run on Ubuntu 22.04 LTS (WSL2) on a 12th Gen Intel(R) Core(TM) i5-12500H 8 GB RAM with SageMath 10.7. RSA key pairs were generated for three modulus sizes: 512, 1024 and 2048 bits. For each key size, MSB leakage was simulated at increasing levels. Wiener's attack and the Boneh–Durfee attack were then applied under identical experimental conditions.

4.1 Experimental Design

The experiment was conducted in six stages. First, RSA key pairs were generated for the selected key sizes. Second, the private exponent d was extracted. Third, MSB leakage was simulated by revealing a fixed proportion of the leading bits of d. Fourth, Wiener's attack was applied to each instance. Fifth, the Boneh–Durfee lattice attack was applied using the same key instances and leakage settings. Finally, the results were compared based on success rate and minimum leakage threshold.
The main experimental variables are summarised in Table 1.

Table 1: Experimental variables used in the comparative evaluation.

Variable	Description
RSA key size	512, 1024 and 2048 bits
Leakage type	Most significant bits of private exponent d
Leakage level	Increasing percentage of exposed MSBs
Attack methods	Wiener's attack and Boneh–Durfee attack
Evaluation metrics	Success rate, minimum leakage and effectiveness

For each configuration, 100 independent RSA key pairs were generated and tested to ensure statistical reliability.

4.2 RSA Key Generation

RSA key pairs are generated using standard RSA key generation procedures. Two distinct primes p and q are generated, and the modulus is computed as

$$N = pq. \tag{11}$$

Euler's totient function is then calculated as

$$\phi(N) = (p-1)(q-1). \tag{12}$$

A public exponent (e) is chosen such that

$$\gcd(e, \phi(N)) = 1. \tag{13}$$

The private exponent (d) is computed as the modular inverse of e modulo ($\phi(N)$), satisfying

$$ed \equiv 1 \pmod{\phi(N)}. \tag{14}$$

The key generation process is repeated until the private exponent satisfies

$$N^{0.25} < d < N^{0.30}. \tag{15}$$

Algorithm 3 RSA Key Generation

Input: Key size n (bits), public exponent e
Output: RSA public key (N, e) and private key d

Generate two distinct primes p, q of size $n/2$ bits
$N \leftarrow p \times q$
$\varphi(N) \leftarrow (p-1)(q-1)$
Choose e such that $\gcd(e, \varphi(N)) = 1$
$d \leftarrow e^{-1} \bmod \varphi(N)$
repeat
    Generate two distinct primes p, q of size $n/2$ bits
    $N \leftarrow p \times q$
    $\varphi(N) \leftarrow (p-1)(q-1)$
    Choose e such that $\gcd(e, \varphi(N)) = 1$
    $d \leftarrow e^{-1} \bmod \varphi(N)$
until $N^{0.25} < d < N^{0.30}$
return (N, e, d)

4.3 MSB Leakage Simulation

MSB leakage is simulated by exposing a fixed number of the most significant bits of the RSA private exponent d. Let d be an m-bit integer and let k denote the number of leaked MSBs. The private exponent can be decomposed as

$$d = 2^{m-k} d_{\text{high}} + d_{\text{low}}, \tag{16}$$

where d_{high} represents the known MSB part and d_{low} represents the unknown remaining part. These values are computed as

$$d_{\text{high}} = \left\lfloor \frac{d}{2^{m-k}} \right\rfloor, \tag{17}$$

and

$$d_{\text{low}} = d \bmod 2^{m-k}. \tag{18}$$

The unknown part satisfies

$$0 \leq d_{\text{low}} < 2^{m-k}. \tag{19}$$

In the experiments, leakage levels of 0%, 10%, 20%, 30%, 40% and 50% of the private exponent bit length are evaluated. Increasing the leakage level reduces the unknown search space and may improve the success probability of key recovery attacks.

A limitation of this study is the assumption of perfect MSB leakage without errors. Only deliberate MSB leakage is considered, and no real-world side-channel sources or noise models are included. In practice, side-channel leakage may be noisy or partial.

Algorithm 4 MSB Leakage Simulation

Input: Private exponent d, bit length m, leakage size k
Output: Known MSB d_{high}, unknown part d_{low}

$d_{\text{high}} \leftarrow \left\lfloor \frac{d}{2^{m-k}} \right\rfloor$
$d_{\text{low}} \leftarrow d \bmod 2^{m-k}$
return $(d_{\text{high}}, d_{\text{low}})$

4.4 Attack Implementation

Both Wiener's attack and the Boneh–Durfee attack are implemented under the same RSA key sizes and leakage levels. For each trial, the attack result is recorded as either successful if the private exponent d is recovered, or unsuccessful if no valid private exponent is found.

For Wiener's attack, the continued fraction expansion of e/N is computed. The convergents are then generated and tested as candidates for recovering the private exponent. The leaked MSBs are used to form candidate values of d, and each candidate is verified using the RSA key relation.

Algorithm 5 Wiener's Continued Fraction Attack with MSB Leakage

Input: Public key (N, e), leaked MSBs d_{high}, leakage size k
Output: Private exponent d or failure

Compute continued fraction expansion of e/N
Generate convergents (k_i/d_i)
for each convergent (k_i/d_i) **do**
    Form candidate d using d_i and d_{high}
    if $ed \equiv 1 \bmod \varphi(N)$ **then return** d
    end if
end for
return failure

For the Boneh-Durfee attack, the leaked MSBs are incorporated into the lattice-based recovery process. The RSA key equation is transformed into a modular polynomial, and a lattice basis is constructed using suitable bounds. The LLL algorithm is then applied to reduce the lattice and extract small roots. In this experiment, the lattice dimension is fixed at 32. A lattice dimension of 32 is chosen as a practical compromise between attack success and computational cost [12]. has shown that the effectiveness and runtime of lattice-based attacks are highly sensitive to the choice of lattice dimension, and that moderate dimensions are commonly adopted in practice to balance efficiency and performance. The

lattice dimension is kept constant across the attack execution to ensure fair and consistent comparison.

Algorithm 6 Boneh-Durfee Attack with MSB Leakage

Input: Public key (N, e), leaked MSBs of d, bounds X, Y
Output: Private exponent d or failure
1: Form RSA polynomial using leaked MSBs obtained from Algorithm 3
2: Construct modular polynomial $f(x, y)$
3: Build lattice using bounds X, Y
4: Apply LLL lattice reduction
5: Extract small roots (x, y)
6: Recover candidate private exponent d
7: **if** RSA decryption succeeds **then**
8: **return** d
9: **end if**
10: **return** failure

5 Attack Effectiveness and Comparative Analysis

The experimental results show a clear difference between Wiener's attack and the Boneh–Durfee attack under private exponent MSB leakage conditions.

5.1 Validation

A numerical example is presented to validate the correctness of the experimental procedures. By using very small RSA parameters, all computations can be verified manually. The purpose of this example is to validate the correctness of the RSA key generation, the MSB leakage simulation and the recovery procedures under both Wiener's and Boneh-Durfee attacks.

To confirm that the implementations are correct, both attacks were first tested on RSA instances where the private exponent lies within their theoretical vulnerable ranges. Wiener's attack successfully recovered d when $d < N^{1/4}$, and the Boneh-Durfee attack succeeded when $d < N^{0.292}$. These validation experiments confirm that both attacks are implemented correctly. Overall, this example verifies that the RSA parameters, MSB leakage simulation and success checking logic used in the experiment are implemented correctly.

5.2 Wiener's Attack Performance

The experimental results show that Wiener's attack failed to recover the RSA private exponent for all tested key sizes and MSB leakage levels. This result is consistent with the selected private exponent range,

$$N^{0.25} < d < N^{0.30}. \tag{20}$$

Wiener's attack is expected to succeed only when the private exponent is sufficiently small, commonly bounded by

$$d < \frac{1}{3} N^{1/4}. \tag{21}$$

Since the generated private exponents are larger than the classical Wiener vulnerable range, the correct value of d is unlikely to appear as a valid denominator among the convergents of the continued fraction expansion of e/N.

Illustrative Example. For a 1024-bit RSA modulus, the Wiener bound corresponds approximately to

$$\log_2(N^{1/4}) = \frac{1024}{4} = 256. \tag{22}$$

This means that Wiener's attack is expected to be effective only when the private exponent is less than approximately 256 bits. However, if the private exponent is generated near $N^{0.26}$, then its approximate bit length is

$$\log_2(N^{0.26}) = 0.26(1024) \approx 266. \tag{23}$$

Thus, the private exponent lies outside Wiener's vulnerable range.

If 25% of the most significant bits of d are leaked, then the number of leaked bits is approximately

$$k = 0.25(266) \approx 67. \tag{24}$$

The private exponent can then be written as

$$d = 2^{266-67} d_{\text{high}} + d_{\text{low}}, \tag{25}$$

where d_{high} is known and d_{low} remains unknown. Although this leakage reduces the number of unknown bits in d, it does not change the continued fraction expansion of e/N. Therefore, the leaked MSBs do not make Wiener's attack successful when d is already larger than its theoretical bound.

Overall, Wiener's attack remains ineffective under the tested MSB leakage conditions. This indicates that MSB leakage alone does not significantly improve the continued fraction-based recovery of d when the private exponent is outside Wiener's success range.

5.3 Boneh–Durfee Attack Performance

The experimental results show that the Boneh–Durfee attack becomes increasingly effective as the MSB leakage level increases. Unlike Wiener's attack, which depends mainly on the private exponent being sufficiently small, the Boneh–Durfee attack can benefit from partial key exposure because leaked MSBs reduce the size of the unknown part of d. This makes the remaining unknown value more suitable for lattice-based small-root recovery.

In this study, the private exponent is represented as

$$d = 2^{m-k} d_{\text{high}} + d_{\text{low}}, \tag{26}$$

where d_{high} is the known MSB part and d_{low} is the unknown remaining part. The unknown component satisfies

$$0 \leq d_{\text{low}} < 2^{m-k}. \tag{27}$$

As the leakage size k increases, the bound on d_{low} becomes smaller. Therefore, the search space for the unknown part of the private exponent decreases from approximately 2^m to 2^{m-k}.

The results indicate that the Boneh–Durfee attack shows a clear transition from low success to high success as the leakage level increases. For 1024-bit and 2048-bit RSA keys, the attack becomes consistently successful when approximately one quarter of the private exponent bits are leaked. For 512-bit RSA, the transition occurs later, requiring a larger leakage percentage before complete recovery is achieved.

Illustrative Example. For a 1024-bit RSA modulus, suppose the private exponent is generated near $N^{0.26}$. The approximate bit length of d is

$$\log_2(N^{0.26}) = 0.26(1024) \approx 266. \tag{28}$$

If 25% of the most significant bits are leaked, then the number of known bits is approximately

$$k = 0.25(266) \approx 67. \tag{29}$$

Hence, the number of unknown bits is approximately

$$m - k = 266 - 67 = 199. \tag{30}$$

Thus, instead of recovering the full 266-bit private exponent, the attack only needs to recover the unknown component d_{low}, which is bounded by

$$0 \leq d_{\text{low}} < 2^{199}. \tag{31}$$

This reduction in the unknown search space explains why the Boneh–Durfee attack becomes more effective under MSB leakage. The leaked MSBs provide structural information that can be incorporated into the lattice-based recovery process. As a result, the attack becomes more successful once enough MSBs are exposed.

Overall, the Boneh–Durfee attack demonstrates stronger performance than Wiener's attack under partial key exposure. The results show that lattice-based methods are more suitable for exploiting MSB leakage because they can use the known part of d to reduce the remaining recovery problem.

5.4 Comparative Analysis

The general behaviour of both attacks is summarised in Table 2. Wiener's attack is efficient but structurally limited, while the Boneh–Durfee attack is more adaptable in partial key exposure settings.

Table 2: General comparison of Wiener's and Boneh–Durfee attacks under MSB leakage.

Attack	Main technique	Effect of MSB leakage	Overall result
Wiener's attack	Continued fractions	No significant improvement	Ineffective under tested leakage conditions
Boneh–Durfee attack	Lattice reduction	Success improves as leakage increases	Effective after sufficient MSB leakage

Table 3 presents the success rates of Wiener's and Boneh–Durfee attacks under different MSB leakage levels. The comparison is made across RSA key sizes of 512, 1024 and 2048 bits.

Table 3: Experimental success rates of Wiener's and Boneh–Durfee attacks under different MSB leakage levels. Recovery time is the average CPU time (in seconds) required to recover the remaining private exponent bits after k MSBs are exposed. Wiener's attack achieved 0% success across all configurations and is omitted.

RSA Key Size	MSB Leakage Level	Leaked Bits (k)	Boneh–Durfee Success Rate	Recovery Time (s)
512-bit	5%	26	0%	0.029
512-bit	10%	51	20%	0.004
512-bit	15%	77	40%	0.004
512-bit	20%	102	70%	0.003
512-bit	25%	128	90%	0.004
1024-bit	5%	51	0%	0.003
1024-bit	10%	102	40%	0.012
1024-bit	15%	154	60%	0.003
1024-bit	20%	205	75%	0.003
1024-bit	25%	256	95%	0.003
2048-bit	5%	102	0%	0.004
2048-bit	10%	205	40%	0.026
2048-bit	15%	307	65%	0.003
2048-bit	20%	410	85%	0.004
2048-bit	25%	512	95%	0.003

The results show that Wiener's attack records a 0% success rate for all RSA key sizes and leakage levels. This indicates that MSB leakage does not improve Wiener's attack when the private exponent is outside its theoretical vulnerable range. Wiener's attack depends on d being very small, not on knowing some of its bits. In this study, the private exponent d is generated within the range $N^{0.25} < d < N^{0.30}$. This interval is intentionally chosen to exclude the regime where Wiener's attack is theoretically guaranteed to succeed ($d < N^{0.25}$). Since the private exponents were generated beyond Wiener's bound ($d > N^{1/4}$), leaking MSBs cannot make the attack succeed.

Wiener's attack remains unsuccessful because it is fundamentally a magnitude-based attack, not an information-based attack. The continued fraction approximation of e/N produces convergents that converge to k/d. However, the quality of this approximation depends on d being sufficiently small relative to N. MSB leakage reduces the entropy of d but does not change its magnitude or the convergence rate of the continued fraction expansion. Therefore, if d lies outside Wiener's bound, no amount of MSB leakage will make the attack succeed.

In contrast, the Boneh–Durfee attack shows a clear increase in success rate as the MSB leakage level increases. At 5% leakage, the attack does not recover the private exponent for any tested key size. However, the success rate begins to increase as more MSBs are exposed. For 512-bit RSA, the success rate increases gradually and reaches 90% at 25% leakage. For 1024-bit and 2048-bit RSA keys, the attack becomes more effective, reaching 95% success at 25% leakage.

These results show that the Boneh–Durfee attack benefits from partial key exposure because the leaked MSBs reduce the unknown part of the private exponent. Therefore, lattice-based attacks are more effective than continued fraction-based attacks under private exponent MSB leakage conditions.

6 Conclusion

This paper presented a comparative evaluation of Wiener's continued fraction attack and the Boneh–Durfee lattice-based attack on RSA under private exponent MSB leakage. RSA key pairs of 512, 1024 and 2048 bits were generated, and MSB leakage was simulated at increasing levels. The attacks were compared based on success rate, minimum leakage required and effectiveness under partial key exposure.

The results show that Wiener's attack remains ineffective even when MSB leakage is introduced. This is because Wiener's attack primarily depends on the private exponent being sufficiently small, rather than on partial exposure of its bits. In contrast, the Boneh–Durfee attack becomes successful when enough MSBs are leaked. The attack achieves almost certain recovery when approximately one quarter of the private exponent bits are exposed for larger RSA key sizes.

These findings demonstrate that lattice-based cryptanalysis is significantly more effective than continued fraction-based cryptanalysis under partial key exposure. The study also emphasises the importance of protecting RSA private exponent bits from side-channel leakage. Future work may extend this study by analysing least significant bit leakage, random bit leakage, runtime performance and memory consumption across a larger number of RSA instances.

References

[1] 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, pp. 120–126, 1978.

[2] M. J. Wiener, "Cryptanalysis of short RSA secret exponents," in *Advances in Cryptology—EUROCRYPT '89*, Lecture Notes in Computer Science, vol. 434, Springer, 1990, pp. 372–379.

[3] D. Boneh and G. Durfee, "Cryptanalysis of RSA with private key d less than $N^{0.292}$," in *Advances in Cryptology—EUROCRYPT '99*, Lecture Notes in Computer Science, vol. 1592, Springer, 1999, pp. 1–11.

[4] D. Boneh, G. Durfee, and Y. Frankel, "An attack on RSA given a small fraction of the private key bits," in *Advances in Cryptology—ASIACRYPT '98*, Lecture Notes in Computer Science, vol. 1514, Springer, 1998, pp. 25–34.

[5] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, pp. 515–534, 1982.

[6] A. Y. Khinchin, *Continued Fractions*. Chicago, IL, USA: University of Chicago Press, 1964.

[7] E. R. Verheul and H. C. A. van Tilborg, "Cryptanalysis of Less Short RSA Secret Exponents," *Applicable Algebra in Engineering, Communication and Computing*, vol. 8, pp. 425–435, 1997.

[8] A. Dujella, "Continued Fractions and RSA with Small Secret Exponent," *Applicable Algebra in Engineering, Communication and Computing*, vol. 15, no. 5, pp. 341–353, 2004.

[9] D. Coppersmith, "Finding a Small Root of a Univariate Modular Equation," in *Advances in Cryptology – EUROCRYPT '96*, Lecture Notes in Computer Science, vol. 1070, Springer, pp. 155–165, 1996.

[10] J. Blömer and A. May, "Low Secret Exponent RSA Revisited," in *Cryptography and Lattices – CaLC 2001*, Lecture Notes in Computer Science, vol. 2146, Springer, pp. 4–19, 2001.

[11] M. Ernst, E. Jochemsz, A. May, and B. de Weger, "Partial Key Exposure Attacks on RSA up to Full Size Exponents," in *Advances in Cryptology – EUROCRYPT 2005*, Lecture Notes in Computer Science, vol. 3494, Springer, pp. 371–386, 2005.

[12] X. Feng, A. Nitaj, and Y. Pan, "Improved partial key exposure attacks on RSA using MSB leakage and Coppersmith's methods," *IACR Cryptology ePrint Archive*, vol. 2024, no. 1329, pp. 1–21, 2024.

[13] Q. Feng, M. Liu, A. Nitaj, and Y. Pan, "Practical small private exponent attacks against RSA," *Cryptology ePrint Archive*, 2024.

[14] E. Jochemsz and A. May, "Finding small roots of multivariate polynomial equations revisited," in *Advances in Cryptology – ASIACRYPT 2017*, Lecture Notes in Computer Science, vol. 10625, Springer, pp. 3–32, 2017.

[15] D. Genkin, A. Shamir, and E. Tromer, "RSA key extraction via low-bandwidth acoustic cryptanalysis," in *Advances in Cryptology – CRYPTO 2015*, Lecture Notes in Computer Science, vol. 9215, Springer, pp. 444–461, 2015.

[16] J. Lü, Z. Li, W. Wang, and H. Chen, "Side-channel attacks and countermeasures on RSA: A survey," *ACM Computing Surveys*, vol. 47, no. 3, pp. 1–38, 2015.

[17] N. Heninger and H. Shacham, "Reconstructing RSA private keys from random key bits," in *Advances in Cryptology – CRYPTO 2009*, Lecture Notes in Computer Science, vol. 5679, Springer, pp. 1–17, 2009.

[18] P. C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," in *Advances in Cryptology—CRYPTO '96*, Lecture Notes in Computer Science, vol. 1109, Springer, pp. 104–113, 1996.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 427–452.

Security Analysis of a DNA-Inspired Symmetric Lightweight Block Cipher

Nik Azura Nik Abdullah[1,2], Nur Hafiza Zakaria[1,3], A H Azni[1,3] and Abdul Alif Zakaria[2]

[1] Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai 71800, Negeri Sembilan, Malaysia {mzhafiza,ahazni}@usim.edu.my
[2] CyberSecurity Malaysia, Cyberjaya 63000, Selangor, Malaysia
{azura,alif}@cybersecurity.my
[3] CyberSecurity and System Research Unit, Faculty of Science and Technology, Universiti Sains Islam Malaysia, Nilai 71800, Negeri Sembilan, Malaysia {mzhafiza,ahazni}@usim.edu.my

Abstract. This paper presents the design and security evaluation of a DNA-Inspired Symmetric Lightweight Block Cipher featuring a novel DNA-Inspired Substitution Table (DNA S-Box) and DNA-Inspired Permutation Table (DNA P-Box). The DNA S-Box is developed using AES S-Box design principles integrated with DNA technologies, random number generation, and DNA sequences obtained from the National Center for Biotechnology Information (NCBI). The DNA P-Box is constructed using actual DNA sequences together with DNA encoding/decoding rules, the Watson-Crick complementary rule, and DNA hybridization to enhance diffusion. The proposed algorithm is comprehensively evaluated through S-Box testing, avalanche effect analysis, statistical analysis, and cryptanalysis. Experimental results show that the DNA S-Box achieves strong cryptographic properties comparable to the AES S-Box, with Nonlinearity (NL) = 112, Algebraic Degree (AD) = 7, and Differential Uniformity (DU) = 4. The DNA P-Box demonstrates a strong avalanche effect with an average Bit Error Rate (BER) of 50%. Statistical analysis shows that the generated ciphertext successfully passes all 15 NIST randomness tests at a 1% significance level across nine categories of data. Furthermore, the proposed cipher exhibits strong resistance against differential and linear cryptanalysis, achieving maximum differential and linear propagation probabilities of 2^{-84} and 2^{-46}, respectively. These findings demonstrate that the proposed DNA-inspired lightweight block cipher provides strong randomness properties and robust security while remaining suitable for lightweight cryptographic applications.

Keywords: DNA Cryptography · DNA Encoding · DNA Hybridization · Watson-Crick Complementary Rules · S-Box Testing · Avalanche Effect · Statistical Analysis · Differential Cryptanalysis · Linear Cryptanalysis

1 Introduction

The rapid growth of interconnected digital systems, Internet of Things (IoT) devices, wireless sensor networks, and embedded applications has significantly increased the demand for secure yet resource-efficient cryptographic algorithms. Conventional cryptographic algorithms such as the Advanced Encryption Standard (AES) provide strong security; however, their computational complexity, memory requirements, and power consumption may not be suitable for lightweight environments with limited processing and storage capabilities. As a result, lightweight cryptography has emerged as an important research area focused on developing secure cryptographic solutions with lower computational

overhead while maintaining adequate security strength. In particular, symmetric lightweight block ciphers are widely adopted due to their efficiency, speed, and suitability for constrained platforms.

Over the years, researchers and cryptographers have continuously explored various approaches to improve the security and efficiency of cryptographic technologies [6], one of which is DNA cryptography. DNA cryptography is a bio-inspired cryptographic approach derived from DNA computing, where DNA molecules are utilized as information carriers and biological DNA techniques are employed to replace the conventional mathematical operations used in traditional cryptographic systems. This emerging field offers a promising and sustainable approach for enhancing cybersecurity and protecting data privacy.

DNA cryptography exploits the unique properties of DNA, particularly its extraordinary data storage capability, which enables vast amounts of information to be stored within a very small physical medium compared to conventional digital storage technologies. In addition, DNA supports highly parallel operations, allowing multiple computational processes to occur simultaneously, thereby improving the efficiency of encryption and decryption compared to the sequential operations of traditional binary computing systems [2]. Moreover, DNA cryptographic techniques utilize biological processes such as DNA hybridization, annealing, transcription, and translation to generate highly complex cryptographic structures and keys, strengthening the security of the encryption process. The inherent randomness found in DNA sequences further enhances unpredictability and increases resistance against cryptanalytic attacks.

This paper presents the security analysis of a DNA-Inspired Symmetric Lightweight Block Cipher that integrates DNA-based concepts into the construction of substitution and permutation mechanisms to enhance confusion and diffusion properties while maintaining computational efficiency suitable for lightweight environments. The study evaluates the security strength of the proposed algorithm through several analyses, including S-Box testing, avalanche effect testing, statistical analysis, and resistance against differential and linear cryptanalysis attacks. The results demonstrate the capability of the DNA-Inspired Symmetric Lightweight Block Cipher to achieve strong randomness properties and improved security performance, making it a promising candidate for lightweight cryptographic applications.

This paper is organized into five sections. Section 1 presents the introduction and background of the study, including the motivation and objectives of the proposed research. Section 2 discusses the security evaluation methods used to analyze the cryptographic strength of the algorithm, including S-Box testing, avalanche effect analysis, statistical analysis, and cryptanalysis attacks. Section 3 describes the development and construction of the proposed DNA-Inspired Symmetric Lightweight Block Cipher, including the design of the DNA S-Box and DNA P-Box. Section 4 presents the experimental results and security analysis of the proposed algorithm, followed by discussions on its performance and resistance against various attacks. Finally, Section 5 concludes the paper and provides recommendations for future work.

2 Development of the DNA-Inspired Symmetric Lightweight Block Cipher

This section outlines the complete algorithm design and construction of the DNA-Inspired Symmetric Lightweight Block Cipher, which includes the construction of the DNA-Inspired Substitution Table (DNA S-Box) and the DNA-Inspired Permutation Table (DNA P-Box). The complete design of the DNA-Inspired Symmetric Lightweight Block Cipher is also presented in this section.

2.1 Construction of the DNA-Inspired Substitution Table (DNA S-Box)

The DNA S-Box is constructed using AES Substitution Box (AES S-Box) construction techniques. This involves the utilization of two actual, distinct DNA sequences sourced from the National Centre for Biotechnology Information (NCBI) database. The construction of the DNA S-Box incorporates the DNA Encoding / Decoding Rules and a random number generation function.

Constructing the AES S-Box involves a two-step process using finite field arithmetic and an affine transformation [11, 5]. AES S-Box is based on the finite field GF (2^8), which consists of 256 elements. The first step in generating the AES S-Box is to take the multiplicative inverse of each element in the field. The inverse of each element in the field is computed using the following irreducible polynomial: $p(x) = x^8 + x^4 + 8^3 + x + 1$. Once the multiplicative inverse of each byte is computed, the result is passed through an affine transformation process. This affine transformation is a linear operation process that involves multiplying the computed multiplicative inverse by a specific matrix and adding the multiplication result to a constant vector. AES S-Box uses $0x63$ as the constant vector [1].

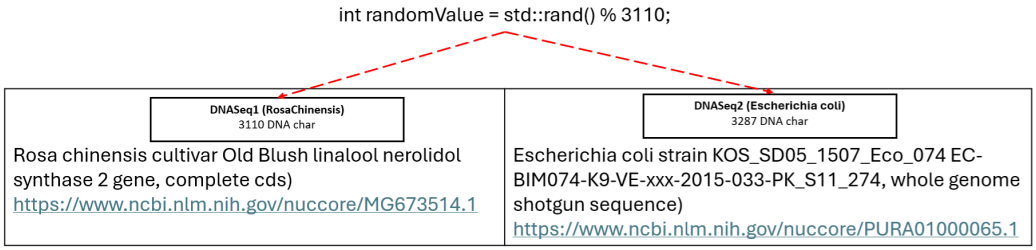
The construction of the DNA S-Box involves modifying the constant vector used in the affine transformation process of AES S-Box construction. The constant vector is derived by using two actual, distinct DNA sequences sourced from the NCBI database, DNA Encoding / Decoding Rules, and a random number generator function. The process for generating the constant vector is illustrated in Figure 1 below.

Eight random values are generated from a random number generator function, each representing the position of nitrogenous bases within the Rosa chinensis cultivar Old Blush linalool nerolidol synthase 2 gene and the Escherichia coli strain KOS_SD05_1507_Eco_074 EC-BIM074-K9-VE-xxx-2015-033-PK_S11_274 sequences. The nitrogenous bases extracted from the Rosa chinensis sequence are referred to as DNA Seq 1, while those from the Escherichia coli sequence are referred to as DNA Seq 2.

The eight random values generate eight corresponding DNA pairs (DNA Pair 1 to DNA Pair 8). The DNA pairs are encoded into a 4-bit binary value using one of the eight DNA Encoding / Decoding Rules as listed in Table 1 below.

Table 1: DNA Rule Pairs.

Rule	C	T	A	G
1	00	01	10	11
2	00	10	01	11
3	11	01	10	00
4	11	10	01	00
5	01	00	11	10
6	10	00	11	01
7	01	11	00	10
8	10	11	00	01



randomValue	DNASeq1	DNASeq2	DNA Pair	DNARule		constant	
						Binary	hexa
41	a	g	DNA Pair 1 - ag	DNARule1	1011	10110110	B6
2917	a	t	DNA Pair 2 - at	DNARule2	0110		
114	a	c	DNA Pair 3 - ac	DNARule3	1011	10110101	B5
1620	a	a	DNA Pair 4 - aa	DNARule4	0101		
509	g	c	DNA Pair 5 - gc	DNARule5	1001	10010010	92
174	t	c	DNA Pair 6 - tc	DNARule6	0010		
2148	c	a	DNA Pair 7 - ca	DNARule7	0100	01001101	4D
1368	t	g	DNA Pair 8 - tg	DNARule8	1101		

Figure 1: Process for Generating the Constant Vector used in DNA S-Boxes.

Each pair of DNA bases produces one 4-bit binary sequence. Two such sequences are then combined to form a single constant value. This process yields four constant values in total: $0xB6$, $0xB5$, $0x92$, and $0x4D$, which are used to generate four DNA S-Boxes through the affine transformation mechanism of AES S-Box construction.

The computed substitution values of the DNA S-Box 1 to DNA S-Box 4 are as follows. These four DNA S-Boxes are subsequently utilized in the Confusion Function, S, within the Round Function, F, of the DNA-Inspired Symmetric Lightweight Block Cipher.

i. Computed Substitution Values of the DNA S-Box 1:

```
{
0xb6, 0xa9, 0xa2, 0xae, 0x27, 0xbe, 0xba, 0x10, 0xe5, 0xd4, 0xb2, 0xfe, 0x2b, 0x02, 0x7e, 0xa3,
0x1f, 0x57, 0x1c, 0xa8, 0x2f, 0x8c, 0x92, 0x25, 0x78, 0x01, 0x77, 0x7a, 0x49, 0x71, 0xa7, 0x15,
0x62, 0x28, 0x46, 0xf3, 0xe3, 0xea, 0x22, 0x19, 0xe1, 0x70, 0x30, 0x24, 0xa4, 0x0d, 0xe4, 0xc0,
0xd1, 0x12, 0xf6, 0x16, 0xcd, 0x43, 0xd0, 0x4f, 0xd2, 0xc7, 0x55, 0x37, 0x3e, 0xf2, 0x67, 0xa0,
0xdc, 0x56, 0xf9, 0xcf, 0xce, 0xbb, 0x8f, 0x75, 0x87, 0xee, 0x03, 0x66, 0xfc, 0x36, 0xfa, 0x51,
0x86, 0x04, 0xd5, 0x38, 0xf5, 0x29, 0x64, 0x8e, 0xbf, 0x1e, 0x6b, 0xec, 0x9f, 0x99, 0x8d, 0x1a,
0x05, 0x3a, 0x7f, 0x2e, 0x96, 0x98, 0xe6, 0x50, 0x90, 0x2c, 0xd7, 0xaa, 0x85, 0xe9, 0x4a, 0x7d,
0x84, 0x76, 0x95, 0x5a, 0x47, 0x48, 0xed, 0x20, 0x69, 0x63, 0x0f, 0xf4, 0xc5, 0x2a, 0x26, 0x07,
0x18, 0xd9, 0xc6, 0x39, 0x8a, 0x42, 0x91, 0xc2, 0x11, 0x72, 0xab, 0xe8, 0xb1, 0x88, 0xcc, 0xa6,
0xb5, 0x54, 0x9a, 0x09, 0xf7, 0xff, 0x45, 0x5d, 0x93, 0x3b, 0x6d, 0xc1, 0x0b, 0x8b, 0xde, 0x0e,
0x35, 0xe7, 0xef, 0xdf, 0x9c, 0xd3, 0xf1, 0x89, 0x17, 0x06, 0x79, 0xb7, 0x44, 0x40, 0x31, 0xac,
0x32, 0x1d, 0xe2, 0xb8, 0x58, 0x00, 0x9b, 0x7c, 0xb9, 0x83, 0x21, 0x3f, 0xb0, 0xaf, 0x7b, 0xdd,
0x6f, 0xad, 0xf0, 0xfb, 0xc9, 0x73, 0x61, 0x13, 0x3d, 0x08, 0xa1, 0xca, 0x9e, 0x68, 0x5e, 0x5f,
0xa5, 0xeb, 0x60, 0xb3, 0x9d, 0xd6, 0x23, 0xdb, 0xb4, 0xe0, 0x82, 0x6c, 0x53, 0x14, 0xc8, 0x4b,
0x34, 0x2d, 0x4d, 0xc4, 0xbc, 0x0c, 0x5b, 0x41, 0x4e, 0xcb, 0x52, 0x3c, 0x1b, 0x80, 0xfd, 0x0a,
0x59, 0x74, 0x5c, 0xd8, 0x6a, 0x33, 0x97, 0xbd, 0x94, 0x4c, 0xf8, 0xda, 0x65, 0x81, 0x6e, 0xc3
};
```

ii. Computed Substitution Values of the DNA S-Box 2.

```
{
0xb5, 0xaa, 0xa1, 0xad, 0x24, 0xbd, 0xb9, 0x13, 0xe6, 0xd7, 0xb1, 0xfd, 0x28, 0x01, 0x7d, 0xa0,
0x1c, 0x54, 0x1f, 0xab, 0x2c, 0x8f, 0x91, 0x26, 0x7b, 0x02, 0x74, 0x79, 0x4a, 0x72, 0xa4, 0x16,
0x61, 0x2b, 0x45, 0xf0, 0xe0, 0xe9, 0x21, 0x1a, 0xe2, 0x73, 0x33, 0x27, 0xa7, 0x0e, 0xe7, 0xc3,
0xd2, 0x11, 0xf5, 0x15, 0xce, 0x40, 0xd3, 0x4c, 0xd1, 0xc4, 0x56, 0x34, 0x3d, 0xf1, 0x64, 0xa3,
0xdf, 0x55, 0xfa, 0xcc, 0xcd, 0xb8, 0x8c, 0x76, 0x84, 0xed, 0x00, 0x65, 0xff, 0x35, 0xf9, 0x52,
0x85, 0x07, 0xd6, 0x3b, 0xf6, 0x2a, 0x67, 0x8d, 0xbc, 0x1d, 0x68, 0xef, 0x9c, 0x9a, 0x8e, 0x19,
0x06, 0x39, 0x7c, 0x2d, 0x95, 0x9b, 0xe5, 0x53, 0x93, 0x2f, 0xd4, 0xa9, 0x86, 0xea, 0x49, 0x7e,
0x87, 0x75, 0x96, 0x59, 0x44, 0x4b, 0xee, 0x23, 0x6a, 0x60, 0x0c, 0xf7, 0xc6, 0x29, 0x25, 0x04,
0x1b, 0xda, 0xc5, 0x3a, 0x89, 0x41, 0x92, 0xc1, 0x12, 0x71, 0xa8, 0xeb, 0xb2, 0x8b, 0xcf, 0xa5,
0xb6, 0x57, 0x99, 0x0a, 0xf4, 0xfc, 0x46, 0x5e, 0x90, 0x38, 0x6e, 0xc2, 0x08, 0x88, 0xdd, 0x0d,
0x36, 0xe4, 0xec, 0xdc, 0x9f, 0xd0, 0xf2, 0x8a, 0x14, 0x05, 0x7a, 0xb4, 0x47, 0x43, 0x32, 0xaf,
0x31, 0x1e, 0xe1, 0xbb, 0x5b, 0x03, 0x98, 0x7f, 0xba, 0x80, 0x22, 0x3c, 0xb3, 0xac, 0x78, 0xde,
0x6c, 0xae, 0xf3, 0xf8, 0xca, 0x70, 0x62, 0x10, 0x3e, 0x0b, 0xa2, 0xc9, 0x9d, 0x6b, 0x5d, 0x5c,
0xa6, 0xe8, 0x63, 0xb0, 0x9e, 0xd5, 0x20, 0xd8, 0xb7, 0xe3, 0x81, 0x6f, 0x50, 0x17, 0xcb, 0x48,
0x37, 0x2e, 0x4e, 0xc7, 0xbf, 0x0f, 0x58, 0x42, 0x4d, 0xc8, 0x51, 0x3f, 0x18, 0x83, 0xfe, 0x09,
0x5a, 0x77, 0x5f, 0xdb, 0x69, 0x30, 0x94, 0xbe, 0x97, 0x4f, 0xfb, 0xd9, 0x66, 0x82, 0x6d, 0xc0
};
```

iii. Computed Substitution Values of the DNA S-Box 3.

```
{
0x92, 0x8d, 0x86, 0x8a, 0x03, 0x9a, 0x9e, 0x34, 0xc1, 0xf0, 0x96, 0xda, 0x0f, 0x26, 0x5a, 0x87,
0x3b, 0x73, 0x38, 0x8c, 0x0b, 0xa8, 0xb6, 0x01, 0x5c, 0x25, 0x53, 0x5e, 0x6d, 0x55, 0x83, 0x31,
0x46, 0x0c, 0x62, 0xd7, 0xc7, 0xce, 0x06, 0x3d, 0xc5, 0x54, 0x14, 0x00, 0x80, 0x29, 0xc0, 0xe4,
0xf5, 0x36, 0xd2, 0x32, 0xe9, 0x67, 0xf4, 0x6b, 0xf6, 0xe3, 0x71, 0x13, 0x1a, 0xd6, 0x43, 0x84,
0xf8, 0x72, 0xdd, 0xeb, 0xea, 0x9f, 0xab, 0x51, 0xa3, 0xca, 0x27, 0x42, 0xd8, 0x12, 0xde, 0x75,
0xa2, 0x20, 0xf1, 0x1c, 0xd1, 0x0d, 0x40, 0xaa, 0x9b, 0x3a, 0x4f, 0xc8, 0xbb, 0xbd, 0xa9, 0x3e,
0x21, 0x1e, 0x5b, 0x0a, 0xb2, 0xbc, 0xc2, 0x74, 0xb4, 0x08, 0xf3, 0x8e, 0xa1, 0xcd, 0x6e, 0x59,
0xa0, 0x52, 0xb1, 0x7e, 0x63, 0x6c, 0xc9, 0x04, 0x4d, 0x47, 0x2b, 0xd0, 0xe1, 0x0e, 0x02, 0x23,
0x3c, 0xfd, 0xe2, 0x1d, 0xae, 0x66, 0xb5, 0xe6, 0x35, 0x56, 0x8f, 0xcc, 0x95, 0xac, 0xe8, 0x82,
0x91, 0x70, 0xbe, 0x2d, 0xd3, 0xdb, 0x61, 0x79, 0xb7, 0x1f, 0x49, 0xe5, 0x2f, 0xaf, 0xfa, 0x2a,
0x11, 0xc3, 0xcb, 0xfb, 0xb8, 0xf7, 0xd5, 0xad, 0x33, 0x22, 0x5d, 0x93, 0x60, 0x64, 0x15, 0x88,
0x16, 0x39, 0xc6, 0x9c, 0x7c, 0x24, 0xbf, 0x58, 0x9d, 0xa7, 0x05, 0x1b, 0x94, 0x8b, 0x5f, 0xf9,
0x4b, 0x89, 0xd4, 0xdf, 0xed, 0x57, 0x45, 0x37, 0x19, 0x2c, 0x85, 0xee, 0xba, 0x4c, 0x7a, 0x7b,
0x81, 0xcf, 0x44, 0x97, 0xb9, 0xf2, 0x07, 0xff, 0x90, 0xc4, 0xa6, 0x48, 0x77, 0x30, 0xec, 0x6f,
0x10, 0x09, 0x69, 0xe0, 0x98, 0x28, 0x7f, 0x65, 0x6a, 0xef, 0x76, 0x18, 0x3f, 0xa4, 0xd9, 0x2e,
0x7d, 0x50, 0x78, 0xfc, 0x4e, 0x17, 0xb3, 0x99, 0xb0, 0x68, 0xdc, 0xfe, 0x41, 0xa5, 0x4a, 0xe7
};
```

iv. Computed Substitution Values of the DNA S-Box 4.

```
{
0x4d, 0x52, 0x59, 0x55, 0xdc, 0x45, 0x41, 0xeb, 0x1e, 0x2f, 0x49, 0x05, 0xd0, 0xf9, 0x85, 0x58,
0xe4, 0xac, 0xe7, 0x53, 0xd4, 0x77, 0x69, 0xde, 0x83, 0xfa, 0x8c, 0x81, 0xb2, 0x8a, 0x5c, 0xee,
0x99, 0xd3, 0xbd, 0x08, 0x18, 0x11, 0xd9, 0xe2, 0x1a, 0x8b, 0xcb, 0xdf, 0x5f, 0xf6, 0x1f, 0x3b,
0x2a, 0xe9, 0x0d, 0xed, 0x36, 0xb8, 0x2b, 0xb4, 0x29, 0x3c, 0xae, 0xcc, 0xc5, 0x09, 0x9c, 0x5b,
0x27, 0xad, 0x02, 0x34, 0x35, 0x40, 0x74, 0x8e, 0x7c, 0x15, 0xf8, 0x9d, 0x07, 0xcd, 0x01, 0xaa,
0x7d, 0xff, 0x2e, 0xc3, 0x0e, 0xd2, 0x9f, 0x75, 0x44, 0xe5, 0x90, 0x17, 0x64, 0x62, 0x76, 0xe1,
0xfe, 0xc1, 0x84, 0xd5, 0x6d, 0x63, 0x1d, 0xab, 0x6b, 0xd7, 0x2c, 0x51, 0x7e, 0x12, 0xb1, 0x86,
0x7f, 0x8d, 0x6e, 0xa1, 0xbc, 0xb3, 0x16, 0xdb, 0x92, 0x98, 0xf4, 0x0f, 0x3e, 0xd1, 0xdd, 0xfc,
0xe3, 0x22, 0x3d, 0xc2, 0x71, 0xb9, 0x6a, 0x39, 0xea, 0x89, 0x50, 0x13, 0x4a, 0x73, 0x37, 0x5d,
0x4e, 0xaf, 0x61, 0xf2, 0x0c, 0x04, 0xbe, 0xa6, 0x68, 0xc0, 0x96, 0x3a, 0xf0, 0x70, 0x25, 0xf5,
0xce, 0x1c, 0x14, 0x24, 0x67, 0x28, 0x0a, 0x72, 0xec, 0xfd, 0x82, 0x4c, 0xbf, 0xbb, 0xca, 0x57,
0xc9, 0xe6, 0x19, 0x43, 0xa3, 0xfb, 0x60, 0x87, 0x42, 0x78, 0xda, 0xc4, 0x4b, 0x54, 0x80, 0x26,
0x94, 0x56, 0x0b, 0x00, 0x32, 0x88, 0x9a, 0xe8, 0xc6, 0xf3, 0x5a, 0x31, 0x65, 0x93, 0xa5, 0xa4,
0x5e, 0x10, 0x9b, 0x48, 0x66, 0x2d, 0xd8, 0x20, 0x4f, 0x1b, 0x79, 0x97, 0xa8, 0xef, 0x33, 0xb0,
0xcf, 0xd6, 0xb6, 0x3f, 0x47, 0xf7, 0xa0, 0xba, 0xb5, 0x30, 0xa9, 0xc7, 0xe0, 0x7b, 0x06, 0xf1,
0xa2, 0x8f, 0xa7, 0x23, 0x91, 0xc8, 0x6c, 0x46, 0x6f, 0xb7, 0x03, 0x21, 0x9e, 0x7a, 0x95, 0x38
};
```

2.2 Construction of the DNA-Inspired Permutation Table (DNA P-Box)

The DNA P-Box is constructed using actual DNA molecules and various DNA technologies. This involves the utilization of two distinct DNA sequences; one is for the content of the P-Box, while the other is to specify the index position of the P-Box. Both DNA sequences are sourced from the NCBI database. The construction of the DNA P-Box incorporates three DNA techniques: the DNA Encoding / Decoding Rules, the Watson-Crick Complementary Rule, and DNA Hybridization. The construction of the DNA P-Box involves a three-stage process; Stage 1 focuses on generating the content of the DNA P-Box, Stage 2 involves creating the index position for the DNA P-Box, and Stage 3 centers on the construction of the DNA P-Box itself.

2.2.1 Stage 1 of DNA P-Box Construction: Generating DNA P-Box Content

In Stage 1 (illustrated in Figure 2), the content within the P-Box is generated using DNA sequence 1, referred to as DNA Seq 1. In this P-Box generation, the Rosa chinensis cultivar Old Blush linalool nerolidol synthase 2 gene is used as the DNA Seq 1. To create the content from DNA Seq 1, three nitrogenous bases are grouped together to form 32 groups of nitrogenous base sequences. If the group of three nitrogenous bases is found to be repeated, one nitrogenous base is to be discarded from the DNA sequence. In total, only 96 nitrogenous bases from DNA Seq 1 are utilized to constitute the 32 groups of three nitrogenous bases. Watson-Crick Complementary Rule is then applied to the 96 nitrogenous bases sequence. In accordance with this rule, the nitrogenous base Adenine (A) is paired with Thymine (T), and the nitrogenous base Guanine (G) is paired with Cytosine (C) due to their complementary nature, which means A is complemented with T and vice versa, and G is complemented with C and vice versa. Subsequently, the procedure of DNA Hybridization is executed on the nitrogenous bases sequence. During this DNA Hybridization process, the nitrogenous bases sequence functions as the sense strand within the DNA structure. This sense strand is then set in the opposite direction, thereby becoming the antiparallel counterpart of the sense strand, which is the antisense strand of the DNA sequence.

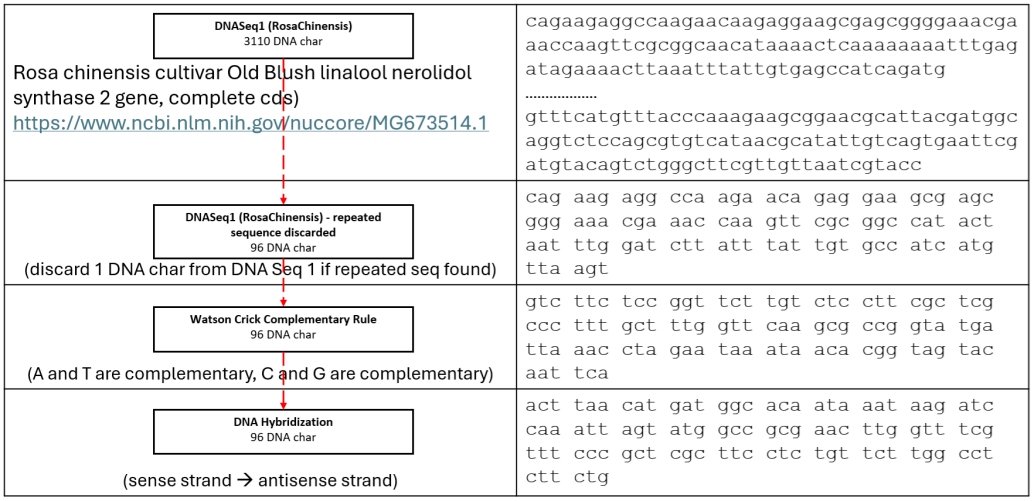


Figure 2: Stage 1 of DNA P-Box Construction: Generating the DNA P-Box Content.

From the Stage 1 process, 32 groups of nitrogenous bases are created. These will be used as the content of the DNA P-Box. DNA Encoding / Decoding Rules 1 to 8 are applied to the nitrogenous bases to represent the 32 groups in binary form. For the purpose of this paper, only DNA Encoding/Decoding Rule 1 is applied to facilitate a clear and structured explanation of the DNA P-Box construction: - Rule 1: 00 - C, 01 - T, 10 - A, and 11 – G. Table 2 shows the generated content of the P-Box in DNA nitrogenous bases form and in decimal representation.

Table 2: Generated Content of P-Box in DNA Nitrogenous Bases Form.

DNASeq1 with DNA Hybridization	act taa cat gat ggc aca ata aat aag atc caa att agt atg gcc gcg aac ttg gtt tcg ttt ccc gct cgc ttc ctc tgt tct tgg cct ctt ctg

64 possible content

0	1	2	3	4	5	6	7
act	taa	cat	gat	ggc	aca	ata	aat
8	9	10	11	12	13	14	15
aag	atc	caa	att	agt	atg	gcc	gcg
16	17	18	19	20	21	22	23
aac	ttg	gtt	tcg	ttt	ccc	gct	cgc
24	25	26	27	28	29	30	31
ttc	ctc	tgt	tct	tgg	cct	ctt	ctg

0	1	2	3	4	5	6	7
33	26	9	57	60	34	38	41
8	9	10	11	12	13	14	15
43	36	10	37	45	39	48	51
16	17	18	19	20	21	22	23
40	23	53	19	21	0	49	12
24	25	26	27	28	29	30	31
20	4	29	17	31	1	5	7

2.2.2 Stage 2 of DNA P-Box Construction: Generating DNA P-Box Index Position

In Stage 2 (illustrated in Figure 3), the index position within the P-Box is generated using DNA sequence 2, referred to as DNA Seq 2. In this P-Box generation, the Escherichia coli strain KOS_SD05_1507_Eco_074 EC-BIM074-K9-VE-xxx-2015-033-PK_S11_274 is used as the DNA Seq 2. To create the index position from DNA Seq 2, the same steps as Stage 1 are repeated. From the Stage 2 process, 32 groups of nitrogenous bases are created. These will be used as the index position of the DNA P-Box. DNA Encoding / Decoding Rule 1 to Rule 8 are applied to the nitrogenous bases to represent the 32 groups in binary form. Table 3 shows the generated index position of the P-Box in DNA nitrogenous bases form and in decimal representation.

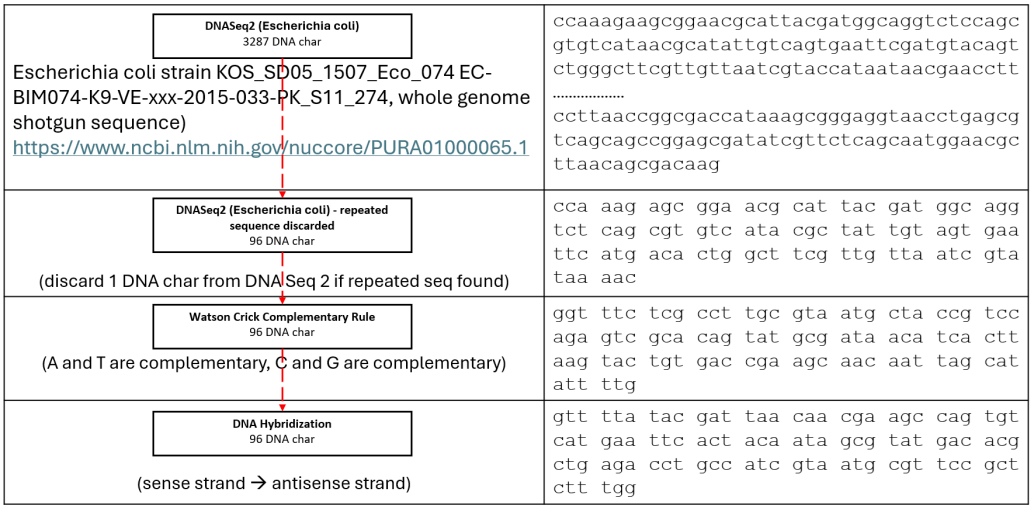


Figure 3: Stage 2 of DNA P-Box Construction: Generating the DNA P-Box Index Position.

Table 3: Generated Index Position P-Box in DNA Nitrogenous Bases Form.

DNASeq1 with DNA Hybridization	gtt tta tac gat taa caa cga agc cag tgt cat gaa ttc act aca ata gcg tat gac acg ctg aga cct gcc atc gta atg cgt tcc gct ctt tgg

64 possible index

0	1	2	3	4	5	6	7
gtt	tta	tac	gat	taa	caa	cga	agc
8	**9**	**10**	**11**	**12**	**13**	**14**	**15**
cag	tgt	cat	gaa	ttc	act	aca	ata
16	**17**	**18**	**19**	**20**	**21**	**22**	**23**
gcg	tat	gac	acg	ctg	aga	cct	gcc
24	**25**	**26**	**27**	**28**	**29**	**30**	**31**
atc	gta	atg	cgt	tcc	gct	ctt	tgg

0	1	2	3	4	5	6	7
58	11	12	60	13	5	7	38
8	**9**	**10**	**11**	**12**	**13**	**14**	**15**
21	30	20	45	10	48	33	35
16	**17**	**18**	**19**	**20**	**21**	**22**	**23**
57	28	44	49	19	39	16	40
24	**25**	**26**	**27**	**28**	**29**	**30**	**31**
34	43	51	22	8	56	18	31

2.2.3 Stage 3 of DNA P-Box Construction: Assembling DNA P-Box

Figure 4 illustrates the Stage 3 process, wherein the DNA P-Box is assembled using the generated DNA P-Box's content from Stage 1 and the generated DNA P-Box's index position from Stage 2. During this phase, the sets of three DNA nitrogenous bases from Stage 2 are transformed into groups of 6-bit binary sequences. The first and the last bits within these 6-bit binary sequences are used to indicate the row of the DNA P-Box, whereas the middle four bits within the binary sequences are used to indicate the column of the DNA P-Box. By determining both the row and column within the DNA P-Box, its index position can be precisely established. Subsequently, the sets of three DNA nitrogenous bases from Stage 1 are inserted into this index position, forming the content of the DNA P-Box.

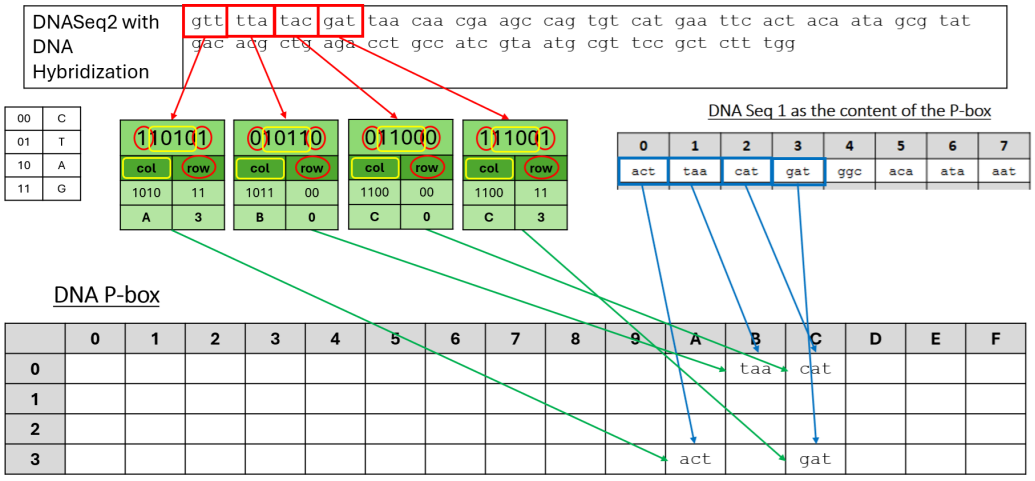


Figure 4: Stage 3 of DNA P-Box Construction: Assembling the DNA P-Box.

The constructed DNA P-Box using DNA Encoding / Decoding Rule 1 is shown in Figure 5. This DNA P-Box is distributed across a table of 64 arrays. To optimize efficiency and conserve resources, the DNA P-Box is shortened into a more concise table with 32 arrays by eliminating any empty arrays within the initial DNA P-Box and reorganizing and mapping the active values into a new compact substitution box format for cryptographic use. The constructed DNA P-Boxes using DNA Encoding / Decoding Rule 1 to Rule 8

are shown in Table 4.

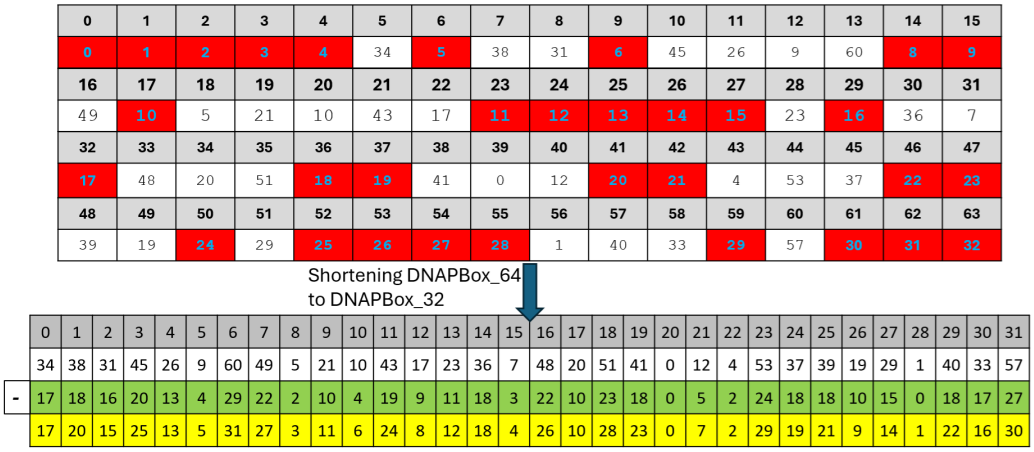


Figure 5: Shortening of the DNA P-Box from 64-bit to 32-bit.

Table 4: DNA P-Boxes using DNA Encoding/Decoding Rule 1 to Rule 8.

DNA P-Box 1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	17	20	15	25	13	5	31	27	3	11	6	24	8	12	18	4
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	26	10	28	23	0	7	2	29	19	21	9	14	1	22	16	30
DNA P-Box 2	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	27	2	5	18	16	21	11	8	12	22	14	26	19	28	24	0
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	25	3	23	17	13	7	1	30	29	9	31	20	6	10	15	4
DNA P-Box 3	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58
DNA P-Box 4	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	15	9	30	17	22	10	12	2	29	24	31	8	3	21	5
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	27	13	19	23	7	25	20	28	4	0	26	18	6	16	11	14
DNA P-Box 5	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	23	10	2	8	0	17	3	15	31	28	7	6	14	20	25	27
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	26	11	30	21	5	24	4	9	13	16	22	1	19	18	29	12
DNA P-Box 6	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	26	24	4	1	20	27	18	17	13	8	7	15	9	16	31	22
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	14	5	10	21	2	29	23	0	25	30	28	3	11	19	6	12
DNA P-Box 7	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	19	25	12	20	28	3	1	6	31	8	2	29	10	21	26	17
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	9	0	15	22	16	24	23	18	14	13	4	11	30	27	7	5
DNA P-Box 8	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	19	2	13	12	30	9	15	18	22	27	7	26	10	1	20	5
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	4	6	11	17	25	24	3	0	16	28	14	31	23	29	21	8

To simplify the construction process of a DNA P-Box, Figure 6 presents a flowchart detailing the three stages involved.

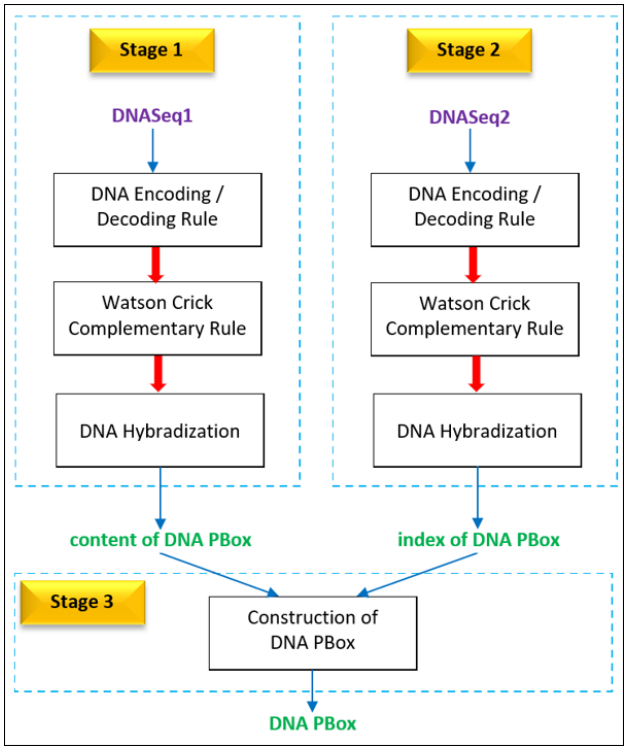


Figure 6: Flowchart of DNA P-Box Construction.

2.3 Design of the DNA-Inspired Symmetric Lightweight Block Cipher

The DNA-Inspired Symmetric Lightweight Block Cipher is designed based on the architecture of the LBlock Block Cipher algorithm. It adopts a modified Feistel structure with a 64-bit block size and utilizes an 80-bit key. The algorithm performs 32 encryption rounds, similar to LBlock Block Cipher, but introduces significant enhancements through DNA-inspired mechanisms. Unlike the original LBlock Block Cipher, which relies on traditional S-Box for its Confusion Function and P-Box for its Diffusion Function, this algorithm replaces them with dynamically generated DNA S-Boxes and DNA P-Boxes derived from actual biological sequences.

The encryption algorithm of the DNA-Inspired Symmetric Lightweight Block Cipher differs from the original LBlock cipher primarily in its integration of DNA-based components. Specifically, it incorporates the eight DNA Encoding/Decoding Rules within all 32 encryption and decryption rounds. It replaces the conventional S-Boxes and P-Boxes with dynamically generated DNA S-Boxes and DNA P-Boxes in the Round Function. The key schedule of the DNA-Inspired Symmetric Lightweight Block Cipher adopts the same key scheduling mechanism as the LBlock block cipher, maintaining its original structure and operations.

2.3.1 Round Function, F

Round Function, F in the encryption algorithm of the DNA-Inspired Symmetric Lightweight Block Cipher is used repeatedly in all 32 rounds. This function is shown in Figure 7. There are also two other sub-functions that construct this function:

i. Confusion Function, S to provide a non-linear layer

ii. Diffusion Function, P to provide a linear layer

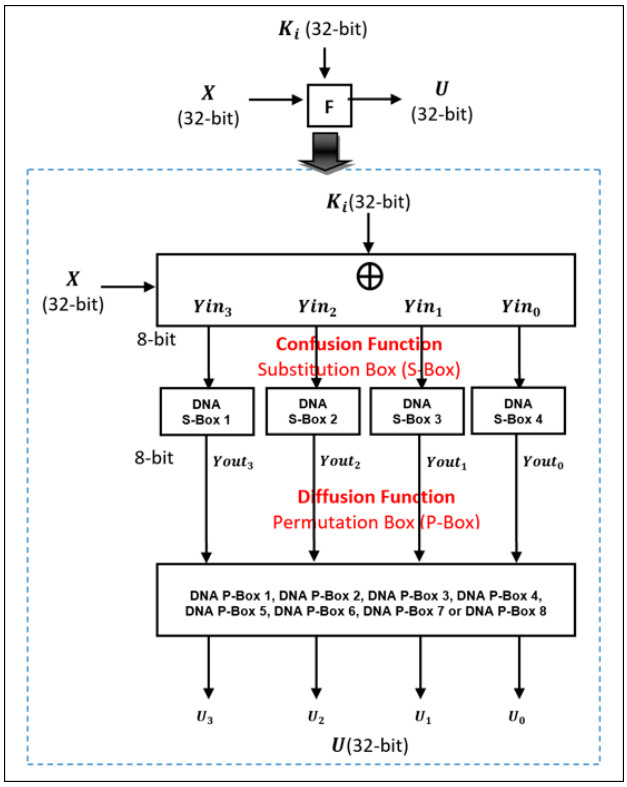


Figure 7: Round Function, F of the DNA-Inspired Symmetric Lightweight Block Cipher, which includes the Confusion Function, S, and Diffusion Function, P.

2.3.2 Confusion Function, S

The Confusion Function, S in the encryption algorithm of the DNA-Inspired Symmetric Lightweight Block Cipher, which is the non-linear layer, contains four 16 X 16 DNA S-Boxes in parallel to replace the original eight 4 X 4 S-Boxes of LBlock Block Cipher. The four 16 X 16 DNA S-Boxes, namely DNA S-Box 1, DNA S-Box 2, DNA S-Box 3, and DNA S-Box 4.

This function accepts a 32-bit data, X and a 32-bit round subkey, K_i as inputs. Exclusive OR operation is applied towards X and K_i to produce Yin which is then divided into four bytes (8-bit binary sequence); Yin_0, Yin_1, Yin_2, and Yin_3. These four bytes are the inputs for the four DNA S-Boxes. Outputs $Yout_0$, $Yout_1$, $Yout_2$, and $Yout_3$ transformed from all four DNA S-Boxes are concatenated to construct $Yout$. The Confusion Function, S is clearly defined as follows:

$$Yin = Yin_3||Yin_2||Yin_1||Yin_0 \rightarrow Yout = Yout_3||Yout_2||Yout_1||Yout_0 \quad (1)$$

where

$$Yout_3 = \text{DNA S-Box 1 } (Yin_3),$$
$$Yout_2 = \text{DNA S-Box 2 } (Yin_2),$$
$$Yout_1 = \text{DNA S-Box 3 } (Yin_1),$$
$$Yout_0 = \text{DNA S-Box 4 } (Yin_0),$$

2.3.3 Diffusion Function, P

The Diffusion Function, P in the encryption algorithm of the DNA-Inspired Symmetric Lightweight Block Cipher, which is the linear layer, permutes inputs $Yout_0$, $Yout_1$, $Yout_2$, and $Yout_3$ to produce outputs U_0, U_1, U_2, and U_3. The four produced bytes are then concatenated to construct U. The Diffusion Function, P is clearly defined as follows:

$$U = U_3||U_2||U_1||U_0 \rightarrow Yout = Yout_3||Yout_2||Yout_1||Yout_0 \tag{2}$$

Note that the DNA P-Box used in this Diffusion Function, P for the 32 encryption rounds are selected based on Table 5.

Table 5: DNA P-Box used in the 32 encryption rounds.

DNA P-Box	1	2	3	4	5	6	7	8
Round	1	2	3	4	5	6	7	8
	9	10	11	12	13	14	15	16
	17	18	19	20	21	22	23	24
	25	26	27	28	29	30	31	32

3 Security Evaluation of Cryptographic Algorithms

The robustness of a cryptographic algorithm is fundamentally determined by its ability to resist a wide range of analytical attacks and statistical evaluations. To assess the strength of a cryptographic block cipher algorithm, rigorous security evaluations are required. These evaluations can be broadly categorized into structural testing, statistical analysis, and cryptanalytic resistance.

First, S-Box Testing focuses on measuring properties such as nonlinearity (NL), algebraic degree (AD), differential uniformity (DU), and compliance with the Strict Avalanche Criterion (SAC). Second, the avalanche effect is examined to verify that small changes in the plaintext or key lead to significant, unpredictable changes in the ciphertext. This evaluation involves metrics such as the Correlation Coefficient, Bit Error Rate (BER), and Key Sensitivity, which collectively measure the effectiveness of diffusion across the algorithm. Third, Statistical Analysis is conducted to evaluate the randomness properties of the algorithm under various data categories. Additionally, randomness tests and the NIST Statistical Test Suite provide a comprehensive framework for assessing whether the ciphertext outputs approximate the characteristics of a truly random sequence, which is an essential requirement for secure encryption. Finally, algorithms must be tested against known cryptanalytic attacks, particularly Differential Cryptanalysis and Linear Cryptanalysis, which remain two of the most powerful methods for assessing block cipher strength. Resistance to these attacks demonstrates the effectiveness of the cipher's confusion and diffusion mechanisms, as well as its overall structural soundness.

3.1 S-Box Testing

Substitution boxes (S-Boxes) are fundamental components of modern block ciphers, providing the essential nonlinearity that ensures resistance against a wide range of cryptanalytic attacks. The design and evaluation of S-Boxes have been extensively studied in the cryptographic literature, as their quality directly impacts the overall security of the cipher. An S-Box with weak properties may leave the cipher vulnerable to attacks such as linear cryptanalysis, differential cryptanalysis, or higher-order differential attacks.

To assess the strengths and weaknesses of S-Boxes, researchers have generally employed two complementary approaches. The first approach focuses on classical cryptographic properties, namely Nonlinearity (NL), Algebraic Degree (AD), and Differential Uniformity (DU) [7]. These properties provide well-established criteria to evaluate resistance against linear and differential cryptanalysis, as well as higher-order attacks. The second approach adopts modern testing techniques, particularly the Strict Avalanche Criterion (SAC) proposed by [8] in 2008. This framework extends classical evaluations by examining dynamic behaviours of S-Boxes through properties such as the avalanche effect, completeness, and strong S-Box property, offering a more comprehensive perspective on S-Box robustness.

In this paper, the discussion of both classical and SAC-based evaluation methods is demonstrated using the Advanced Encryption Standard (AES) S-Box, which serves as the reference model. The AES S-Box is chosen not only for its widespread adoption but also for its well-documented cryptographic strength, making it an ideal benchmark for illustrating S-Box analysis.

3.1.1 Nonlinearity

Nonlinearity (NL) is one of the most important criteria in measuring the cryptographic strength of an S-Box. High nonlinearity ensures resistance against linear cryptanalysis. For an S-Box in $GF(2^n)$, the upper bound of nonlinearity is defined as $NL(f) = 2^{n-1} - (2^{-1} * 2^{n/2})$.

For the AES S-Box, which is defined over $GF(2^8)$, the upper bound value of nonlinearity is 120. A secure S-Box requires its nonlinearity parameter to fall within $100 < NL < 120$. The AES S-Box achieves a nonlinearity of 112 [7, 4], thus, the S-Box of AES is not susceptible to linear cryptanalysis attack. This satisfies that the AES S-Box meets the characteristic of good nonlinearity property.

3.1.2 Algebraic Degree

The Algebraic Degree (AD) of an S-Box is another essential property for cryptographic resistance, particularly against higher-order differential attacks. A general guideline suggests that an S-Box should have $AD \geq 4$ to ensure robustness. The AD is calculated as: $AD = r - 1$, where r is the degree of the S-Box. For the AES S-Box, the algebraic degree is calculated as $AD = 8 - 1 = 7$ [7, 4]. This high algebraic degree demonstrates that the AES S-Box provides strong resistance to higher-order differential cryptanalysis.

3.1.3 Differential Uniformity

The Differential Uniformity (DU) of an S-Box quantifies its resistance to differential cryptanalysis. To evaluate this property, the input and output difference pairs of the S-Box are tabulated in a difference uniformity table, where each entry denotes the frequency of an output difference corresponding to a specific input difference. The highest frequency in this table represents the DU value. An effective S-Box generally exhibits a DU within the range of $2 \leq DU \leq 6$. For instance, the AES S-Box has a DU value of 4 [7, 4], which falls within the secure threshold. This result indicates that the AES S-Box is resistant to differential cryptanalysis, thereby satisfying the requirement for good DU property.

3.1.4 Strict Avalanche Criterion

The Strict Avalanche Criterion (SAC) was first proposed by [17] in 1985 within the context of S-Box design, aiming to ensure that complementing a single input bit causes each output bit to change with a probability of one-half. Originally introduced to strengthen the design of substitution-permutation networks (SPNs), SAC asserts that changing a single input bit should change each output bit with a probability of one-half. In other words, flipping one input bit should ideally cause half of the output bits to flip. This property ensures that the cipher exhibits strong diffusion, making it more resistant to correlation-based attacks.

[8] in 2008 extended the application of SAC to S-Box testing and proposed a framework to measure three key properties: the avalanche effect, completeness, and the strong S-Box property.

i. Avalanche Effect: The avalanche effect is assessed by analysing the distribution of Hamming weights in the output when random inputs are applied to the S-box. For an ideal S-box, flipping one input bit should lead to an approximately equal probability of flipping each output bit. The property is considered satisfied if the resulting frequency distribution of Hamming weights follows a normal distribution curve [8, 4]. In practice, the AES S-Box has been shown to satisfy this property, ensuring strong diffusion and minimizing statistical biases.

ii. Completeness Property: Completeness is determined by analysing the distribution of differential outputs under random inputs. If the frequencies of differential output vary, the tested S-Box is poor, whereas if the frequencies are uniformly distributed, then the tested S-Box is considered as a good S-Box [8, 4].

iii. Strong S-Box Property: This test examines Hamming weights across different bit positions under random inputs. If the frequencies of hamming weight vary, the S-Box tested is poor, whereas if the frequencies are uniformly distributed, then the S-Box tested is considered as a good S-Box [8, 4].

3.2 Avalanche Effect

The avalanche effect is a critical property in the cryptographic testing of encryption algorithms. It refers to a phenomenon where a small change in the input data, the encryption key, or any part of the cryptographic algorithm, such as the initialization vectors, leads to a significantly different output or ciphertext [14]. This property is essential for the security and strength of cryptographic systems. A good cryptographic algorithm should always satisfy the following relation: Avalanche effect $\approx 50\%$. It should be highly sensitive to even the slightest changes in the input data. In other words, a change in a 1-bit plaintext or encryption key would affect half of the output bits, where it could cause a substantial and unpredictable change in the resulting ciphertext. This sensitivity ensures that attackers cannot easily deduce information about the input or key by analyzing the ciphertext.

3.2.1 Correlation Coefficient

The Correlation Coefficient in the context of the avalanche effect in cryptography refers to a statistical measure used to quantify the degree of non-linear relationship between two sets of data [13]. In the context of this research, the Correlation Coefficient is used to assess the degree to which changes in the input (plaintext or key) correlate with changes in the output (ciphertext) of a cryptographic algorithm. This is denoted by r_{pc} and can be calculated using the following formula [18], where E is the avalanche effect, p_i is the ith plaintext bit, and c_i is the ith ciphertext bit.

$$r_{pc} = \frac{\sum_{i=1}^{s}(p_i - E)(c_i - E)}{\sqrt{\sum_{i=1}^{s}(p_i - E)^2}\sqrt{\sum_{i=1}^{s}(c_i - E)^2}} \tag{3}$$

The r_{pc} takes values range from +1 to -1, where the correlation matrix is listed in Table 6. A lower r_{pc} indicates a stronger avalanche effect and a more secure encryption algorithm. Cryptographers use this measure to evaluate the algorithm's resistance to certain types of attacks, such as linear and differential cryptanalysis, and to ensure that it behaves in a way that enhances data security.

Table 6: Correlation Matrix.

Condition	Result
$r_{pc} = 0$	Non-linear relationship
$-0.3 \leq r_{pc} < 0$ or $0 < r_{pc} \leq 0.3$	Weak negative/positive linear relationship
$-0.7 < r_{pc} < -0.3$ or $0.3 < r_{pc} < 0.7$	Moderate negative/positive linear relationship
$-1 < r_{pc} \leq -0.7$ or $0.7 \leq r_{pc} < 1$	Strong negative/positive linear relationship
$r_{pc} = -1$ or $r_{pc} = 1$	Perfect negative/positive linear relationship

3.2.2 Bit Error Rate

Bit Error Rate, denoted by BER, is a critical measurement that is usually used in telecommunication and digital communication to measure the percentage of incorrect bits received compared to the total number of bits transmitted over any communication medium. This concept is applied in measuring the difference in output bits (ciphertext) caused by a change in its associated input bits (plaintext) of a cryptographic algorithm [12]. An optimum BER should be 0.5 or 50% modification of the total plaintext bits. BER can be calculated using the following formula [13].

$$BER = \frac{\text{Number of ciphertext bit difference}}{\text{Total number of ciphertext}} \tag{4}$$

3.2.3 Key Sensitivity

Key Sensitivity Test is an important part of cryptographic algorithm analysis and validation, to confirm that the tested cryptographic algorithms meet the fundamental security requirement of being highly sensitive to changes in the key, making them robust against various forms of cryptanalysis. Key Sensitivity Test assesses how sensitive a cryptographic algorithm is to changes in its encryption key. The purpose of this test is to determine if a minor modification exists in the encryption key, even by replacing a single bit of the encryption key, it will impact the resulting ciphertext [13]. The Key Sensitivity properties are observed using the BER calculation, where the optimum BER should be 0.5 or 50% modification of the total ciphertext bits.

3.3 Statistical Analysis

Statistical analysis serves as a fundamental approach in evaluating the cryptographic strength and randomness quality of block cipher algorithms. It ensures that the ciphertext exhibits statistical properties equivalent to truly random data, thereby minimizing susceptibility to statistical or differential attacks.

Ciphertext produced from block ciphers consists of a sequence of bits with a length equal to the block size of the block cipher. However, evaluating the randomness of a cryptographic algorithm requires a ciphertext with a significantly larger sequence of bits. To achieve this, Data Categories are utilized to generate inputs, such as plaintext or key, for block ciphers. These inputs produce ciphertexts that are concatenated in a specific

manner to form a longer sequence. [15, 16] in 1999 - 2000 outlined nine different categories of data required to conduct the statistical analysis. These data categories are as presented in Table 7 below.

Table 7: Data Categories.

Data Category
1. Strict Key Avalanche (SKA)
2. Strict Plaintext Avalanche (SPA)
3. Plaintext / Ciphertext Correlation (PCC)
4. Cipher Block Chaining Mode (CBCM)
5. Random Plaintext / Random Key (RPRK)
6. Low Density Key (LDK)
7. High Density Key (HDK)
8. Low Density Plaintext (LDP)
9. High Density Plaintext (HDP)

The NIST Statistical Test Suite, developed by the National Institute of Standards and Technology (NIST), is a comprehensive statistical tool used to assess the randomness of binary sequences produced by hardware- or software-based cryptographic random or pseudorandom number generators. It is designed to detect various forms of non-randomness that may exist within a sequence. The suite includes fifteen statistical tests, which are classified into two main categories [19]. Non-Parameterized Test Selection consists of eight tests that do not require users to input any parameter to obtain p-values for each test. Parameterized Test Selection consists of seven tests that require users to define one or two parameter value(s). All fifteen tests are listed according to their category as per listed in Table 8 below.

Table 8: NIST Statistical Tests.

Non-Parameterized Test Selection	Parameterized Test Selection
1. Frequency Test	9. Block Frequency Test
2. Runs Test	10. Non-Overlapping Templates Test
3. Longest Runs of Ones Test	11. Overlapping Template Test
4. Spectral Discrete Fourier Transform (DFT) Test	12. Maurer's Universal Test
5. Cumulative Sums (Forward/Reverse) Test	13. Linear Complexity Test
6. Random Excursion Variant Test	14. Serial Test
7. Random Excursion Test	15. Approximate Entropy Test
8. Binary Matrix Rank Test	

3.4 Cryptanalysis Attacks

Cryptanalysis has long been the principal method for evaluating the strength of cryptographic algorithms. It involves a systematic study of techniques aimed at exposing weaknesses in cryptographic primitives and assessing their robustness against potential adversarial strategies. Among the many cryptanalytic techniques that have emerged, Differential Cryptanalysis and Linear Cryptanalysis are regarded as two of the most powerful and influential methods in the study of symmetric-key cryptography. Both techniques played a decisive role in revealing vulnerabilities in the Data Encryption Standard (DES) [3, 10] and have since become standard benchmarks in the design and evaluation of modern block ciphers.

Differential Cryptanalysis, pioneered by [3] in 1990, exploits statistical patterns in the propagation of input differences through multiple rounds of a cipher to derive information about the secret key. Linear Cryptanalysis, introduced by [9] in 1993, applies linear approximations to model the nonlinear components of a cipher, thereby correlating plaintext, ciphertext, and key bits.

4 Security Analysis Results of the DNA-Inspired Symmetric Lightweight Block Cipher

This section provides a comprehensive security evaluation and assessment of the DNA-Inspired Symmetric Lightweight Block Cipher's strength. The results not only highlight the ability of the proposed algorithm to achieve Shannon's confusion and diffusion properties but also demonstrate its resilience against statistical biases and classical cryptanalytic techniques.

4.1 Results and Analysis for S-Box Testing

The S-Box Testing performed for this research project focuses on the four newly constructed DNA S-Boxes. The S-Box Testing results are summarized in Table 9. For benchmarking purposes, their results are compared against those obtained from the AES S-Box, which serves as the standard reference. The comparative analysis demonstrates that the DNA S-Boxes exhibit cryptographic properties closely aligned with those of the AES S-Box. Overall, the findings indicate that the proposed DNA S-Boxes successfully replicate the key security characteristics of the AES S-Box, thereby validating their suitability for use in secure cryptographic applications.

Table 9: S-Box Testing Results for DNA S-Box 1, S-Box 2, S-Box 3, and S-Box 4

S-Box Testing	Ideal/Expected Value	AES S-Box	DNA S-Box 1	DNA S-Box 2	DNA S-Box 3	DNA S-Box 4
Nonlinearity (NL)	$100 < NL < 120$	112	112	112	112	112
Algebraic Degree (AD)	$AD \geq 4$	7	7	7	7	7
Differential Uniformity (DU)	$2 \leq DU \leq 6$	4	4	4	4	4
Avalanche Effect	Normally distributed Hamming weights.	Normal distribution	Normal distribution	Normal distribution	Normal distribution	Normal distribution
Completeness Property	Frequencies of differential output are uniformly distributed.	Uniform distribution	Uniform distribution	Uniform distribution	Uniform distribution	Uniform distribution
Strong S-box Property	Hamming weights across bit positions are uniformly distributed.	Uniform distribution	Uniform distribution	Uniform distribution	Uniform distribution	Uniform distribution

4.2 Results and Analysis for Avalanche Effect Testing

The Avalanche Effect testing performed for this research project focuses on the newly constructed DNA P-Box, where the output generated from the DNA P-Box will be examined. The Correlation Coefficient analysis and Bit Error Rate testing utilize 32 random samples, with each input sample consisting of precisely 32 bits. These 32 randomly selected samples were subsequently inserted into the DNA P-Box, from which 32 corresponding outputs were generated. The assessment of Key Sensitivity involves the use of five random keys, each containing 32 bits, where each of the five random keys is XORed with 32 random input samples, also comprising 32 bits each. The resulting output from this XOR operation was then integrated into the DNA P-Box, which yielded 32 corresponding output sequences. The avalanche effect was also tested on four other existing 32-bit permutation tables utilized in ANU, VAYU, SFN, and MIBS cryptographic algorithms. For the comparison analysis activities, 50 random samples, each with precisely 32 bits, were utilized for all five P-Boxes. In total, 492 input–output relationships were analyzed, derived from the P-Box transformations and computed as $(32 + (32 * 5keys)) + (50 + (50 * 5keys))$. The 32-bit

permutation tables utilized in ANU, VAYU, SFN, and MIBS cryptographic algorithms are as shown in Table 10 below.

Table 10: The 32-bit permutation tables utilized in ANU, VAYU, SFN, and MIBS cryptographic algorithms.

Algorithm	Permutation Table
ANU [32]	{20, 16, 28, 24, 17, 21, 25, 29, 22, 18, 30, 26, 19, 23, 27, 31, 11, 15, 3, 7, 14, 10, 6, 2, 9, 13, 1, 5, 12, 8, 4, 0}
VAYU [32]	{24, 8, 0, 19, 23, 28, 12, 4, 9, 25, 18, 1, 13, 5, 22, 29, 17, 2, 10, 26, 6, 14, 30, 21, 3, 16, 27, 11, 31, 20, 7, 15}
SFN [32]	{9, 28, 7, 13, 8, 12, 29, 6, 0, 2, 17, 23, 30, 24, 18, 11, 31, 4, 15, 19, 5, 1, 25, 27, 3, 10, 22, 21, 26, 16, 20, 14}
MIBS [32]	{4, 5, 6, 7, 28, 29, 30, 31, 0, 1, 2, 3, 8, 9, 10, 11, 20, 21, 22, 23, 24, 25, 26, 27, 12, 13, 14, 15, 16, 17, 18, 19}

4.2.1 Results and Analysis for Correlation Coefficient

The Correlation Coefficient values, r_{pc} resulting from the examination of the 32 generated input-output pairs from the tested DNA P-Box is shown in Figure 8 below. It is shown that these values are distributed around the lower range of r_{pc} and are summarized in Table 11. The r_{pc} values span from -0.16129, the minimum value, to 0.20000, the maximum value. The average r_{pc} value for this analysis is 0.01665, and the standard deviation is 0.09613. Values concentrated around the lower range of r_{pc} indicates that the DNA P-Box being tested exhibits a strong avalanche effect, which also implies that the constructed DNA P-Box should operate as a secure component in an encryption algorithm.

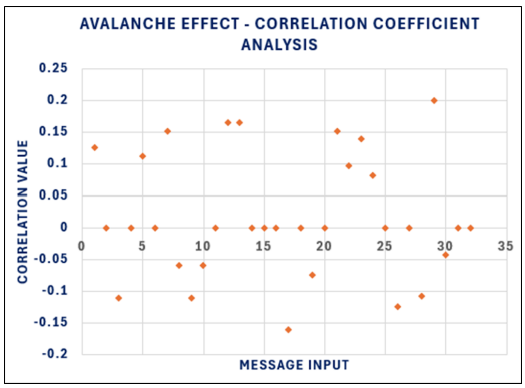


Figure 8: Distribution of the Correlation Coefficient, r_{pc} Results Computed from the Input-Output Relationship of the DNA P-Box.

Table 11: Summary of Correlation Coefficient Analysis Results for DNA P-Box.

Condition	Result	Number of r_pc
$r_{pc} = 0$	Non-linear relationship	13
$0 < r_{pc} \leq 0.3$	Weak positive linear relationship	10
$-0.3 < r_{pc} \leq 0$	Weak negative linear relationship	9

In addition, comparisons have been made to the Correlation Coefficient resulting from the examination of the 50 generated input-output pairs from the tested DNA P-Box, and P-Boxes used in the ANU, VAYU, SFN, and MIBS cryptographic algorithms. The average, minimum, and maximum r_{pc} values are presented in Table 12. It is demonstrated that the average r_{pc} value from the assessment of the DNA P-Box is closer to zero, signifying a stronger non-linear relationship compared to the permutation tables of other algorithms.

Table 12: Correlation Coefficient Analysis Results for DNA P-Box and P-Boxes used in the ANU, VAYU, SFN, and MIBS.

Message Input	r_{pc}				
	DNA P-Box	ANU P-Box	VAYU P-Box	SFN P-Box	MIBS P-Box
Average	0.02150	0.03575	0.05510	0.02475	0.02340
Minimum	-0.16129	-0.21212	-0.37500	-0.25714	-0.21212
Maximum	0.49367	0.36842	0.36842	0.40299	0.40000

4.2.2 Results and Analysis for Bit Error Rate

The BER values resulting from the examination of the 32 generated input-output pairs from the tested DNA P-Box are shown in Figure 9 below. It is shown that these values are concentrated around the optimum BER value of 0.5, with the minimum value of 0.4375 and the maximum value of 0.6250. The average BER value for this analysis is 0.50195, which is around 50.20%. This finding indicates that the Bit Error Rate, in terms of bits, amounts to 16.06. The BER assessment exhibits that 50% of the total input bits undergo modification after entering the DNA P-Box.

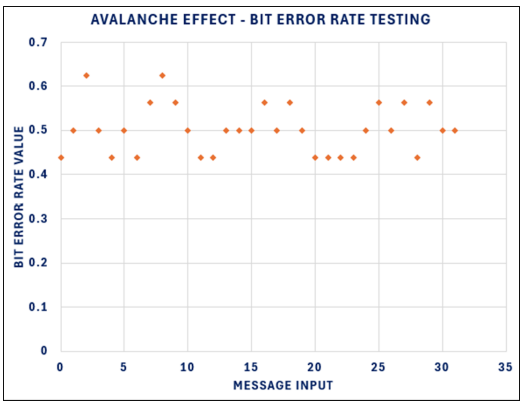


Figure 9: Distribution of the Bit Error Rate, BER Results Computed from the Input-Output Relationship of the DNA P-Box.

Additionally, comparisons have been conducted with the BER values from the permutation table used by other algorithms. The average, minimum, and maximum r_{pc} values resulting from the examination of the 50 generated input-output pairs from the tested DNA P-Box, and P-Boxes used in the ANU, VAYU, SFN, and MIBS cryptographic algorithms are as detailed in Table 13. The assessment indicated that the average BER of the DNA P-Box is closer to the optimum BER value of 50%, compared to the permutation tables of other algorithms.

Table 13: Correlation Coefficient Analysis Results for DNA P-Box and P-Boxes used in the ANU, VAYU, SFN, and MIBS.

Message Input	BER				
	DNA P-Box	ANU P-Box	VAYU P-Box	SFN P-Box	MIBS P-Box
Average	49.88%	48.75%	48.38%	50.88%	50.88%
Minimum	37.50%	0.00%	37.50%	31.25%	37.50%
Maximum	62.50%	0.00%	68.75%	68.75%	68.75%

4.2.3 Results and Analysis for Key Sensitivity

When evaluating the Key Sensitivity characteristics of the DNA P-Box, the Correlation Coefficient analysis and the Bit Error Rate testing are employed to investigate the impact of integrating the random keys into the input of the DNA P-Box. The resulting Correlation Coefficient values, r_{pc}, derived from the examination of the 32 generated input-output pairs involving five random keys, are presented in Figure 10 below. The average r_{pc} values and summary for this analysis are presented in Table 14. The r_{pc} values range from the minimum value of -0.44262 to the maximum value of 0.57447. These values are distributed within the lower range of r_{pc}. This observation shows that even with the incorporation of keys into the input of the DNA P-Box, the tested DNA P-Box still demonstrates a strong avalanche effect.

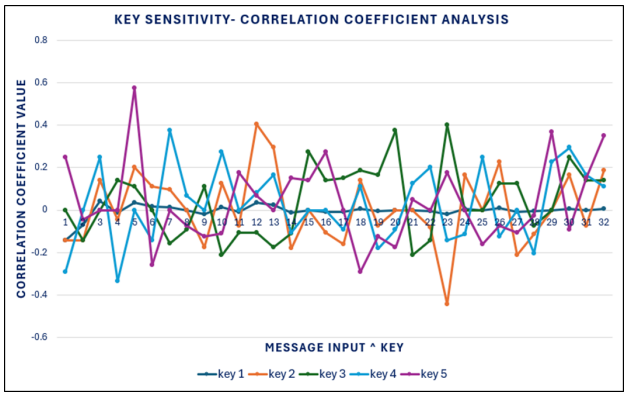


Figure 10: Distribution of the Correlation Coefficient, r_{pc} Results Computed from the Input-Output Relationship of the DNA P-Box with the Influence of Random Keys.

Table 14: Summary of Correlation Coefficient Analysis Results for DNA P-Box with the Influence of Random Keys.

Condition	Without Key	Key 1	Key 2	Key 3	Key 4	Key 5
$r_{pc} = 0$	13	6	6	7	7	10
$0 < r_{pc} \leq 0.3$	10	12	15	4	12	11
$-0.3 < r_{pc} \leq 0$	9	14	11	11	13	11
Average Correlation	0.01665	0.00690	0.04036	0.02713	0.03330	0.02221

The Bit Error Rate, BER, derived from analysis of the 32 generated input-output pairs involving five random keys, is detailed in Table 15 and illustrated in Figure 11 below. The overall minimum BER value for this testing is 0.3125, and the maximum BER value for this testing is 0.6875. The average BER values are concentrated around the optimum BER value of 0.5, ranging from a minimum average value of 0.49219 (49.22%) to a maximum of 0.51172 (51.17%). This finding implies that the average Bit Error Rate, in terms of bits, ranges from 15.750 to 16.375. The BER assessment exhibits that 50% of the total input bits undergo modification after entering the DNA P-Box.

Table 15: Bit Error Rate Testing Results for DNA P-Box with the Influence of Random Keys.

Message Input	BER					
	Without Key	Key 1	Key 2	Key 3	Key 4	Key 5
Average	50.20%	51.17%	49.80%	49.22%	50.20%	50.20%
Minimum	43.75%	31.25%	31.25%	31.25%	31.25%	31.25%
Maximum	62.50%	68.75%	68.75%	68.75%	68.75%	68.75%

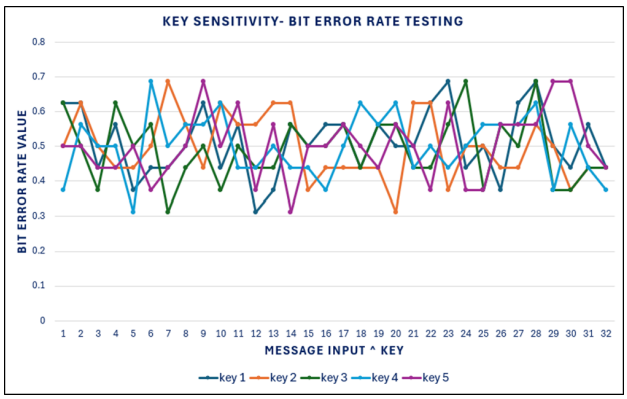


Figure 11: Distribution of the Bit Error Rate, BER Results Computed from the Input-Output Relationship of the DNA P-Box with the Influence of Random Keys.

Comparison has also been made to the Key Sensitivity assessment from the permutation table utilized by other algorithms. The average Correlation Coefficient, r_{pc} values and the average Bit Error Rate, BER values resulting from the examination of the 50 generated input-output pairs with the influence of random keys, from the tested DNA P-Box, and P-Boxes used in the ANU, VAYU, SFN, and MIBS cryptographic algorithms, are summarized in Table 16 below. Even under the influence of a key, the DNA P-Box demonstrates a stronger non-linear relationship compared to the permutation tables of other algorithms. Additionally, the DNA P-Box also managed to achieve the optimum BER value of 50%.

Table 16: Average Correlation Coefficient, r_{pc} values and average Bit Error Rate, BER values for DNA P-Box and P-Boxes used in the ANU, VAYU, SFN, and MIBS.

P-Box	Average r_{pc} Value	Average Bit Error Rate
DNA P-Box	0.01148	50.00%
ANU	0.01650	50.00%
VAYU	0.02566	51.75%
SFN	0.02475	49.63%
MIBS	0.03533	49.38%

4.3 Results and Analysis for Statistical Analysis

The statistical analysis performed for this research project focuses on the full round of the DNA-Inspired Symmetric Lightweight Block Cipher, which operates on 64-bit plaintext blocks using an 80-bit key, executes in 32 rounds of encryption to produce a 64-bit ciphertext. To evaluate the randomness of the generated ciphertext, the NIST Statistical Test Suite (STS) was employed. This suite comprises fifteen tests, categorized into Non-Parameterized and Parameterized selections, with significance levels set to 1%

(0.01). Ciphertext blocks generated by the DNA-Inspired Symmetric Lightweight Block Cipher were concatenated to form a continuous binary sequence for statistical testing. Nine different data categories were used to generate input pairs (plaintext and key), with each category containing 300 samples.

Table 17 summarizes the analysis results gathered for the DNA-Inspired Symmetric Lightweight Block Cipher at 1% (0.01) significance levels for all nine data categories. Cells highlighted in green show the maximum allowable number of rejected binary sequences over the number of samples evaluated for each test. Grey cells indicate that tests are not applicable for that particular data category. It is shown that ciphertexts generated from the DNA-Inspired Symmetric Lightweight Block Cipher passed all fifteen NIST randomness tests for all nine data categories.

Table 17: Number of actual rejections in full round testing of the DNA-Inspired Symmetric Lightweight Block Cipher at 1% significance level for all data categories.

	Non-Parameterized Test Selection									
	Frequency	Runs	Longest Runs of Ones	Spectral DFT	Cumulative Sums		Binary Matrix Rank		Random Excursion Variant	Random Exercusion
					Forward	Reverse				
# of max rejection	9/300	9/300	9/300	9/300	9/300	9/300	9/300	# of max rejection		
Data Category SKA	1	2	3	2	0	1	5	6/187	5	3
SPA	5	4	4	5	5	5	1	6/187	5	5
PCC	1	2	6	4	0	0	1	3/40	1	1
CBCM	4	5	1	6	4	4	5	2/26	1	1
RPRK	2	1	2	6	2	3	2	5/182	4	5
LDK	3	2	6	4	3	3	1	–	–	–
HDK	2	3	3	2	2	1	2	–	–	–
LDP	7	2	3	1	7	8	3	–	–	–
HDP	3	1	5	2	3	3	1	–	–	–

	Parameterized Test Selection							
	Block Frequency	Non-Overlapping	Overlapping	Maurer's Universal	Linear Complexity	Serial		Approximate Entropy
						$p-value1$	$p-value2$	
# of max rejection	9/300	9/300	9/300	9/300	9/300	9/300	9/300	9/300
Data Category SKA	5	8	2	4	5	1	2	3
SPA	2	8	2	5	3	6	4	6
PCC	2	9	5	0	5	0	2	2
CBCM	3	9	4	0	3	3	5	1
RPRK	2	9	4	3	4	2	1	2
LDK	5	9	–	–	6	2	2	3
HDK	0	9	–	–	3	4	3	5
LDP	6	9	–	–	4	5	2	4
HDP	6	9	–	–	3	3	1	4

4.4 Results and Analysis for Cryptanalysis Attacks

The cryptanalysis attack for both Differential Cryptanalysis and Linear Cryptanalysis performed for this research project focuses on the encryption algorithm of the DNA-Inspired Symmetric Lightweight Block Cipher, which is the Round Function, F starting from the input of the Confusion Function, S to the output of the Diffusion Function, P. In this experiment, a 64-bit input is applied to the Round Function, F of the DNA-Inspired Symmetric Lightweight Block Cipher to evaluate its propagation characteristics. For each round, input bits with a value of '1' are traced through the Confusion Function, S layer to identify the DNA S-Boxes that are activated. Subsequently, these outputs are then propagated through the Diffusion Function, P layer to observe the number of active bits (bit '1') produced at the end of each round. This procedure was repeated iteratively across successive rounds to evaluate the propagation characteristics of the algorithm.

4.4.1 Differential Cryptanalysis Results

The highest probability of difference propagation is found at the 6th round, which is 2^{-84} which is lower than 2^{-63}, with a total of thirteen active DNA S-Boxes. The number and

overall number of active S-Boxes, active bits, and the probabilities for each round are presented in Table 18 below.

Table 18: Active S-Boxes, Active Bits and Probabilities of Differential Cryptanalysis.

Rounds	Input	Output	Active S-Box		Active bits		Probability	
			Per round	Total	Input	Output	Per round	Total
0	0000000000000001	0000010000000000	0	0	1	1	1	1
1	0000010000000000	0000000200000100	1	1	1	2	2^{-7}	2^{-7}
2	0000000200000100	0001002000000002	1	2	2	3	2^{-7}	2^{-14}
3	0001002000000002	0410020000010020	2	4	3	5	2^{-14}	2^{-21}
4	0410020000010020	2100208804100200	3	7	5	8	2^{-21}	2^{-42}
5	2100208804100200	1002000421002088	3	10	8	8	2^{-21}	2^{-63}
6	1002000421002088	0020C82910020004	3	13	8	10	2^{-21}	2^{-84}

4.4.2 Linear Cryptanalysis Results

The highest probability of linear propagation is found at the 5th round, which is 2^{-46}, which is lower than 2^{-32}, with a total of ten active DNA S-Boxes. The number and overall number of active S-Boxes and active bits for each round are presented in Table 19 below. This table also presents the probabilities, biases, and correlation potentials for each round.

Table 19: Active S-Boxes, Active Bits, Probabilities, Bias, and Correlation Potentials of Linear Cryptanalysis.

Rounds	Input	Output	Active S-Box		Active bits		Probability	Bias	Corr
			Per round	Total	Input	Output			
0	0000000000000001	0000010000000000	0	0	1	1	0	0	1
1	0000010000000000	0000400000000100	1	1	1	2	0.46975	-2^{-5}	-2^{-5}
2	0000400000000100	0401000000000040	1	2	2	3	0.49609	2^{-4}	-2^{-9}
3	0401000000000040	0040000104010000	2	4	3	4	0.50049	2^{-9}	-2^{-18}
4	0040000104010000	0102020800040001	2	6	4	6	0.50000	2^{-10}	-2^{-28}
5	0102020800040001	5900011001020208	4	10	6	10	0.50000	-2^{-28}	-2^{-46}

5 Conclusion and Recommendation for Future Work

Results and analysis of the S-Box Testing for the DNA-Inspired Symmetric Lightweight Block Cipher show that the four newly constructed DNA S-Boxes demonstrate strong cryptographic characteristics comparable to those of the AES S-Box. The evaluation results confirm that these DNA S-Boxes achieve Nonlinearity (NL) within the secure range of 100 to 120, and the Algebraic Degree (AD) satisfies the recommended threshold of $AD \geq 4$, and the Differential Uniformity (DU) is within the optimal range of $2 \leq DU \leq 6$. Furthermore, the Strict Avalanche Criterion (SAC) results demonstrate their ability to produce highly unpredictable output changes in response to minimal input variations, reflecting excellent confusion properties. The DNA S-Boxes exhibit normal distribution behaviour for the Avalanche Effect property and uniform distribution for the Completeness and Strong S-Box criteria.

Results and analysis of the Avalanche Effect Testing for the DNA-Inspired Symmetric Lightweight Block Cipher show that the overall evaluation of the DNA P-Box through the Correlation Coefficient, r_{pc} and Bit Error Rate, BER, analyses demonstrate that it possesses strong diffusion and key sensitivity properties. The r_{pc} analysis indicates a weak linear relationship between input and output pairs. This distribution around the lower r_{pc} range confirms that the DNA P-Box exhibits a strong avalanche effect, where small input variations produce highly unpredictable output changes. When compared with the permutation tables of other lightweight cryptographic algorithms, the DNA P-Box

consistently showed an average r_{pc} value closer to zero, signifying superior nonlinearity and stronger diffusion characteristics. The BER indicates that approximately half of the output bits change when a single input bit is altered, which is an ideal condition for achieving strong diffusion. Comparisons with existing algorithms confirmed that the DNA P-Box achieves a BER value closer to the optimal 50%. When assessing key sensitivity, both r_{pc} and BER analyses demonstrated that the DNA P-Box maintains robust diffusion even under the influence of random keys. Compared to other P-Boxes implemented in other algorithms, the DNA P-Box exhibited a stronger non-linear correlation and maintained the optimal BER, confirming its resilience to key-related attacks and its effectiveness in ensuring ciphertext unpredictability. The combined results from the Correlation Coefficient, Bit Error Rate, and Key Sensitivity analyses validate that the DNA P-Box demonstrates excellent diffusion, nonlinearity, and key sensitivity properties.

The results from the Statistical Analysis, as specified in Section 4.3, confirm that the DNA-Inspired Symmetric Lightweight Block Cipher exhibits excellent randomness characteristics across all evaluated datasets. The ciphertexts generated by the new algorithm successfully passed all fifteen NIST randomness tests for each of the nine data categories at the 1% (0.01) significance level.

The results of the cryptanalysis evaluations confirm that the DNA-Inspired Symmetric Lightweight Block Cipher demonstrates strong resistance to both Differential Cryptanalysis and Linear Cryptanalysis. The highest probability of difference propagation was observed at the 6th round with a value of 2^{-84}, which is significantly lower than the acceptable upper limit of 2^{-63} for a 64-bit block size, involving thirteen active DNA S-Boxes. Similarly, the highest probability of linear propagation occurred at the 5th round with a value of 2^{-46}, remaining well below the secure threshold of 2^{-32}, with ten active DNA S-Boxes. These findings confirm that the algorithm effectively achieves low correlation between input and output differences, and strong nonlinearity. The low differential and linear probabilities indicate that it would require an impractically large number of plaintext–ciphertext pairs for an attacker to exploit these relationships successfully.

In conclusion, the overall findings from the security evaluations affirm that the DNA-Inspired Symmetric Lightweight Block Cipher exhibits robust cryptographic strength through its carefully designed DNA-based substitution and permutation components. Collectively, these results validate that the proposed algorithm achieves an excellent balance of confusion, diffusion, nonlinearity, and randomness, establishing it as a secure and efficient DNA-inspired symmetric block cipher suitable for modern lightweight cryptographic applications.

While this research successfully achieved its objectives and demonstrated the capability of the DNA-Inspired Symmetric Lightweight Block Cipher, several promising directions remain open for future exploration and enhancement.

i. Hardware and Lightweight Device Implementation: Future work may focus on implementing the DNA-Inspired Symmetric Lightweight Block Cipher in embedded and Internet of Things (IoT) environments to evaluate its real-time performance, power efficiency, and hardware feasibility. Deploying the algorithm on microcontroller or FPGA platforms would provide valuable insights into its energy consumption, processing latency, and resource utilization, ensuring its suitability for lightweight and resource-constrained applications.

ii. Extended Cryptanalysis Evaluation: Other than the cryptanalysis methods presented in Section 4.4, there exist various other advanced cryptanalytic techniques that can be employed to further assess the robustness of the proposed cipher. It is therefore recommended that future work include evaluating the security of the DNA-Inspired Symmetric Lightweight Block Cipher against additional forms of attacks, such as side-channel attacks (which exploit power consumption, timing, or electromagnetic

leakage) and algebraic attacks (which target mathematical structures within the cipher). In this work, cryptanalysis related to related-key attacks and side-channel attacks was not conducted. These aspects were intentionally excluded as they fall outside the defined scope of this thesis and are strongly dependent on specific implementation details and adversarial models. Furthermore, related-key and side-channel attacks generally target advanced threat models or physical implementations, which are more appropriately addressed in extended studies or practical deployment phases. However, conducting these analyses would provide a more comprehensive understanding of the algorithm's security strength against multiple attack vectors. Nevertheless, the security evaluation conducted in this thesis is considered sufficient at the design level, as comprehensive differential and linear cryptanalysis were already performed.

iii. Integration with Quantum-Resistant Frameworks: As advancements in quantum computing pose new challenges to classical cryptography, an important direction for future research lies in integrating DNA-based cryptographic methods with post-quantum frameworks. Developing hybrid cryptosystems that combine DNA-inspired operations with quantum-resistant primitives can enhance resilience against both classical and quantum adversaries, providing a robust foundation for next-generation secure communications.

iv. Expansion of DNA-Based Cryptographic Primitives: Beyond block cipher design, the DNA computing paradigm can be further extended to other cryptographic primitives such as hash functions, key scheduling algorithms, and stream ciphers. Applying DNA-inspired design principles across these domains could lead to a comprehensive suite of bio-inspired cryptographic mechanisms with unique randomness, diffusion, and parallelism properties.

Acknowledgments

This research was funded by the Ministry of Higher Education (MOHE) Malaysia under the Fundamental Research Grant Scheme (FRGS/1/2024/ICT04/USIM/02/1). The authors would like to express their gratitude to Universiti Sains Islam Malaysia (USIM) and MOHE for the support and facilities provided.

References

[1] W. E. Ahmed. A modern method for constructing the s-box of advanced encryption standard. *Applied Mathematics*, 10(4):234–244, 2019.

[2] A. J. Ashiq. Dna cryptography and information security, 2015. Infosec Institute.

[3] E. Biham and A. Shamir. Differential cryptanalysis of des-like cryptosystems. In *Advances in Cryptology — CRYPTO '90*, volume 537, pages 2–21, 1990.

[4] L. C. Nizam Chew and E. S. Ismail. S-box construction based on linear fractional transformation and permutation function. *Symmetry*, 12(5):826, 2020.

[5] J. Daemen and V. Rijmen. The rijndael block cipher: Rijndael amended version. Technical report, National Institute of Standards and Technology, 2003.

[6] M. A. Jimale, M. R. Z'aba, M. L. B. M. Kiah, M. Y. I. Idris, N. Jamil, M. S. Mohamad, and M. S. Rohmad. Authenticated encryption schemes: A systematic review. *IEEE Access*, 10:14739–14766, 2022.

[7] S. Kavut and M. D. Yucel. On some cryptographic properties of rijndael. In *Information Assurance in Computer Networks*, volume 2052, pages 300–311, 2001.

[8] P. P. Mar and K. M. Latt. New analysis methods on strict avalanche criterion of s-boxes. *World Academy of Science, Engineering and Technology*, 48:150–154, 2008.

[9] M. Matsui. Linear cryptanalysis method for des cipher. In *Advances in Cryptology — EUROCRYPT '93*, volume 765, pages 386–397, 1993.

[10] M. Matsui. The first experimental cryptanalysis of the data encryption standard. In *Advances in Cryptology — CRYPTO '94*, volume 839, pages 1–11, 1994.

[11] National Institute of Standards and Technology. Advanced encryption standard (aes). Technical report, U.S. Department of Commerce, 2001.

[12] A. Salam, D. R. I. M. Setiadi, E. H. Rachmawanto, and C. A. Sari. Shiftmod cipher: A symmetrical cryptosystem scheme. In *International Seminar on Application for Technology of Information and Communication*, pages 1–5, 2019.

[13] A. I. Salam, E. S. M. El-Rabaie, and O. S. Faragallah. Cabac-based selective encryption for hevc using rc6 in different operation modes. *Multimedia Tools and Applications*, 77(21):28395–28416, 2018.

[14] S. D. Sanap and V. More. Performance analysis of encryption techniques based on avalanche effect and strict avalanche criterion. In *3rd International Conference on Signal Processing and Communication (ICSPC)*, pages 676–679, 2021.

[15] J. Soto. Statistical testing of random number generators. In *Proceedings of the 22nd National Information Systems Security Conference*, pages 1–12, 1999.

[16] J. Soto and L. Bassham. Randomness testing of the advanced encryption standard finalist candidates. Technical report, NIST, 2000.

[17] A. F. Webster and S. E. Tavares. On the design of s-boxes. In *Advances in Cryptology — CRYPTO '85*, volume 218, pages 523–534, 1985.

[18] A. A. Zakaria, A. H. Azni, F. Ridzuan, N. H. Zakaria, and M. Daud. Extended rectangle algorithm using 3d bit rotation to propose a new lightweight block cipher for iot. *IEEE Access*, 8:198646–198658, 2020.

[19] A. A. Zakaria, A. H. Azni, F. Ridzuan, N. H. Zakaria, and M. Daud. Randomness analysis on rectangle block cipher. In *Cryptology and Information Security Conference*, pages 133–142, 2020.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 453–479.

Post-Quantum Cryptography Audit and Compliance Frameworks: A Systematic Literature Review and Implications for Malaysia's National PQC Migration Governance

Muhammad Sufyan Basri[1], Noorul Halimin Mansol[1], Solahuddin Shamsuddin[1], Nor Effah Aina Safari[1] and Faizura Haneem Mohamed Ali[2]

[1] Malaysia Cryptology Technology & Management Centre (PTPKM), Universiti Putra Malaysia (UPM), Selangor, Malaysia
{sufyanbasri,noorulhalimin,solahuddin,effahaina}@upm.edu.my
[2] Ministry of Economy, Putrajaya, Malaysia
haneem.ali@ekonomi.gov.my

Abstract. The study scrutinises governance frameworks for the transition to post-quantum cryptography and the associated measures. Cryptographic asset inventory, maturity assessment for the migration, and compliance checks in critical sectors have been assessed. We carried out a systematic review of audit, compliance, maturity, and governance frameworks related to PQC published from January 2017 to December 2025. The process followed the PRISMA 2020 reporting standard and Kitchenham–Charters SLR guidelines, searching three bibliographic databases; authoritative grey literature informs the background and the jurisdictional comparison but contributed no primary study. Only 18 primary studies met the inclusion criteria, and none predates 2022 — a finding in its own right, given the volume of post-quantum research published over the same period. The governance layer of the PQC transition is markedly thinner than the algorithmic layer. The included studies fall into three instrument types: migration and process frameworks ($n = 8$), assessment, audit and compliance instruments ($n = 6$), and composite or sector/national instruments ($n = 4$). Coverage across ten audit dimensions is uneven. Alignment with international standards is common (61%), but maturity assessment and threat intelligence integration each appear in a single study (6%), cryptographic sovereignty in two (11%), and budget and resource planning in three (17%). Half the corpus reports no empirical validation of the instrument it proposes, and no study reports a multi-organisation evaluation. This research analyses national governance instruments in seven jurisdictions and derives ten evidence-based implications for Malaysia's PQC governance, addressing IKPK, CED No. 9, MySEAL, MyCV, and PKTN.

Keywords: post-quantum cryptography · PQC audit framework · compliance governance · national PQC migration · systematic literature review · Malaysia NCII

1 Introduction

1.1 Malaysia's National PQC Governance Ecosystem

Malaysia has developed a comprehensive and legally grounded national PQC governance architecture that, as the comparative analysis in Section 4.5 demonstrates, is distinctive among the seven jurisdictions reviewed. The Cyber Security Act 2024 (Act 854) [1],

gazetted in 2024, provides the legislative basis for national cybersecurity governance and defines eleven National Critical Information Infrastructure (NCII) sectors subject to mandatory audit and compliance requirements.

The National Cyber Security Agency (NACSA) Chief Executive Directive No. 9 (CED No. 9) [2] operationalises PQC migration under Act 854. It requires NCII entities, upon notification, to submit a cryptographic asset inventory (CBOM/SBOM), conduct a risk assessment, and develop a migration plan within three months. This is the most operationally specific national compliance directive in the reviewed literature.

PTPKM (Pusat Teknologi dan Pengurusan Kriptologi Malaysia) [4] was set up in 2025 through a partnership between NACSA and Universiti Putra Malaysia (UPM). It brings together experts from UPM, Universiti Sains Malaysia, Universiti Teknikal Malaysia, Universiti Teknologi MARA, Universiti Malaysia Sabah, and CyberSecurity Malaysia (CSM). PTPKM has eight departments that focus on governance, technical research and development, and building capacity for the national PQC transition.

1.2 The Quantum Cryptographic Threat

The quantum threat to conventional public-key cryptography has driven a major international standardisation response. In August 2024, the National Institute of Standards and Technology (NIST) published three Federal Information Processing Standards (FIPS): FIPS 203 (ML-KEM) [11], FIPS 204 (ML-DSA) [12], and FIPS 205 (SLH-DSA) [13]. A fifth algorithm, HQC, followed [14], adding a code-based option alongside the lattice-based FIPS suite. On the standardisation side the work is largely finished; what is not finished is the question of whether organisations are doing the migration these standards assume.

The standardisation urgency stems from Shor's algorithm [39], which demonstrated that integer factorisation and discrete logarithms can be solved in polynomial time on a sufficiently large quantum computer, breaking the security underpinning RSA, ECC, and Diffie-Hellman key exchange. Mosca's theorem [40] formalises the migration urgency condition: if migration time plus data shelf life exceeds the time to a cryptographically relevant quantum computer, the organisation is already at risk.

The operational urgency is heightened by the phenomenon of harvest-now-decrypt-later (HNDL) attacks. Attackers currently intercept and store encrypted information with the intention of decrypting it once quantum computing capabilities become available. This practice indicates that governments, healthcare providers, financial organizations, and operators of critical infrastructure who depend on long-term sensitive data are already exposed to quantum threats, rather than facing them solely in the future. This concept has been openly acknowledged by institutions such as the US National Security Agency (NSA) [24], the UK National Cyber Security Centre (NCSC) [21], and associated Five Eyes organisations.

1.3 Research Problem

Malaysia has established a national PQC governance structure with multiple instruments but a major evidence gap exists. No systematic review of global literature on PQC auditing and compliance frameworks has been carried out. Two practical consequences follow. First, framework designers at PTPKM and NACSA have no published evidence base against which to benchmark Malaysian instruments, identify gaps in their coverage, or prioritise development effort. Second, claims about the distinctiveness of Malaysian instruments cannot be substantiated without such a base. The theoretical motivation for this review is also important. While PQC governance literature has grown rapidly since 2022, the lack of a synthesising effort means that the field may produce overlapping and/or invalidated frameworks. This review addresses both the practical and theoretical concerns by providing the first systematic synthesis of the PQC audit and compliance framework literature.

1.4 Paper Contributions

This paper offers two main contributions. First, it provides the first systematic literature review that focuses on PQC audit, compliance, maturity, and governance frameworks. Earlier reviews concentrate on PQC algorithm performance, implementation difficulty, or application-specific deployments; this review instead synthesises the governance layer — how migration is audited, what evidence standards apply, and how compliance is enforced. Second, it derives ten evidence-based implications for Malaysia's national PQC audit framework, each grounded in identified literature gaps and targeted at specific Malaysian governance instruments.

1.5 Research Questions

This systematic literature review (SLR) addresses five research questions:

RQ1: What PQC audit and compliance frameworks have been proposed, evaluated, or implemented in the literature between 2017 and 2025?

RQ2: What dimensions do these frameworks address, and which are consistently absent or underrepresented?

RQ3: What research methodologies and validation approaches have been used to develop or evaluate these frameworks?

RQ4: What national and sector-specific PQC governance instruments are used around the world, and how do they compare with Malaysia's NACSA/PTPKM ecosystem?

RQ5: What evidence-based gaps and recommendations should guide the design and improvement of Malaysia's national PQC audit framework?

1.6 Paper Organisation

Section 2 provides background on PQC standardisation, the conceptual distinction between migration frameworks and audit frameworks, Malaysia's governance ecosystem, and related reviews. Section 3 describes the SLR methodology. Section 4 presents results across the five research questions. Sections 5 and 6 discuss findings and derive evidence-based implications for Malaysia. Sections 7 and 8 address limitations and conclusions.

2 Background

2.1 PQC Threat and Standardisation Landscape

The quantum threat to public-key cryptography is well established [41, 42]. RSA and ECC derive their security from the hardness of integer factorisation and discrete logarithm problems—problems that Shor's algorithm [39] renders tractable on a sufficiently large quantum computer. Mosca's theorem [40] formalises the migration urgency condition: $X + Y > Z$ places an organisation at risk, where X is the data shelf life, Y is the migration time, and Z is the time until a cryptographically relevant quantum computer exists.

NIST's post-quantum cryptography standardisation project, launched in 2016, concluded its first phase in August 2024 with FIPS 203 (ML-KEM) [11], FIPS 204 (ML-DSA) [12], and FIPS 205 (SLH-DSA) [13]. HQC was subsequently selected as a fifth algorithm [14], providing a code-based alternative to the lattice-dominant FIPS suite. ETSI [16, 17], ISO/IEC [18, 19], and national bodies in the US [15, 20, 24, 25], UK [21], EU [22], Canada [23], Australia [26], and Singapore [27] have issued complementary migration guidance.

2.2 PQC Migration Frameworks — Global Overview

Several national and international bodies have published PQC migration guidance. The NIST NCCoE's Migration to Post-Quantum Cryptography project [20] released a series of practice guides and playbooks. The EU NIS Cooperation Group published a coordinated implementation roadmap [22]. The UK NCSC issued migration timelines [21]; Canada's CSE published ITSM.40.001 [23].

These are migration tools, not audit and compliance mechanisms. They specify what should be done (sequencing activities, selecting algorithms, prioritising systems) but not how an independent auditor should verify that an organisation has done it, what evidence standards apply, or how non-compliance should be handled. The distinction is consequential for Malaysia's regulatory model, which requires both a migration framework and an audit and compliance capability.

2.3 PQC Audit and Compliance — Conceptual Landscape

The literature contains three instrument types that are often conflated. Migration frameworks define pathways. Maturity models assess organisational capability on a defined scale. Audit frameworks set out the criteria that an independent auditor uses to check if standards are met. There is a big difference between a maturity assessment and a formal audit. A maturity model shows where an organisation stands on a scale of capability, but it does not say what evidence the auditor should look at, what needs to be fixed, or what happens if standards are not met.

2.4 Malaysia's National PQC Governance Ecosystem — Technical Review

Malaysia's national post-quantum cryptography (PQC) governance framework is notable for its comprehensive scope and strong legal foundation.

The Cyber Security Act 2024 (Act 854) [1] establishes eleven National Critical Information Infrastructure (NCII) sectors: Government, Banking and Finance, Transportation, Defence and National Security, Information and Communications Technology (ICT), Health, Water and Waste Management, Energy, Agricultural Food and Beverage, Trade and Industry, and Other Strategic Services. The Act mandates audit and compliance requirements for these sectors and designates the National Cyber Security Agency (NACSA) as the authority responsible for conducting audits.

CED No. 9 [2], issued under Act 854, provides detailed procedures for PQC migration. It requires NCII entities to compile a cryptographic asset inventory, including a Cryptographic Bill of Materials (CBOM) and Software Bill of Materials (SBOM), conduct a risk assessment, and submit a migration plan within three months of notification. This directive represents the most operationally specific national compliance mandate among the documents reviewed.

Established in 2025, the PTPKM [4] acts as the national centre for cryptology technology and management. It houses the IKPK (Indeks Kematangan dan Pematuhan Kebangsaan — PQC Migration National Compliance and Maturity Index) [4], a seven-component framework comprising: cryptographic asset evaluation, policy and governing management, technical maturity, workforce competence, secure supply chain, incident management, and sovereignty.

The PQC Audit and Compliance Guidelines [5], produced by PTPKM, sets out eight audit criteria that extend IKPK's seven components, adding Cryptographic Incident Management — a criterion absent from all 18 primary studies reviewed here.

The MySEAL [6] produced by CSM identifies the national cryptographic algorithm list as a trusted cryptographic algorithm list for deployment in all NCII entities. Meanwhile the

MyCV [7] is the Malaysian Cryptography Validation programme, developed in accordance with ISO/IEC 19790:2012 [18] and ISO/IEC 24759:2017 [19], and serves as Malaysia's cryptographic module certification scheme. Furthermore, the PKTN [8] is established as Malaysia's trusted cryptographic product certification programme.

The National PQC Migration Plan 2025–2030 [9], officially launched during ICE-PQC 2025, sets out a five-year strategy of staggered migration across all NCII domains, coordinated by PTPKM, NACSA, CyberSecurity Malaysia, and the regulator for each sector. Moreover, to promote innovative approaches for PQC implementation in an organised and safe environment, the PQC Proof-of-Concept Sandbox Initiative [10] is deployed.

2.5 Related Reviews

Previous systematic reviews of post-quantum cryptography (PQC) have primarily focused on algorithm performance, implementation challenges, or application-specific contexts [59, 60, 61]. Bibliometric analyses of PQC research [60] indicate rapid expansion within the field; however, these studies have not addressed governance, audit, or compliance considerations. To date, no systematic literature review (SLR) has specifically examined PQC audit and compliance frameworks, thereby identifying a gap that this study aims to address.

3 Systematic Review Methodology

3.1 SLR Protocol

The PRISMA 2020 reporting guideline [36] and the Kitchenham and Charters [37] guidelines for systematic literature reviews in software engineering and information systems were followed. The review protocol was established prior to conducting the search.

3.2 Research Questions

The five research questions presented in Section 1.5 informed all design decisions, including database selection, search string construction, inclusion and exclusion criteria, quality appraisal, and synthesis approach.

3.3 Search Strategy

3.3.1 Databases

Three electronic bibliographic databases were searched: Scopus (Elsevier), IEEE Xplore Digital Library, and ACM Digital Library. These databases were selected because they collectively index the leading journals and conferences in computer science, information security, cryptography, and governance. Scopus offers the broadest cross-disciplinary coverage, including policy and governance venues. IEEE Xplore and ACM Digital Library primarily cover engineering and computing venues where post-quantum cryptography (PQC) governance literature is concentrated. Web of Science and SpringerLink were evaluated but not included, as pilot searches using a representative subset of queries revealed over 90% overlap with Scopus results in the target domain and only marginal additional yield. This observation aligns with previous systematic literature reviews (SLRs) in information security, which found Scopus to subsume most Web of Science coverage for this subject area [61]. The source base was supplemented by authoritative grey literature, including Malaysian policy instruments [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], PQC standards and national migration instruments [11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27], maturity and audit models [28, 29, 30, 31, 32, 33, 34, 35], and SLR methodology

and quantum-threat references [36, 37, 38, 39, 40]. These grey literature sources inform the background in Section 2 and the jurisdictional comparison in Section 4.5. None was eligible as a primary study under IC1–IC6, and they therefore do not enter the PRISMA flow reported in Section 3.5.

3.3.2 Search Period

The search period spanned from January 2017 to December 2025. The selection of 2017 as the starting point aligns with NIST's December 2016 public call for post-quantum cryptography (PQC) algorithm proposals, which initiated the current standardisation cycle and prompted the emergence of governance-oriented PQC literature.

3.3.3 Search Strings

The primary search string was developed through an iterative process, following the methodology outlined by Kitchenham and Charters [37]:

> *Main string:* ("post-quantum cryptography" OR "post-quantum" OR "PQC" OR "quantum-safe cryptography" OR "quantum-resistant cryptography") AND ("audit" OR "compliance" OR "assessment" OR "framework" OR "maturity" OR "governance" OR "migration" OR "roadmap" OR "certification") AND ("organisation" OR "enterprise" OR "government" OR "critical infrastructure" OR "national" OR "sector")
>
> *Secondary string:* Main string AND ("national" OR "government" OR "critical infrastructure" OR "NCII" OR "policy")

Search queries were tailored to the syntax requirements of each database. The title, abstract, and keyword fields were examined. Only studies published in English were considered.

3.4 Inclusion and Exclusion Criteria

The inclusion and exclusion criteria were established before the search and were applied in a consistent manner at all stages of the screening process (Table 1).

Table 1: Inclusion and Exclusion Criteria for Primary Study Selection

#	Inclusion Criteria	Exclusion Criteria
IC1/EC1	Peer-reviewed journal articles, conference papers, technical reports, or standards documents	Purely algorithmic PQC papers with no audit, compliance, or governance dimension
IC2/EC2	Studies that propose, evaluate, or describe a PQC audit, compliance, maturity, or governance framework	Opinion pieces or editorials without identifiable methodology
IC3/EC3	Published between January 2017 and December 2025	Duplicates identified across databases
IC4/EC4	Available in full text	Studies not available in full text
IC5/EC5	English language	Studies not in English
IC6/EC6	Addresses PQC (not classical cryptography or QKD alone)	Papers addressing QKD exclusively with no PQC governance content

3.5 Study Selection Process (PRISMA 2020)

Study selection followed the four-stage PRISMA 2020 flow, reported in full in Figure 1.

Identification. Database searching returned 3,071 records from Scopus, IEEE Xplore, and the ACM Digital Library. Removing 1,594 duplicate records left 1,477 records for screening.

Screening. Title and abstract screening excluded 849 records, leaving 628 records for full-text retrieval.

Eligibility. Of the 628 records sought, 298 could not be retrieved or were excluded on inspection, leaving 330 full-text records for eligibility assessment. Applying the inclusion and exclusion criteria in Table 1 excluded a further 312 records. Reasons for exclusion were insufficient methodological information (96), duplicates identified at full text (83), studies devoted solely to QKD (47), full text unavailable (38), no audit, compliance, maturity, or governance framework under EC1 (36), and language (12).

Included. Application of the criteria in Table 1 at full text yielded **18 primary studies**. Exclusions at this stage arose on methodological grounds and under EC1, the latter applying where a record addressed PQC algorithms, implementations, or performance without proposing or evaluating an audit, compliance, maturity, or governance framework. Records were also excluded where the earlier of the publication year and the first-available-online year fell outside the January 2017–December 2025 window; eligibility was determined against the publisher record in every case.

3.6 Quality Assessment

Every included study was appraised against an eight-item checklist adapted from the Critical Appraisal Skills Programme (CASP) [38] and from Kitchenham's quality criteria for software engineering SLRs [37] (Table 2). The weakest items across the corpus were QA4 (has the framework been validated?) and QA5 (are validation results reported in sufficient detail?). Many frameworks state that they were "reviewed by experts" without identifying those experts, specifying what was reviewed, or reporting what the review produced. Such cases were awarded partial rather than full credit. This is a calibration decision, and borderline QA4/QA5 cases were the least clear-cut to score.

Table 2: Quality Appraisal Checklist (8-item)

QA#	Quality Criterion	Rating Scale
QA1	Does the study clearly state research objectives?	Met / Partially / Not Met
QA2	Is the methodology clearly described and reproducible?	Met / Partially / Not Met
QA3	Is the proposed framework explicitly described with defined components?	Met / Partially / Not Met
QA4	Has the framework undergone any form of validation?	Met / Partially / Not Met
QA5	Are validation results reported with sufficient detail?	Met / Partially / Not Met
QA6	Are study limitations acknowledged?	Met / Partially / Not Met
QA7	Does the study address a real-world implementation or governance context?	Met / Partially / Not Met
QA8	Does the study contribute to audit/compliance/governance dimensions?	Met / Partially / Not Met

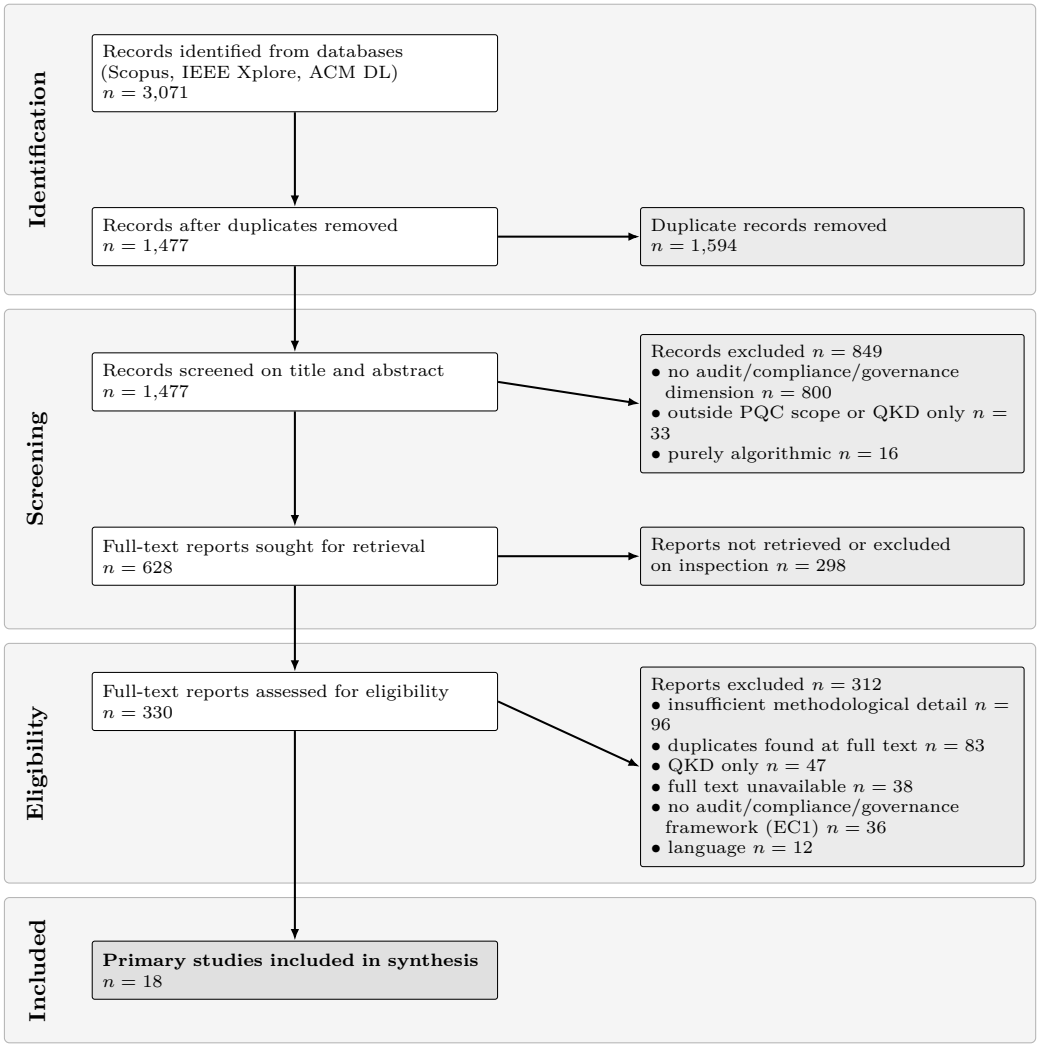


Figure 1: PRISMA 2020 flow diagram for study selection, showing the four stages of the selection process: Identification, Screening, Eligibility, and Included. Boxes on the right record exclusions at each stage with reasons.

We rated each study High (6–8 items met), Medium (4–5), or Low (fewer than 4). All 18 studies were appraised by a single reviewer against the explicit QA rubric described above. Where an item was genuinely borderline — most often QA4 and QA5 — the more conservative (lower) score was assigned. No second appraiser was available, so inter-rater reliability could not be established for this review; the implications are set out in Section 7.

3.7 Data Extraction

A structured extraction form was applied to each included study by the reviewer. Fields captured bibliographic metadata, framework name and type, primary sector, geographic context, research methodology, validation approach, QA score, and coverage of each of the ten audit dimensions (D1–D10). Table 3 summarises the extraction fields.

Table 3: Data Extraction Form Fields

Field	Description
Study ID	Unique identifier (S01–S18)
Authors	Full author list
Year	Publication year
Venue	Journal / conference / technical body
Country of Origin	Primary author's institutional country
Study Type	Conceptual / Empirical / Review / Standard
Framework Name	Name of proposed or evaluated framework
Framework Type	Type I–III classification
Dimensions Covered	Which of D1–D10 are explicitly addressed
Validation Method	Conceptual / Expert review / Case study / Survey / Delphi / Empirical
National Context	Country or jurisdiction focus
Key Findings	Summary of primary contributions
Quality Score	QA total score (0–8) and High/Medium/Low rating

3.8 Data Synthesis

Data were synthesised using five approaches, one per research question. RQ1 was addressed through descriptive thematic typing, with instruments inductively grouped into three types. RQ2 used frequency analysis of dimension coverage across all 18 studies. RQ3 summarised the distribution of research methodologies and quality ratings. RQ4 used a structured comparison matrix of national governance instruments. RQ5 identified recurrent structural gaps through thematic synthesis and organised them into the evidence-based implications set out in Section 6.

4 Results

4.1 Search Results and Study Characteristics

A database search identified 3,071 records. After removing 1,594 duplicates, 1,477 records remained for title and abstract screening. Of these, 849 were excluded: 800 did not address audit, compliance, or governance; 33 were outside the scope of post-quantum cryptography (PQC) or focused exclusively on quantum key distribution (QKD); and 16 were purely algorithmic. The remaining 628 records underwent full-text retrieval. Of these, 298 could not be retrieved or were excluded upon inspection, resulting in 330 records for eligibility assessment. Application of the criteria in Table 1 to the full texts excluded an additional 312 records, resulting in 18 primary studies.

Two characteristics of the corpus should be noted at the outset, as they influence all subsequent findings. First, the corpus is limited in size. Eighteen studies constitute a modest evidence base for a field that has received considerable research attention. This limitation arises because most PQC publications focus on algorithms, implementations, and performance, rather than governance. Studies were excluded at full text not because they were weak but because they did not propose, evaluate, or describe an audit, compliance, maturity, or governance framework as required by IC2. The thinness of this literature is itself a principal finding of the review.

Second, the corpus is recent. No included study predates 2022. One study was published in 2022 (6%), two in 2023 (11%), six in 2024 (33%), and nine in 2025 (50%) — so 83% of the corpus appeared in the final two years of the review window (Figure 2). The 2017–2021 period, which accounts for five of the nine years searched, contributes no eligible study

at all. Governance-oriented PQC research is therefore not merely growing; it begins only after the NIST algorithm selections of 2022.

Publication venues are dominated by conference proceedings: 13 studies (72%) appeared in conference or workshop proceedings, 4 (22%) in peer-reviewed journals, and 1 (6%) is an arXiv preprint retained for its sector-specific contribution. The scarcity of journal-length treatments is consistent with a field at an early stage, in which position papers and short conference contributions predominate over mature, extensively validated work.

By primary author institution, the United States accounts for the largest single share ($n = 4$, 22%), followed by Germany, China, and Canada ($n = 2$ each, 11%), with the Netherlands, Spain, Italy, Estonia, Bulgaria, Brazil, Colombia, and a joint Malaysia–Saudi Arabia contribution supplying one study each. Taken together, EU member states account for 7 studies (39%). Country of origin was assigned by the primary or corresponding author's institutional affiliation at the time of publication.

On quality, 7 studies (39%) rated High, 8 (44%) Medium, and 3 (17%) Low. As in the earlier literature, the weakest items were QA4 and QA5: 9 studies (50%) scored only "Partially Met" or "Not Met" on whether the proposed framework had been validated and whether validation results were reported in sufficient detail.

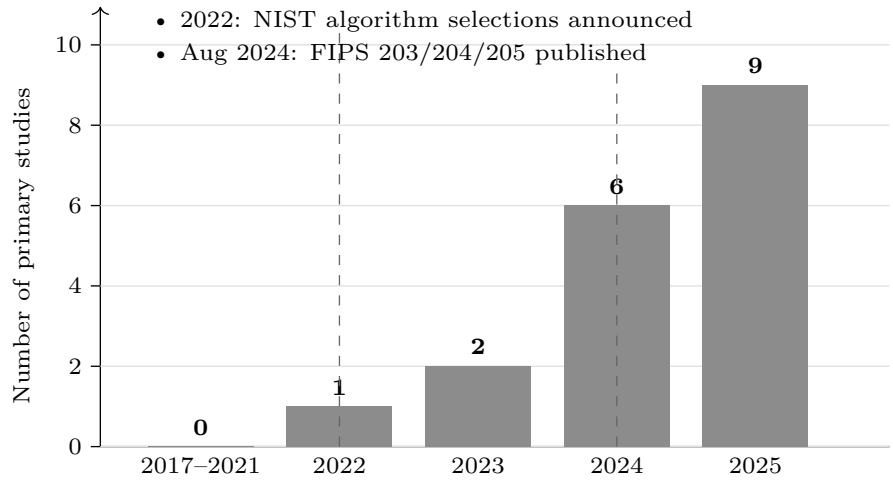


Figure 2: Annual distribution of the 18 primary studies across the review period. The 2017–2021 window, covering five of the nine years searched, contributes no eligible study. Governance-oriented PQC literature begins in 2022, following the NIST algorithm selections, and 83% of the corpus appears in 2024–2025.

4.2 RQ1 — Frameworks Identified

The 18 primary studies were classified inductively into three instrument types (Table 4). **Type I — Migration and Process Frameworks** ($n = 8$, 44%) define a pathway: a sequence of activities, phases, or deployment approaches by which an organisation moves from classical to post-quantum cryptography. **Type II — Assessment, Audit and Compliance Instruments** ($n = 6$, 33%) define criteria by which a state of affairs is evaluated, verified, or measured, rather than a route to be followed. **Type III — Composite and Sector/National Instruments** ($n = 4$, 22%) either integrate two or more functions into a single architecturally unified instrument or operate at sector or national scale.

The three types partition the corpus: every study carries exactly one type, and the totals sum to 18.

This typology is a revision of the finer five-category scheme used in earlier drafts of this review. On a corpus of this size that scheme proved unusable: maturity models and national governance instruments each attracted a single member, and two governance-relevant empirical studies fitted no category at all. The three-type scheme is coarser but it is exhaustive and mutually exclusive, and it preserves the distinction that matters most for audit practice — between instruments that prescribe a route and instruments that evaluate a state.

The near-absence of maturity models is a finding rather than an artefact of the classification. Only one study in the corpus, PMMP [44], incorporates a maturity dimension, and it does so by importing the externally developed Crypto Agility Maturity Model rather than by constructing its own. No study in the corpus proposes an original PQC maturity model with a defined capability scale. This bears directly on Malaysia's IKPK, which is precisely such an instrument, and is taken up in Section 6.

A second observation concerns the balance between the types. Type I outnumbers Type II by eight to six, and within Type II only three studies — S03 [45], S04 [46], and S13 [64] — define anything resembling audit criteria against which an independent assessor could form a judgement. The literature is markedly better developed on *how to migrate* than on *how to verify that migration has occurred*, which is the asymmetry this review was designed to test.

4.3 RQ2 — Dimension Coverage Analysis

To investigate coverage across the corpus, ten audit dimensions were derived inductively and defined. D1 (Migration Phases): whether the framework covers phases of the migration process. D2 (Maturity Assessment): whether it provides maturity or capability scoring. D3 (Regulatory/Compliance Alignment): whether it aligns with legal or regulatory requirements. D4 (Sector Specificity): whether recommendations are tailored to specific critical national infrastructure sectors. D5 (Supply Chain/SBOM/CBOM): whether cryptographic and software supply-chain risks are addressed. D6 (Workforce and Awareness): whether training, competency, and awareness are covered. D7 (International Standards Alignment): whether the framework aligns with ISO, ETSI, NIST, or equivalent standards. D8 (Threat Intelligence Integration): whether threat intelligence informs the audit or compliance process. D9 (Budget and Resource Planning): whether budget and resource planning is addressed. D10 (Cryptographic Sovereignty): whether national cryptographic autonomy, trusted product lists, or algorithm validation is considered.

Coverage was coded as explicit, partial, or absent. Figure 3 reports explicit coverage; partial coverage is given in Appendix A. The exclusive reporting of explicit coverage represents a conservative methodological choice and is consistently applied throughout the analysis.

D7 (International Standards Alignment) is the most consistently addressed dimension, appearing explicitly in 61% (11/18) of studies. This prevalence is expected, as alignment with the NIST suite serves as a common reference point for the field. D3 (Regulatory Alignment) is addressed in 39% (7/18) of studies. D1 (Migration Phases) and D4 (Sector Specificity) are each present in 33% (6/18) of studies, while D5 (Supply Chain) and D6 (Workforce) are each addressed in 22% (4/18) of studies.

Three dimensions are notably underrepresented, and this pattern diverges from trends observed in the broader post-quantum cryptography (PQC) literature. D9 (Budget and Resource Planning) appears explicitly in 17% (3/18) of studies: in S01 [43], which treats funding as a policy instrument; in S02 [44], which requires management to allocate resources for training and external support; and in S12 [63], which specifies a fund-use plan linked to an implementation schedule. D10 (Cryptographic Sovereignty) appears in 11% (2/18), in S12 [63] and S17 [56], both Chinese. D2 (Maturity Assessment) and D8 (Threat Intelligence

Table 4: Primary studies by instrument type. Dimensions listed are those with explicit (not partial) coverage; the complete coding, including partial coverage, is given in Appendix A.

ID	Framework / Instrument	Yr	Source	Key Dims	Validation
Type I — Migration and Process Frameworks (n = 8)					
S02	PMMP Migration Management Process	2024	von Nethen et al. [44]	D1, D2, D5, D6, D7, D9	Expert review
S05	Organisational PQC transition model	2022	Joseph et al. [47]	D1, D7	Conceptual
S06	ISO 15489 / NIST PQC lifecycle model	2025	Santa Quintero & Meneses [48]	D1, D3, D7	Case study
S09	Beyond-algorithm-replacement migration	2024	Giron [53]	D6, D7	Conceptual
S14	Hybrid crypto-agility framework	2025	Ivanov et al. [50]	D5, D7	Conceptual
S15	TCPS post-quantum transition	2024	Mamun et al. [51]	D4, D7	Conceptual
S16	Embedded deployment approach	2024	Muzikant & Willemson [54]	D7	Case study
S18	QuantumCharge (ISO 15118)	2023	Kern et al. [57]	D7	Case study
Type II — Assessment, Audit and Compliance Instruments (n = 6)					
S01	QS transition policy assessment	2023	Kong et al. [43]	D3, D4, D6, D7, D9	Empirical
S03	PQC-OAV evidential assurance	2025	Mills & Ghorbani [45]	D8	Empirical
S04	Quantum-readiness of standards	2025	Chouhan et al. [46]	D3, D4, D7	Empirical
S07	Quartz PQC-readiness observability	2025	Kundu & Kompella [49]	D5	Empirical
S10	PQC Network Instrument	2024	Sowa et al. [55]	D7	Empirical
S13	Proportional PQC regulatory analysis	2025	Gagliardi & D'Elia [64]	D3, D4	Legal-doctrinal
Type III — Composite and Sector/National Instruments (n = 4)					
S08	PQ-REACT framework	2024	García-Cid et al. [52]	D1, D5	Case study
S11	AES post-quantum integration roadmap	2025	Ganesh et al. [62]	D1, D3, D4, D7	Empirical
S12	Power sector protection framework	2025	Huang & Pan [63]	D1, D3, D4, D6, D9, D10	Institutional
S17	5G PQC application planning	2025	Lu et al. [56]	D1, D3, D4, D5, D7, D10	Institutional

Integration) are the least covered dimensions in the corpus, each appearing explicitly in a single study (6%, 1/18).

The D2 result deserves emphasis because it inverts a common assumption. Maturity models are frequently described as a well-populated part of the PQC governance landscape, but within a corpus restricted to audit and compliance frameworks only S02 [44] incorporates a maturity dimension, and it does so by adopting the external CAMM model rather than developing one. The D8 result is similarly concentrated: only S03 [45] integrates operational threat-intelligence feeds — IP reputation, asset databases, and threat-intelligence sources — into an assurance pipeline. Elsewhere in the corpus, threat intelligence appears only as background discussion of harvest-now-decrypt-later risk, which was coded as partial rather than explicit coverage.

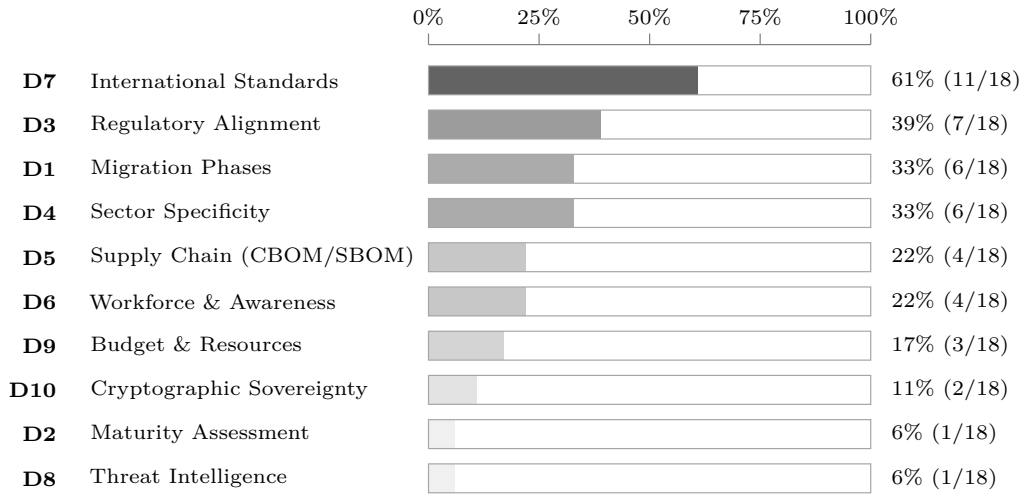


Figure 3: Explicit coverage of the ten audit dimensions across the 18 included studies (RQ2).

4.4 RQ3 — Methodologies Used

The 18 included studies used the following research methodologies: conceptual design ($n = 5$, 28%), in which a framework is proposed without empirical testing; empirical measurement, prototyping, or benchmarking ($n = 5$, 28%); case study or pilot deployment ($n = 4$, 22%); institutional or documentary analysis ($n = 3$, 17%); and structured expert review ($n = 1$, 6%). One study, S13 [64], employs legal-doctrinal analysis and is the only such contribution in the corpus.

Nine studies (50%) report some form of empirical validation, defined here as case study, pilot deployment, prototype evaluation, structured expert consultation with a documented protocol, or empirical measurement. The remaining nine (50%) rely on conceptual design, institutional description, or doctrinal argument alone. Within the empirically validated group, the evidence is uneven: several studies evaluate a technical prototype rather than the governance framework the prototype is intended to support, so the proportion of studies in which a *governance* instrument has been operationally tested is lower still.

This has direct consequences for the credibility of PQC audit governance. Half the instruments in this literature have never been applied in an operational setting, and the strongest evidence available for any of them is a single-organisation pilot. No study in the corpus reports a multi-organisation or sector-wide evaluation of a PQC audit or compliance framework. This underlines the value of empirically validating IKPK and the PQC Audit and Compliance Guidelines, addressed in Implication 3.

4.5 RQ4 — National Governance Instruments Comparison

Table 5 compares the PQC governance instruments of seven jurisdictions, drawn from the academic and grey literature and assessed against seven governance dimensions: Legal Foundation, Phase Model, Maturity/Audit Criteria, National Algorithm List, Cryptographic Validation Scheme, Trusted Product Certification, and CBOM/SBOM Mandate. A dimension is recorded as satisfied only where a jurisdiction has published an instrument that is (i) specific to post-quantum migration or explicitly extended to cover it, and

(ii) applicable across the jurisdiction's critical infrastructure sectors rather than to a single sector or classification level.

Three patterns emerge. First, coverage is heavily weighted towards the front of the governance pipeline. All seven jurisdictions have published a phase model or migration roadmap, and all seven have some legal or policy foundation, but only three (USA, EU, Malaysia) pair this with a cryptographic validation scheme, and only two (EU, Malaysia) with a trusted product certification programme that is both operational and PQC-scoped. While guidance on what to migrate is widely available, mechanisms for verifying the completion of migration are comparatively scarce.

Second, jurisdictions can be grouped into two governance models. The United Kingdom, Canada, Australia, and Singapore utilize advisory regimes, recommending post-quantum cryptography (PQC) migration through agency guidance without binding statutory instruments or maturity indices. In contrast, the United States, European Union, and Malaysia implement mandate-backed regimes, each based on distinct legal frameworks: the United States relies on executive orders and procurement conditions primarily for national security systems; the European Union enforces NIS2 and eIDAS 2.0 obligations, which are unevenly transposed across member states; and Malaysia employs a single statutory instrument (Act 854) supported by a directive (CED No. 9).

Third, the most pronounced divergence appears in the final comparison criterion. Only Malaysia enforces a mandatory requirement for CBOM/SBOM submission with a specified deadline. Executive Order 14028 and NIS2 introduce partial or sector-specific SBOM obligations, but do not mandate a cryptographic bill of materials. The other four jurisdictions do not impose such requirements. This aspect of the comparison is the least ambiguous and, in light of Gap G4, also highlights that Malaysia's leadership in this area is not strongly supported by published technical specifications.

Table 5: Comparative Analysis of National PQC Governance Instruments. A dimension is considered satisfied only if the jurisdiction has published an instrument that is specific to post-quantum migration, or has been explicitly extended to address it, and is applicable across all critical infrastructure sectors rather than limited to a single sector or classification level. The notation "ltd" indicates the existence of an instrument that does not fulfill both criteria.

Country	Legal Foundation	Phase Model	Maturity/ Audit	Algo List	Valid. Scheme	Product Cert.	CBOM Mandate
USA	US Quantum Act 2022; EO 14028	NIST NCCoE Playbook	CSF 2.0 overlay (partial)	CNSA 2.0 (NSS only)	CMVP/ CAVP	N/A	EO 14028 (partial)
UK	NCSC guidance (no PQC Act)	NCSC 3-phase (2025)	No formal index	N/A	CESG/ NCSC (ltd)	CAPS (ltd)	None
EU	NIS2; eIDAS 2.0	NIS CG Roadmap (2025)	ENISA guidance (non-binding)	No unified list	EUCC	EUCC cert	NIS2 (partial)
Canada	Directive on Security Mgmt	ITSM. 40.001 (2025)	No formal index	N/A	CSEC CSTC (ltd)	N/A	None
Australia	PSPF; ASD ISM	ASD ISM guidance	ASD E8 (ltd PQC)	ASD ISM list	ASD/ ACSC (ltd)	AISEP (ltd)	None
Singapore	Cybersecurity Act 2018	NQSN+ (2024)	NQSN+ guidance	N/A	CSA (ltd)	N/A	None
Malaysia	Act 854 (statutory) + CED No. 9	CSM 5-phase model	IKPK (7-comp.) + Audit Guidelines	MySEAL (all sectors)	MyCV (ISO/ IEC 19790/ 24759)	PKTN	**Mandatory** (3-month window)

4.6 RQ5 — Evidence-Based Gaps from the Literature

A review of the 18 included studies identifies five recurrent structural gaps within the literature on post-quantum cryptography (PQC) audit and compliance.

Gap G1: Underrepresented Dimensions. Four dimensions receive less than 20% explicit coverage. Maturity assessment (D2) and threat intelligence integration (D8) are each addressed in only one study. Budget and resource planning (D9) is discussed in three studies, while cryptographic sovereignty (D10) appears in two. No study in the corpus proposes an original PQC maturity model with a defined capability scale, and only one integrates operational threat-intelligence feeds into an assurance process. D10 is documented exclusively in the two Chinese studies, S12 [63] and S17 [56]; no other jurisdiction represented in the corpus has published an equivalent.

Gap G2 — Limited Empirical Validation: Half the studies in the corpus (9/18) report no empirical validation of the instrument they propose, relying instead on conceptual design, institutional description, or doctrinal argument. Among the nine that do report validation, the strongest available evidence is a single-organisation pilot or prototype evaluation. No study reports a multi-organisation or sector-wide evaluation of a PQC audit or compliance framework. The evidence base for PQC audit governance is therefore narrow, and the narrowness becomes consequential once instruments built on this literature are applied to critical infrastructure at national scale.

Gap G3 — Sector-Specificity Deficit: Only 33% (6/18) of studies address sector-specific calibration (D4), and those that do concentrate in a small number of sectors — energy (S12 [63]), telecommunications (S17 [56]), transportation (S15 [51]), and finance (S13 [64]). Health, water, government, and defence are not addressed by any study in the corpus. Most instruments apply generic controls that do not distinguish between, for example, an energy grid operator and a hospital system.

Gap G4 — CBOM/SBOM Operationalisation: Four studies address cryptographic supply-chain inventory explicitly, and S07 [49] demonstrates automated CBOM generation through observability tooling. None, however, provides a standardised technical specification for CBOM content, data schema, tooling integration, or submission format suitable for national regulatory compliance. The gap is not conceptual — the value of a CBOM is well understood — but specificational.

Gap G5 — Phase-Maturity Integration: No study in the corpus integrates maturity thresholds as formal phase-gate requirements. This is a stronger finding than it was in earlier readings of this literature: since only one study incorporates a maturity dimension at all, the question of gating phase progression on demonstrated maturity is not merely unanswered but largely unposed. Phase progression consequently remains a planning artefact rather than a governed milestone with measurable completion criteria.

Gaps G1 and G5 are related and should be read together. The scarcity of maturity instruments (D2, 6%) is the upstream cause of the absence of phase-gating: a framework cannot gate progression on maturity thresholds it does not define.

5 Discussion

5.1 State of PQC Audit and Compliance Framework Development

The 18 primary studies present a field that is early rather than merely uneven. The most consequential characteristic of this literature is its size: with a literature search covering a nine years period, and against a very large volume of PQC publication overall, only 18 studies propose, evaluate, or describe an audit, compliance, maturity, or governance framework. Governance-oriented work begins in 2022 and accelerates after the publication of FIPS 203, 204, and 205 in August 2024, with 83% of the corpus appearing in 2024–2025.

Persistent heterogeneity in scope, validation quality, and dimension coverage indicates that the field has not yet established common audit standards or a shared evidence base.

Migration and process frameworks (Type I, $n = 8$) form the largest group and are also the least validated: five of the eight rest on conceptual argument alone. Assessment, audit and compliance instruments (Type II, $n = 6$) are better evidenced — five of the six report empirical work — but only three define criteria against which an independent assessor could form a judgement. Composite, sector, or national instruments (Type III, $n = 4$) offer the most extensive dimension coverage in the corpus. This finding indicates that comprehensiveness results from deliberate integration rather than incremental extension of single-purpose instruments. The primary substantive finding is the asymmetry between Type I and Type II: the literature is significantly more developed regarding migration processes than in verifying that migration has occurred.

5.2 The Maturity-Phase Integration Gap

A significant challenge is that maturity models and migration phase frameworks generally function independently. Maturity models assess an organisation's capability level but do not provide guidance for transitioning between phases or specify advancement requirements. Conversely, phase frameworks outline sequential steps but rarely connect progression to demonstrated maturity. Bridging this gap necessitates a model that explicitly links phase advancement to demonstrated maturity. Malaysia represents the closest example in the reviewed corpus: IKPK and the CSM five-phase model coexist within a single national programme and are, in principle, linkable. No other reviewed jurisdiction publishes both a maturity index and a phase model under unified governance. However, this link is not yet operational. Neither instrument specifies the minimum IKPK component scores required for advancement between CSM phases, so the connection remains structural rather than procedural. This residual gap is addressed by Implication 2 in Section 6 and aligns with Gap G5: no reviewed framework, Malaysian or otherwise, has formalised maturity thresholds as phase-gate criteria.

5.3 The National-versus-Organisational Level Gap

Most existing literature focuses on organisations, offering frameworks intended for self-assessment and planning by individual entities. National-level audit frameworks that mandate sector-wide compliance, define enforcement mechanisms, and establish evidence standards for independent audits are rare outside the Malaysian and, to a limited extent, US contexts. The mandatory CBOM submission requirement under CED No. 9 constitutes the most explicit national compliance directive identified in the reviewed literature.

5.4 The Cryptographic Sovereignty Gap

Cryptographic sovereignty, defined as a state's capacity to establish, verify, and approve trusted algorithms and products independently of foreign vendors or standards, is a critical yet underexplored topic in the literature. Only 11% of the corpus (2/18) addresses D10, and both studies are Chinese: S12 [63] documents commercial cryptography regulation applied to the power sector, and S17 [56] describes the national commercial cryptography standards programme and its domestic algorithm development campaign. No other jurisdiction represented in the corpus has published a comprehensive sovereignty model integrating algorithm approval, validation, and product certification within a unified national system. The distinction from Malaysia is one of scope and documentation rather than of kind: the Chinese instruments are documented sectorally or as standards-development activity, whereas Malaysia's MySEAL–MyCV–PKTN arrangement is published as a cross-sector national programme. Because the review is restricted to English-language sources, the

possibility that comparable models are documented in Chinese, Korean, Japanese, or German-language literature cannot be excluded.

5.5 The CBOM/SBOM Operationalisation Gap

The CBOM, which enumerates cryptographic assets, algorithms, libraries, key sizes, and operational modes, is considered essential for planning post-quantum cryptography (PQC) migration. However, none of the frameworks reviewed offer a standard CBOM technical specification for national compliance, a data format, tool suggestions, or a set way to submit it. CED No. 9 requires CBOM submission, but the details of what must be included have not yet been published.

5.6 Methodological Quality Concerns

Approximately half of the instruments reviewed are based on conceptual design, institutional description, or doctrinal argument rather than operational testing. None has been evaluated across multiple organisations. This limitation is significant, as audit tools lacking practical testing may not be trusted by NCII operators and regulators. Malaysia's PQC Sandbox initiative [10] provides a mechanism for the country to test these tools in a controlled environment prior to broader adoption.

5.7 Bibliometric Trends and Projection

The body of literature on PQC audit and compliance is expanding from a minimal foundation, primarily in response to standards-publication events. No eligible studies were published between 2017 and 2021; one was published in 2022, two in 2023, six in 2024, and nine in 2025. On the observed trajectory the volume of governance-oriented publication is likely to continue rising through 2026–2027 as national compliance programmes mature and empirical case studies accumulate. Given both the growth rate and the small absolute size of the current evidence base, an annual update to this review is recommended.

6 Evidence-Based Implications for Malaysia's National PQC Audit Framework

Based on the SLR evidence in Sections 4 and 5, this section derives ten evidence-based implications for PTPKM and NACSA. Each implication identifies the supporting SLR evidence, the target Malaysian instrument, and a concrete recommended action.

Implication 1: IKPK — Supply Chain and Cryptographic Sovereignty Enhancement

SLR Evidence: Gap G1 shows that supply chain (D5, 22%) and cryptographic sovereignty (D10, 11%) are among the least addressed audit dimensions in the corpus. Only four studies address cryptographic supply-chain inventory explicitly, and only two — both Chinese — treat sovereignty at all. No study combines the two. IKPK currently covers both areas only at a high level, and there is no published instrument against which to benchmark a more granular treatment.

Target Instrument: IKPK (7-component maturity index).

Recommended Action: PTPKM should enhance IKPK by adding granular supply-chain sub-components within Component 2 (Cryptographic Asset Assessment) to assess CBOM depth, third-party library dependencies, and vendor cryptographic lifecycle commitments explicitly. A new Component 8 (Cryptographic Sovereignty Compliance) should

be added to assess MySEAL algorithm adoption, MyCV validation status, and PKTN certification of deployed products.

Implication 2: Phase-Gate Integration of IKPK Maturity Thresholds

SLR Evidence: Gap G5 identifies that no reviewed instrument integrates maturity thresholds as formal phase-gate requirements. The absence is consistent across all 18 studies and is compounded by Gap G1: only one study in the corpus incorporates a maturity dimension at all, so the question of gating phase progression on maturity is largely unposed in the literature. If there is no phase gating, migration phases end up as planning artefacts instead of clear milestones with measurable completion criteria.

Target Instrument: IKPK integrated with CSM 5-phase model.

Recommended Action: PTPKM should set minimum IKPK component scores for NCII entities to move between CSM phases. For instance, completion of Phase 1 (Assess) requires IKPK Component 1 to achieve a minimum score of 3 out of 5. Initiation of Phase 2 (Select) necessitates that Components 1 through 3 each attain at least 3 out of 5. Advancement to Phase 3 (Validate) requires all seven components to score a minimum of 4 out of 5, and MyCV must validate the selected algorithms.

Implication 3: Empirical Validation Methodology for IKPK and PQC Audit Guidelines

SLR Evidence: RQ3 findings show that half the instruments reviewed report no empirical validation, and that none has been evaluated across more than one organisation (Gap G2). Audit frameworks are only scientifically credible if they show proven results in real-world situations.

Target Instrument: IKPK, PQC Audit Guidelines.

Recommended Action: PTPKM should arrange for real-world case studies to test IKPK and the PQC Audit Guidelines in at least three NCII sectors. Finance, Government, and Telecommunications are suggested because they have the highest HNDL risk. The PQC Proof-of-Concept Sandbox [10] can be used as a controlled environment for these tests. The results should be published in internationally indexed journals.

Implication 4: Malaysian CBOM Technical Specification

SLR Evidence: Gap G4 identifies the absence of a standardised CBOM technical specification for national regulatory compliance across the global literature. CED No. 9 mandates CBOM submission without a corresponding technical specification for content and format.

Target Instrument: CED No. 9, PTPKM technical documentation.

Recommended Action: PTPKM should develop and publish a Malaysian CBOM Technical Specification defining mandatory CBOM fields (algorithm name, key size, protocol context, library/version, dependency chain, operational mode), submission format (CycloneDX CBOM [58] JSON schema with Malaysian extensions), update cadence (annual minimum; event-triggered on algorithm substitution), and integration pathway with CED No. 9 submissions.

Implication 5: Cryptographic Incident Management as International Best Practice

SLR Evidence: The eighth audit criterion of the PQC Audit and Compliance Guidelines — Cryptographic Incident Management — is absent from all 18 reviewed instruments. This dimension covers detection, response, and recovery processes for cryptographic incidents, including algorithm compromise events.

Target Instrument: PQC Audit and Compliance Guidelines.

Recommended Action: PTPKM should formally document the Cryptographic Incident Management criterion as a novel audit dimension and submit it as a concept paper or best practice proposal to relevant international bodies (ISO/IEC JTC 1/SC 27, ETSI TC CYBER). Establishing this criterion in the international literature would position Malaysia as the originating jurisdiction for a globally significant governance innovation.

Implication 6: MySEAL Annual Review Cycle and HQC Inclusion

SLR Evidence: RQ2 dimension analysis shows algorithm currency—whether a framework's approved algorithm list remains aligned with evolving NIST standards—is a critical but unevenly addressed aspect of D7 (International Standards Alignment).

Target Instrument: MySEAL.

Recommended Action: PTPKM/NACSA should establish a formal MySEAL Review Cycle with two review triggers: (a) a mandatory review within 90 days of any new NIST PQC FIPS publication or significant update; and (b) an annual calendar review. The first priority for the next review cycle is inclusion of HQC [14] following NIST finalisation, maintaining MySEAL's alignment with the complete NIST PQC suite.

Implication 7: MyCV PQC Testing Readiness Statement

SLR Evidence: Reviewed literature on cryptographic validation programmes identifies pre-standardisation testing gaps as compliance bottlenecks where organisations cannot validate implementations against national schemes before commercial products become available.

Target Instrument: MyCV (Malaysian Cryptography Validation).

Recommended Action: PTPKM or CyberSecurity Malaysia should publish a MyCV PQC Testing Readiness Statement covering: testing capabilities for ML-KEM (FIPS 203), ML-DSA (FIPS 204), and SLH-DSA (FIPS 205) under ISO/IEC 19790:2012 and ISO/IEC 24759:2017; testing timeline for HQC following finalisation; interim conformance assessment approaches for NCII entities requiring validation certificates before full MyCV PQC capability is in place; and interoperability pathways between MyCV certificates and international validation mutual recognition arrangements.

Implication 8: NCII Sector Calibration Supplements for IKPK

SLR Evidence: Gap G3 shows only 33% of reviewed studies address sector-specific calibration, and that those which do cluster in energy, telecommunications, transportation and finance — leaving health, water, government and defence unaddressed by any study in the corpus. Different NCII sectors have substantially different PQC risk profiles, migration constraints, and regulatory obligations.

Target Instrument: IKPK.

Recommended Action: PTPKM should develop IKPK Sector Calibration Supplements for the four highest-risk NCII sectors: Banking and Finance (long-lived financial instruments and interbank settlement systems face the greatest HNDL exposure); Government (classified communications require sovereignty-grade algorithm alignment); Telecommunications (backbone infrastructure underpins all other sectors); and Health (long-term patient record confidentiality is directly at risk). Each supplement should specify sector-specific maturity thresholds, priority components, and sector-appropriate evidence requirements for each IKPK component.

Implication 9: International Positioning of the MySEAL-MyCV-PKTN Sovereignty Model

SLR Evidence: Section 5.4 establishes that, within the 18-study reviewed corpus, no other jurisdiction has published a documented three-pillar national cryptographic sovereignty architecture that combines, in a single coordinated national programme, an endorsed algorithm list, a national validation scheme, and a trusted product certification programme. The closest documented comparators are the two Chinese studies, S12 [63] and S17 [56], which describe commercial cryptography regulation and a national algorithm standards programme respectively, but are presented in the reviewed sources as a sector-specific deployment and as standards-development activity rather than as a cross-sector national reference architecture. The MySEAL-MyCV-PKTN model is therefore without a documented cross-sector equivalent in the reviewed corpus. This claim is bounded by the review's English-language and database scope: equivalent arrangements documented only in non-English-language sources cannot be ruled out.

Target Instrument: All Malaysian national PQC instruments.

Recommended Action: PTPKM should publish the MySEAL-MyCV-PKTN model as a reference architecture in internationally visible forums: submission of a reference paper to the IACR Cryptology ePrint Archive; presentation at ICE-PQC 2026; submission to the ASEAN CERT regional PQC working group proposing the three-pillar model as an ASEAN reference architecture; and engagement with ITU-APT on inclusion of the model in regional capacity-building guidance.

Implication 10: NACSA-PTPKM-CSM Research Agenda for Under-Researched Areas

SLR Evidence: Three areas are both highly relevant to Malaysia's national PQC governance and severely under-researched globally: national-level PQC audit index design and empirical validation; CBOM tooling for resource-constrained NCII entities; and PQC migration cost modelling for national budget planning (D9 appears in only 17% of reviewed studies, and in no study as a costed national model).

Target Instrument: NACSA, PTPKM, CyberSecurity Malaysia research collaboration.

Recommended Action: NACSA-PTPKM-CSM should establish a joint research agenda prioritising: (a) design science research on IKPK empirical validation and index calibration across NCII sectors; (b) development of an open-source CBOM collection and analysis tool aligned to CED No. 9 requirements and the CycloneDX CBOM schema, suited to resource-constrained NCII entities; and (c) a national PQC migration cost modelling study covering hardware, software, training, and operational transition costs across NCII sectors, to inform Treasury submissions and National PQC Migration Plan 2025–2030 budget planning. Results should be published in internationally indexed journals.

7 Limitations

Several limitations warrant acknowledgement.

Database and grey literature scope: Three major bibliographic databases (Scopus, IEEE Xplore, ACM Digital Library) were searched. Web of Science and SpringerLink were not included; as noted in Section 3.3, pilot searches confirmed high overlap with Scopus in the target domain. It is still possible that governance-focused studies which are listed only in these databases have been overlooked, and studies from regional conferences, non-indexed institutional repositories, or proceedings not in English might also have been missed. The national quantum-safe cryptography governance measures from Germany, South Korea, Japan, and China could be underrepresented because of the criterion of

including only English-language materials, this criterion having been kept in order to guarantee a consistent quality review.

Grey literature dependency: The main primary sources cited in this review—namely IKPK, PQC Audit Guidelines, MySEAL, MyCV, and PKTN—are government technical documents which have not yet been published in indexed peer-reviewed journals. This means that they are difficult to cite formally and are not amenable to independent external verification. The fact that the grey literature comes from a single country (Malaysia) is a consequence of the review's focus on national governance, although it is recognised as a structural limitation. The gap could be filled if PTPKM published peer-reviewed academic papers on these instruments.

Publication bias: The majority of the literature that has been published concentrates on suggesting positive or innovative frameworks, whereas studies which describe implementation failures or negative results are relatively rare, causing the frameworks under review to appear more methodologically sound than they really are.

Search cutoff and field changes: The search concluded in December 2025, so any new developments after that date—such as a possible forthcoming NIST HQC FIPS publication, new national governance tools, or upcoming PTPKM publications—are not covered. It would be useful to update this review annually.

Malaysian instrument accessibility: Most of the key instruments discussed in this review—for example, IKPK, the PQC Audit Guidelines, MySEAL, MyCV, and PKTN—are to be found in government technical documents rather than in peer-reviewed journals. This in turn makes it more difficult to cite them formally or to verify their details. The academic community would benefit from peer-reviewed publications on these instruments.

Single-rater appraisal: Quality appraisal, framework typing, and dimension classification were performed by a single reviewer. No independent second appraisal was conducted, so inter-rater reliability was not measured and the consistency of these judgements cannot be quantified. This is the most significant methodological limitation of the review. The eight-item rubric, the dimension definitions in Section 4.3, and the full study-level codings in Appendix A are reported so that readers can assess and, if they wish, re-derive every classification. Independent re-appraisal of the corpus would strengthen confidence in the coverage frequencies reported for RQ2 in particular.

8 Conclusion

This study presents the first systematic review of post-quantum cryptography (PQC) audit, compliance, maturity, and governance frameworks. It analyses 18 primary studies published between January 2017 and December 2025 — a corpus whose small size, after a nine-year search window, is itself among the review's principal findings. The review adheres to the PRISMA 2020 reporting standard and the Kitchenham–Charters systematic literature review protocol. It is positioned as both a systematic review and a position paper, offering evidence-based implications for Malaysia's national PQC governance ecosystem.

In response to Research Question 1, the review catalogued the 18 instruments across three types: migration and process frameworks ($n = 8$), assessment, audit and compliance instruments ($n = 6$), and composite or sector/national instruments ($n = 4$). The types partition the corpus. Type I outnumbers Type II, and only three studies define audit criteria against which an independent assessor could form a judgement: the literature is better developed on how to migrate than on how to verify that migration has occurred. Addressing RQ2, frequency analysis across ten audit dimensions shows international standards alignment (D7, 61%) and regulatory alignment (D3, 39%) best represented, while maturity assessment (D2) and threat intelligence integration (D8) each appear in a single study (6%), cryptographic sovereignty (D10) in two (11%), and budget planning (D9) in three (17%). Addressing RQ3, half the corpus reports no empirical validation of the instrument

it proposes, and no study reports a multi-organisation or sector-wide evaluation — a systemic weakness in the evidence base. Addressing RQ4, comparative analysis across seven jurisdictions shows Malaysia as the only jurisdiction in the reviewed corpus documented as satisfying all seven compared governance dimensions simultaneously (legal mandate, phase model, maturity and audit criteria, national algorithm list, national cryptographic validation scheme, trusted product certification, and mandatory CBOM/SBOM requirement), with CED No. 9's mandatory CBOM/SBOM directive being the most operationally specific national compliance instrument in the reviewed literature. Addressing RQ5, five structural gaps were identified: dimension underrepresentation (G1), limited empirical validation (G2), sector-specificity deficit (G3), CBOM/SBOM operationalisation gap (G4), and phase-maturity integration gap (G5).

The position component derives ten evidence-based implications for Malaysia's national PQC audit framework. Key recommendations include: enhancing IKPK with supply-chain and cryptographic sovereignty sub-components; formalising phase-gate maturity thresholds that integrate IKPK with the CSM five-phase model; empirically validating IKPK and PQC Audit Guidelines across NCII sectors using the PQC Sandbox; developing a Malaysian CBOM Technical Specification aligned with CycloneDX; establishing a formal MySEAL annual review cycle incorporating new PQC algorithm i.e. HQC; publishing a MyCV PQC Testing Readiness Statement; developing IKPK sector calibration supplements; and positioning the MySEAL-MyCV-PKTN three-pillar sovereignty model as an ASEAN reference architecture.

Three specific calls to action go to PTPKM, NACSA, and CyberSecurity Malaysia: (1) publish IKPK, PQC Audit Guidelines, MySEAL, MyCV, and PKTN as peer-reviewed academic papers to establish these instruments in the global literature, where they are currently absent despite their sophistication; (2) conduct further studies on PQC governance literature published from 2025 onward to capture the latest developments; and (3) lead an ASEAN comparative study building on this review's national governance comparison to produce an ASEAN PQC governance benchmark framework. Malaysia's MySEAL-MyCV-PKTN three-pillar sovereignty model, CED No. 9's mandatory CBOM directive, and IKPK's integrated maturity index collectively form a national PQC governance ecosystem that merits serious attention from the academic and policy communities.

Declarations

Conflict of Interest: The authors hold a professional affiliation with one of the institutions responsible for several of the national instruments discussed in this review (IKPK, the PQC Audit and Compliance Guidelines, MySEAL, MyCV, and PKTN), and a number of the recommendations in Section 6 are directed at those instruments. A full, non-anonymised disclosure has been provided to the editors in the submission cover letter. To mitigate bias, the SLR protocol fixed the inclusion and exclusion criteria and the quality appraisal rubric before data collection; all national instruments, Malaysian and otherwise, were appraised against the same eight-item CASP-adapted checklist as every other primary study. Appraisal was nonetheless performed by a single reviewer, and readers should weigh the disclosure above accordingly; the full study-level codings are reported in Appendix A so that every judgement can be independently checked.

Acknowledgements: The authors thank the reviewers of the earlier version of this manuscript, whose comments prompted the verification pass that led to the revised corpus reported here.

Data Availability: Appendix A contains the complete data extraction table for all 18 primary studies, including full dimension coding and quality appraisal scores.

Funding: No specific grant from any public, commercial, or not-for-profit funding agency supported this research.

References

[1] Malaysia. *Cyber Security Act 2024 (Act 854)*. Laws of Malaysia, Act 854. Official Gazette: 26 June 2024. https://lom.agc.gov.my

[2] NACSA. *Arahan Ketua Eksekutif No. 9: PQC Migration Directive*. National Cyber Security Agency Malaysia, 2025.

[3] CyberSecurity Malaysia. *Post-Quantum Cryptography Migration Framework*. https://mykripto.cybersecurity.my, 2024.

[4] PTPKM. *IKPK: Indeks Kematangan dan Pematuhan Kebangsaan — PQC Migration National Compliance and Maturity Index*. Pusat Teknologi dan Pengurusan Kriptologi Malaysia, 2025.

[5] PTPKM. *PQC Audit and Compliance Guidelines Version 1.0*. Pusat Teknologi dan Pengurusan Kriptologi Malaysia, 2025.

[6] CyberSecurity Malaysia. *MySEAL: National Trusted Cryptographic Algorithm List*. https://mykripto.cybersecurity.my, 2025.

[7] CyberSecurity Malaysia. *MyCV: Malaysian Cryptography Validation*. https://mykripto.cybersecurity.my/index.php/services/mycv, 2025.

[8] CyberSecurity Malaysia. *PKTN: Produk Kriptografi Terpercaya Negara*. https://www.cybersecurity.my/portal-main/services/pktn-overview, 2025.

[9] NACSA/PTPKM. *National PQC Migration Plan 2025–2030*. Launched at ICE-PQC 2025. https://ice-pqc.upm.edu.my/, 2025.

[10] NACSA/PTPKM. *PQC Proof-of-Concept Sandbox Initiative*. https://ice-pqc.upm.edu.my/, 2025.

[11] NIST. *FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard (ML-KEM)*. National Institute of Standards and Technology, 2024.

[12] NIST. *FIPS 204: Module-Lattice-Based Digital Signature Standard (ML-DSA)*. National Institute of Standards and Technology, 2024.

[13] NIST. *FIPS 205: Stateless Hash-Based Digital Signature Standard (SLH-DSA)*. National Institute of Standards and Technology, 2024.

[14] NIST. NIST selects HQC as fifth algorithm for post-quantum encryption. National Institute of Standards and Technology. https://www.nist.gov/news-events/news/2025/03/nist-selects-hqc-post-quantum-encryption, 2025.

[15] NIST NCCoE. *NIST CSWP 48: Mappings of Migration to PQC Project Capabilities to Risk Framework Documents*. National Cybersecurity Center of Excellence, 2024.

[16] ETSI. *ETSI TR 103 619: Migration Strategies and Recommendation to Quantum-Safe Schemes*. European Telecommunications Standards Institute, 2021.

[17] ETSI. *ETSI TR 104 016: Quantum-Safe Cryptography Implementation Guidance*. European Telecommunications Standards Institute, 2024.

[18] ISO/IEC. *ISO/IEC 19790:2012: Security Requirements for Cryptographic Modules*. International Organisation for Standardisation, 2012.

[19] ISO/IEC. *ISO/IEC 24759:2017: Test Requirements for Cryptographic Modules*. International Organisation for Standardisation, 2017.

[20] NIST NCCoE. *Migration to Post-Quantum Cryptography project*. National Cybersecurity Center of Excellence. https://www.nccoe.nist.gov/crypto-agility-considerations-migrating-post-quantum-cryptographic-standards, 2024.

[21] NCSC UK. *Timelines for migration to post-quantum cryptography*. National Cyber Security Centre, United Kingdom. https://www.ncsc.gov.uk/guidance/pqc-migration-timelines, 2025.

[22] NIS Cooperation Group. *A Coordinated Implementation Roadmap for the Transition to Post-Quantum Cryptography*. European Commission. https://digital-strategy.ec.europa.eu, 2024.

[23] Canadian Centre for Cyber Security. *Roadmap for the migration to post-quantum cryptography for the Government of Canada (ITSM.40.001)*. Communications Security Establishment, 2025.

[24] NSA. *Commercial National Security Algorithm Suite 2.0 (CNSA 2.0)*. National Security Agency, United States, 2022.

[25] CISA. *Post-Quantum Cryptography Initiative*. Cybersecurity and Infrastructure Security Agency. https://www.cisa.gov/post-quantum-cryptography, 2024.

[26] Australian Signals Directorate. *Information Security Manual, December 2024 release*. Australian Cyber Security Centre. https://www.cyber.gov.au/sites/default/files/2024-12/ism.pdf, 2024.

[27] IMDA Singapore. *Quantum-safe technologies and National Quantum-Safe Network Plus (NQSN+)*. https://www.imda.gov.sg/Content-and-News/Press-Releases-and-Speeches/Press-Releases/2024, 2024.

[28] Baseri, Y., et al. Evaluation framework for quantum security risk assessment: A comprehensive strategy for quantum safe migration. *Computers & Security*, 2024.

[29] Hohm, J.S., Heinemann, A., & Wiesmaier, A. Towards a maturity model for crypto-agility assessment. *Foundations and Practice of Security*. Springer, 2021.

[30] Erol, V. Quantum Readiness in Cryptography: A Maturity-Based Framework for Post-Quantum Transition. *Preprints*, https://doi.org/10.20944/preprints, 2024.

[31] DigiCert. *DigiCert Post-Quantum Cryptography Maturity Model*. Industry White Paper, 2023.

[32] Ma, C., et al. CARAF: Crypto Agility Risk Assessment Framework. *Journal of Cybersecurity*, https://doi.org/10.1093/cybsec, 2021.

[33] DOE. *Cybersecurity Capability Maturity Model (C2M2) v2.0*. U.S. Department of Energy, 2022.

[34] NIST. *Cybersecurity Framework (CSF) 2.0*. National Institute of Standards and Technology. https://doi.org/10.6028/NIST.CSWP.29, 2024.

[35] ISO/IEC. *ISO/IEC 27001:2022: Information Security Management Systems — Requirements*. International Organisation for Standardisation, 2022.

[36] Page, M.J., et al. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*, 372, n71, https://doi.org/10.1136/bmj.n71, 2021.

[37] Kitchenham, B., & Charters, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering* (EBSE Technical Report EBSE-2007-01). Keele University, 2007.

[38] CASP. *CASP Checklist: 10 Questions to Help You Make Sense of Qualitative Research.* Critical Appraisal Skills Programme. https://casp-uk.net, 2018.

[39] Shor, P.W. Algorithms for quantum computation: discrete logarithms and factoring. *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 124–134, 1994.

[40] Mosca, M. Cybersecurity in an era with quantum computers: will we be ready? *IEEE Security & Privacy*, 16(5), 38–41. https://doi.org/10.1109/MSP.2018.3761723, 2018.

[41] Bernstein, D.J., & Lange, T. Post-quantum cryptography. *Nature*, 549(7671), 188–194. https://doi.org/10.1038/nature23461, 2017.

[42] Chen, L., et al. *Report on Post-Quantum Cryptography (NISTIR 8105)*. NIST, 2016.

[43] Kong, I., Janssen, M.F.W.H.A., & Bharosa, N. Realizing quantum-safe information sharing: Implementation and adoption challenges and policy recommendations for quantum-safe transitions. *Government Information Quarterly*, 40(4), 101884. https://doi.org/10.1016/j.giq.2023.101884, 2023.

[44] von Nethen, N., Wiesmaier, A., Alnahawi, N., & Henrich, J. PMMP — PQC Migration Management Process. In *Proceedings of the European Interdisciplinary Cybersecurity Conference (EICC 2024)*. ACM. https://doi.org/10.1145/3655693.3655719, 2024.

[45] Mills, M.O., & Ghorbani, A.A. PQC-OAV in Action: A High-Fidelity Prototype for Evidential Assurance of PQC in TLS. In *2025 International Conference on Intelligent Computing and Next Generation Networks (ICNGN)*, 1–5. https://doi.org/10.1109/ICNGN67480.2025.11413806, 2025.

[46] Chouhan, V., Aldarwbi, M., Sadeghi, S., Ghorbani, A., Chow, A., & Burko, R. Assessing the quantum readiness of cryptographic standards: Recommendations toward quantum-era compliance. *Computer Standards & Interfaces*. Available online 2025. https://www.sciencedirect.com/science/article/pii/S0920548925001436, 2025.

[47] Joseph, D., Misoczki, R., Manzano, M., Tricot, J., Dominguez Pinuaga, F., Lacombe, O., Leichenauer, S., Hidary, J., Venables, P., & Hansen, R. Transitioning organizations to post-quantum cryptography. *Nature*, 605(7909), 237–243. https://doi.org/10.1038/s41586-022-04623-2, 2022.

[48] Santa Quintero, R.A., & Meneses Velandia, S.L. Integrating Post-Quantum Cryptography in Document Lifecycle Management: Harmonizing ISO 15489-1 and NIST PQC Standards. ACM. https://doi.org/10.1145/3771678.3771685, 2025.

[49] Kundu, A., & Kompella, R. Quantum-Resistant Security: PQC Readiness and Research Challenges (Invited). In *2025 62nd ACM/IEEE Design Automation Conference (DAC)*. https://doi.org/10.1109/DAC63849.2025.11132566, 2025.

[50] Ivanov, K., Nenova, M., & Tsochev, G. Hybrid Cryptography: A Framework for Post-Quantum Agility and Resilience. In *2025 28th International Symposium on Wireless Personal Multimedia Communications (WPMC)*. https://doi.org/10.1109/WPMC67460.2025.11351237, 2025.

[51] Mamun, A.A., Abrar, A., Rahman, M., Salek, M.S., & Chowdhury, M. Enhancing Transportation Cyber-Physical Systems Security: A Shift to Post-Quantum Cryptography. *arXiv preprint* arXiv:2411.13023. https://arxiv.org/abs/2411.13023, 2024.

[52] García-Cid, M.I., Kourtis, M.-A., Domingo, D., Tcholtchev, N., Markakis, E.K., Niemiec, M., Faba, J., Ortiz, L., Martín, V., López, D., Xilouris, G., Gagliardi, M., González, J., García, M., Comande, G., & Stoianov, N. PQ-REACT: Post Quantum Cryptography Framework for Energy Aware Contexts. In *Proceedings of the 19th International Conference on Availability, Reliability and Security (ARES 2024)*. ACM. https://doi.org/10.1145/3664476.3670868, 2024.

[53] Giron, A.A. Migrating Applications to Post-Quantum Cryptography: Beyond Algorithm Replacement. In *Proceedings of the 21st International Conference on Security and Cryptography (SECRYPT 2024)*, 2024.

[54] Muzikant, P., & Willemson, J. Deploying Post-quantum Algorithms in Existing Applications and Embedded Devices. In *Ubiquitous Security*, Communications in Computer and Information Science, 147–162. Springer Nature Singapore. https://doi.org/10.1007/978-981-97-1274-8_10, 2024.

[55] Sowa, J., Hoang, B., Yeluru, A., Qie, S., Nikolich, A., Iyer, R., & Cao, P. Post-Quantum Cryptography (PQC) Network Instrument: Measuring PQC Adoption Rates and Identifying Migration Pathways. In *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 1, 1835–1846. https://doi.org/10.1109/QCE60285.2024.00213, 2024.

[56] Lu, X., Zheng, T., Ge, R., Zhang, M., Xie, Z., & Lin, Z. Exploration of Post-Quantum Cryptography Application Scenarios in 5G and Evolved Networks. In *2025 10th International Conference on Cyber Security and Information Engineering (ICCSIE)*. https://doi.org/10.1109/ICCSIE65832.2025.00076, 2025.

[57] Kern, D., Krauß, C., Lauser, T., Alnahawi, N., Wiesmaier, A., & Niederhagen, R. QuantumCharge: Post-Quantum Cryptography for Electric Vehicle Charging. In *Applied Cryptography and Network Security (ACNS 2023)*, Lecture Notes in Computer Science, 85–111. Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-33491-7_4, 2023.

[58] CycloneDX. *Cryptography Bill of Materials (CBOM) capability in CycloneDX v1.6*. OWASP Foundation. https://cyclonedx.org/capabilities/cbom/, 2024.

[59] Almutairi, M., & Sheldon, F.T. Resilience of Post-Quantum Cryptography in Lightweight IoT Protocols: A Systematic Review. *Eng*, 6(12), 346. https://doi.org/10.3390/eng6120346, 2025.

[60] Hasija, T., Ramkumar, K.R., Kaur, A., & Bali, M.S. Exploring the landscape of post quantum cryptography: a bibliometric analysis of emerging trends and research impact. *Journal of Big Data*, 12(1), 225. https://doi.org/10.1186/s40537-025-01269-5, 2025.

[61] Näther, C., Herzinger, D., Gazdag, S.-L., Steghöfer, J.-P., Daum, S., & Loebenberger, D. Migrating Software Systems Toward Post-Quantum Cryptography — A Systematic Literature Review. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3450306, 2024.

[62] Ganesh, R., Khan, B.U.I., Khan, A.R., & Kamsin, A.B. A panoramic survey of the advanced encryption standard: from architecture to security analysis, key management, real-world applications, and post-quantum challenges. *International Journal of Information Security*. https://doi.org/10.1007/s10207-025-01116-x, 2025.

[63] Huang, K., & Pan, Y. Research on the Commercial Cryptography Protection Framework for the Power Sector: A Roadmap from Legacy Cryptography to Post-Quantum Migration. In *2025 5th International Conference on Computer Science, Electronic Information Engineering and Intelligent Control Technology (CEI)*. https://doi.org/10.1109/CEI66465.2025.11398533, 2025.

[64] Gagliardi, M., & D'Elia, C. A Proportional Approach to Cybersecurity Challenges in the Financial Sector: Ideas from Post-Quantum Cryptography Legal Analysis. In *Proceedings of ITASEC & SERICS 2025*, CEUR Workshop Proceedings, 2025.

A Catalogue of Primary Studies (S01–S18)

Table 6 presents all primary studies included in the review, detailing the study identifier, source reference, framework type (I–III as defined in Section 4.2), addressed audit dimensions (D1–D10 as described in Section 4.3), validation approach, and quality appraisal score out of 8 with the corresponding High/Medium/Low classification. A tick indicates explicit coverage, while a tilde indicates partial coverage. Studies S14–S18 represent the borderline inclusions discussed in Section 3.5.

Table 6: Complete catalogue of the 18 primary studies included in the review.

ID	Ref.	Year	Type	Dimensions covered	Validation	QA (band)
S01	[43]	2023	II	D3, D4, D6, D7, D9; D1∼, D5∼	Empirical (interview)	7.5 (H)
S02	[44]	2024	I	D1, D2, D5, D6, D7, D9; D3∼, D4∼	Expert review	7.0 (H)
S03	[45]	2025	II	D8; D1∼, D3∼, D5∼, D7∼	Empirical (prototype)	7.0 (H)
S04	[46]	2025	II	D3, D4, D7; D1∼, D5∼, D8∼	Empirical (documentary)	6.5 (H)
S05	[47]	2022	I	D1, D7; D3∼, D4∼, D5∼, D6∼	Conceptual	4.5 (M)
S06	[48]	2025	I	D1, D3, D7; D4∼, D6∼, D8∼	Case study	6.0 (M)
S07	[49]	2025	II	D5; D1∼, D2∼, D3∼, D7∼, D8∼	Empirical (tool)	4.5 (M)
S08	[52]	2024	III	D1, D5; D3∼, D4∼, D6∼, D7∼, D10∼	Case study (pilots)	5.5 (M)
S09	[53]	2024	I	D6, D7; D1∼, D3∼, D4∼	Conceptual	3.0 (L)
S10	[55]	2024	II	D7; D1∼, D2∼, D4∼, D8∼	Empirical (measurement)	6.5 (H)
S11	[62]	2025	III	D1, D3, D4, D7; D5∼, D6∼	Empirical (benchmark)	7.0 (H)
S12	[63]	2025	III	D1, D3, D4, D6, D9, D10; D7∼	Institutional	5.0 (M)
S13	[64]	2025	II	D3, D4; D5∼, D7∼, D8∼	Legal-doctrinal	5.5 (M)
S14	[50]	2025	I	D5, D7; D1∼, D3∼, D4∼, D8∼	Conceptual	2.0 (L)
S15	[51]	2024	I	D4, D7; D1∼, D3∼, D8∼	Conceptual	4.0 (M)
S16	[54]	2024	I	D7; D1∼	Case study	4.0 (M)
S17	[56]	2025	III	D1, D3, D4, D5, D7, D10; D6∼, D9∼	Institutional	2.5 (L)
S18	[57]	2023	I	D7; D1∼, D4∼	Case study (prototype)	6.5 (H)

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 480–494.

Implementation of Dynamic Double-Lock Bézier-Menezes-Vanstone Elliptic Curve Cryptography

Amadeus Lim Min Chern[1], Md Yushalify Misro[1,2] and Hailiza Kamarulhaili[1,2]

[1] School of Mathematical Sciences, Universiti Sains Malaysia,
11800 Gelugor, Pulau Pinang, Malaysia
minchern1274@gmail.com,{yushalify,hailiza}@usm.my
[2] Malaysia Cryptology Technology and Management Centre (PTPKM), c/o Universiti Putra Malaysia, UPM
43400 Serdang, Selangor, Malaysia {yushalify,hailiza}@usm.my

Abstract. This paper presents the implementation and preliminary evaluation of a Double-Lock Bézier-Menezes-Vanstone Elliptic Curve Cryptography scheme, abbreviated as DL-BMVECC, for text encryption. The proposed method integrates Menezes-Vanstone Elliptic Curve Cryptography, a dynamic Bézier coefficient matrix, and a transaction-specific double-lock mechanism. The dynamic Bézier coefficient matrix introduces a one-parameter shape parameter to vary the matrix transformation across transactions, while the double-lock layer adds independent sender and receiver lock values. The scheme also applies one-time transaction parameters, where private keys, public keys, Bézier shape parameters, and double-lock components are generated uniquely for each transaction. Key-space and key-sensitivity analyses show that the tested configuration provides a large search space and strong sensitivity to small key changes. However, NIST statistical testing shows mixed randomness behaviour, with several baseline tests returning non-random results. Therefore, DL-BMVECC is presented as a layered implementation that requires additional diffusion and formal security analysis before stronger cryptographic indistinguishability or post-quantum claims can be made.

Keywords: No keywords given.

Keywords: MVECC, Bézier Coefficient Matrix, Double-Lock Encryption, Text Encryption

1 Introduction

Secure communication is essential in today's digital landscape to protect sensitive data from cyber threats. While traditional encryption methods are effective, they often require significant computing power, which can slow down high-speed networks [4]. To address this, Neal Koblitz and Victor Miller introduced Elliptic Curve Cryptography (ECC) in 1985 as a more efficient alternative to older systems like RSA [1, 2]. ECC secures data using the algebraic structure of elliptic curves. Its security relies on the Elliptic Curve Discrete Logarithm Problem (ECDLP), a mathematical problem in which scalar multiplication of points on a curve is efficient, but recovering the scalar from the resulting point is computationally difficult without the private key [3].

The primary advantage of ECC is that it provides strong security using much smaller encryption keys [8, 9]. For instance, a 256-bit ECC key offers the same level of protection as a much larger 3072-bit RSA key [8]. Because the keys are smaller, the system requires less processing power, memory, and bandwidth. This makes ECC faster and more energy-efficient, which is especially important for devices with limited battery and processing capabilities, such as smartphones, wireless networks, and Internet of Things (IoT) devices [8, 9].

Within the ECC family, the Menezes-Vanstone Elliptic Curve Cryptosystem (MVECC) was introduced in 1993 to streamline the encryption process [5]. Unlike standard ECC, MVECC does not require the complex step of converting the plaintext message into mathematical points on the curve. Instead, the sender uses a randomly generated shared secret key to encrypt the message, and the receiver uses their private key to recreate that secret and decrypt the data. However, MVECC has its drawbacks: the mathematical operations required for decryption can be computing-intensive, and the system can be vulnerable to attacks if its security parameters are not carefully chosen. To resolve these issues, researchers are actively developing new modifications to make MVECC faster, more robust, and more secure [5].

Bézier curves were originally created by Pierre Bézier and Paul de Casteljau to assist in computer-aided geometric design [6, 7]. Mathematically, these curves rely on a set of control points to smoothly guide their shape and trajectory. They possess several unique geometric properties such that the curve will start and end exactly at their first and last control points [7]. Recently, these geometric properties have been recognized as a novel way to improve digital security. Researchers have started blending Bézier equations into existing encryption frameworks, such as the Menezes-Vanstone Elliptic Curve Cryptosystem (MV-ECC) [5]. By using the curve's control points and shape parameters as dynamic secret keys, the encryption equations become much more complex and unpredictable. This integration allows for adaptive security models that use flexible geometric variables to improve how secret keys are generated and distributed [5, 7].

Despite these promising innovations, standard Elliptic Curve Cryptography (ECC) and MV-ECC still face significant operational hurdles [12]. For example, standard MV-ECC requires heavy computational steps during the decryption process and causes the encrypted data to become twice the size of the original message [10, 12]. These inefficiencies can severely slow down performance and increase latency, especially for devices with limited computing power or bandwidth, such as mobile networks and Internet of Things (IoT) gadgets [12]. Compounding these inefficiencies is the fact that most current encryption systems are entirely static. Once the security parameters are set at the beginning of a session, they do not change. This static setup gives attackers unlimited time to study the system, analyze data traffic patterns, and slowly break down the defenses, as the system has no way to adapt to incoming threats [11]. In today's rapidly evolving cyber threat landscape, this inability to adapt is a critical vulnerability.

While Bézier curves offer excellent mathematical tools to create dynamic, ever-changing encryption parameters, they remain largely underutilized in practical cybersecurity [11]. So far, there have only been early attempts to incorporate these geometric features into ElGamal and MVECC systems [5, 13]. The field still lacks a comprehensive, theoretical framework that fully leverages adjustable geometric shapes to continuously scramble data and act as a built-in defense mechanism [11, 13].

To address the limitations of existing encryption models, this study proposes a layered cryptographic framework known as Double-Lock Bézier-Menezes-Vanstone Elliptic Curve Cryptography (DL-BMVECC). In the previous study [6, 13], the encryption scheme used the standard cubic Bézier coefficient matrix, whose coefficients are fixed once the degree is chosen. In this work, we introduce a cubic Bézier-like coefficient matrix with an adjustable shape parameter, allowing the matrix to vary and providing greater flexibility for encryption. This also increases the key space compared with using a fixed coefficient matrix. In addition, the previous study employed only a single encryption layer based on the Bézier coefficient matrix. In contrast, the proposed method introduces a double lock security framework, which provides an additional layer of protection of the encryption scheme.

The remainder of this paper is structured as follows. Section 2 provides the implementation of the proposed Dynamic Double-Lock Bézier-Menezes-Vanstone Elliptic Curve Cryptography (DL-BMVECC) model. Section 3 presents key-space analysis, key-sensitivity analysis, information entropy, NIST statistical testing, and the current security limitations of the implementation. Finally, Section 4 summarizes the study's key contributions and outlines potential directions for future research in adaptive cryptographic

systems.

2 Implementation

The proposed DL-BMVECC scheme consists of three main encryption layers. The first layer uses ECC-based key generation and Menezes-Vanstone encryption. The second layer applies a dynamic Bézier coefficient matrix controlled by a shape parameter. The third layer applies a double-lock encryption mechanism. The scheme is applied to text encryption. Figure 1 demonstrates the flow of the proposed scheme.

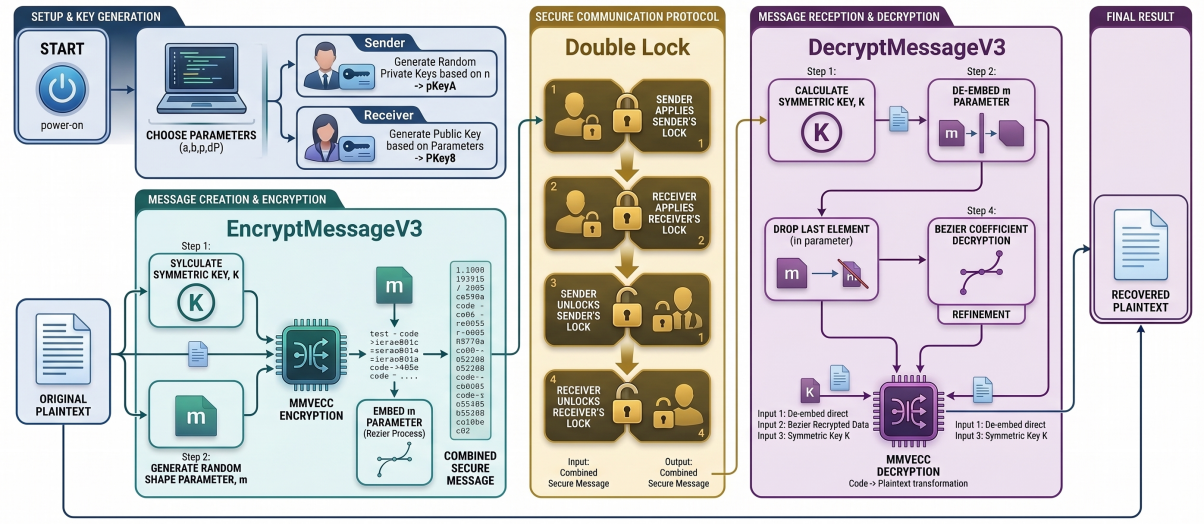


Figure 1: Flowchart of the proposed scheme

2.1 Version Selection

Several versions of the cryptographic framework were considered before selecting DL-BMVECC as the final proposed scheme as tabulated in Table 1. The earlier versions used ECC with ElGamal key exchange and either static or dynamic Bézier coefficient matrices. Later versions introduced the double-lock mechanism and replaced ElGamal with Menezes-Vanstone ECC. This system utilizes a layered approach to encryption, offering different options for each layer to enhance security.

Version 1: Base Model with Elliptic Curve Cryptography, ElGamal Public Key Exchange, and Static Bézier Coefficient Matrix

Version 2: Dynamic Bézier Coefficient Matrix with Shape Parameter

Version 3: Menezes-Vanstone ECC with Dynamic Bézier Coefficient Matrix

Version 4: Double-Lock Encryption with ElGamal and Dynamic Bézier Coefficient Matrix

Version 5: Double-Lock Menezes-Vanstone with Dynamic Bézier ECC (DL-BMVECC)

Table 1: Summary of Cryptographic Versions

Version	Key Exchange	Bézier Matrix	Double-Lock
Version 1	ElGamal	Static	No
Version 2	ElGamal	Dynamic	No
Version 3	MVECC	Dynamic	No
Version 4	ElGamal	Dynamic	Yes
Version 5	MVECC	Dynamic	Yes

Version 5 was selected because it combines Menezes-Vanstone ECC, dynamic Bézier parameters, and the additional double-lock layer. It also uses one-time transaction keys, meaning that the private keys, public keys, shape parameter, and double-lock components are not reused across transactions. A comparison between the proposed method and the previous work is presented in Table 2.

Table 2: Comparison with Related Cryptographic Constructions

Method	Core Construction	Double-Lock	Main Distinction
Standard MVECC	Menezes-Vanstone ECC	No	Uses an ECC shared secret for encryption, without a Bézier matrix or lock layer.
Bézier-MVECC [5]	Menezes-Vanstone ECC with a Bézier-based component	No	Adds a Bézier component, but does not include the proposed double-lock mechanism.
Bézier-ElGamal ECC [13]	ElGamal ECC with a Bézier coefficient matrix	No	Uses an ElGamal-based ECC construction rather than Menezes-Vanstone ECC.
Dynamic Bézier MVECC	Menezes-Vanstone ECC with a dynamic one-parameter Bézier matrix	No	Uses dynamic Bézier parameters, but does not apply sender and receiver lock values.
Proposed DL-BMVECC	Menezes-Vanstone ECC with a dynamic one-parameter Bézier matrix	Yes	Combines ECC keys, the Bézier shape parameter, and transaction-specific lock values.

2.2 Initializing Elliptic Curve Parameters

Suppose User A (Sender) wants to send an encrypted media to User B (Receiver) using the proposed method above, DL-BMVECC. They first agreed on the parameters of elliptic curve through unsecured public network as in Equation (1)

$$E : y^2 = x^3 + x + 2547 \pmod{3023} \tag{1}$$

where $a = 1$, $b = 2547$, $p = 3023$ satisfying the condition in Equation (2) below

$$4a^3 + 27b^2 \pmod{p} = 2027 \neq 0. \tag{2}$$

Since the order of the elliptic curve group is $\#E = 3083$, which is prime, then any point on E can be chosen as the base point. In this context, we choose $DP = \{2237, 2480\}$ such that

$$2480^2 \pmod{3023} = 1618 = 2237^3 + 2237 + 2547 \pmod{3023}$$

implying that the point DP lies on the elliptic curve, E.

2.3 Key Generation

The private key generation should be in the range of $pKey \in [1, p-1]$. We use the notation of $pKey$ and $PKey$ for private and public key respectively as demonstrated in Table 3. The private keys will be kept securely by the respective own user whereas the public keys are shared publicly over the unsecured network to the opposing users. At this point, both users have established a shared key known only to themselves.

Table 3: Key Generation Simulation Table for User A and B

	Private Key	**Public Key**	**Symmetric Key**
User A (Sender)	$pKeyA = 1377$	$PKeyA = pKeyA \cdot DP$ $= \{433, 1974\}$	$K = pKeyA \cdot PKeyB$ $= \{2470, 1295\}$
User B (Receiver)	$pKeyB = 2680$	$PKeyB = pKeyB \cdot DP$ $= \{1108, 2311\}$	$K = pKeyB \cdot PKeyA$ $= \{2470, 1295\}$

2.4 Encryption Method

Suppose User A wants to encrypt the message below to User B.

> "Target Alpha located at coordinates 45.6789° N, 123.4567° W. ETA for extraction: 2300 hours."

We will first perform Modified Menezes-Vanstone Elliptic Curve Cryptography (MMVECC). Converting each character in the string to ASCII decimal format gives $\{d_1, d_2, d_3, \dots\}$ as shown in Table 4.

Table 4: Plaintext to Decimal ASCII Simulation Conversion Table

Plaintext	T	a	r	g	e	...
Decimal ASCII	84	97	114	103	101	...
	d_1	d_2	d_3	d_4	d_5	...

We denote the symmetric key $K = \{x, y\} = \{2470, 1295\}$. Then for the first ciphertext, $C_1 = \{c_1, c_2, c_3, \dots\}$ where $c_1 = (d_1 + x + y) \pmod p$ and $c_2 = (d_2 + c_1) \pmod p$. The first ciphertext encryption is shown below in Table 5.

Table 5: MMVECC Decimal ASCII Simulation Encryption

Decimal ASCII	84	97	114	103	101	...
	d_1	d_2	d_3	d_4	d_5	...
$\boldsymbol{C_1}$	826	923	856	959	843	...
	c_1	c_2	c_3	c_4	c_5	...

The second subsequent encryption will be performed using Bézier Coefficient Matrix, CB with shape parameter, $m \in (0, 3]$. The Bézier-like cubic basis functions are as such, $B_i(u), i = 0, 1, 2, 3$.

The shape parameter m is the free parameter that controls the weights of the Bézier-like cubic basis functions and, consequently, the entries of the Bézier coefficient matrix. Changing m changes the matrix transformation applied to each ciphertext block. In the proposed scheme, m is generated for each transaction so that the same plaintext block is not always transformed using the same coefficient matrix. The restriction $m \in (0, 3]$ is used to preserve the intended one-parameter Bézier-like basis behaviour,

while $m \neq 0$ is required because the inverse coefficient matrix contains terms of the form $1/m$. Therefore, a zero value of m would make decryption impossible.

In this work, we intentionally adopt a one-parameter specialization as a design choice for the proposed cryptographic scheme. The use of a single parameter enables the construction of a direct coefficient matrix with a simple inverse, thereby reducing computational complexity and minimizing the number of parameters that need to be transmitted or shared during the encryption and decryption processes. This simpler parameterization also facilitates efficient key generation while preserving the mathematical properties required for the proposed encryption framework. Therefore, the use of one parameter is intentional and motivated by implementation efficiency rather than a restriction of the Bézier-like formulation.

$$B_0(u) = (1-u)^2(1+(2-m)u), \tag{3}$$
$$B_1(u) = m(1-u)^2u, \tag{4}$$
$$B_2(u) = m(1-u)u^2, \tag{5}$$
$$B_3(u) = u^2(1+(2-m)(1-u)). \tag{6}$$

The resulting polynomial is

$$C(t) = \sum_{i=0}^{3} P_i B_i(t). \tag{7}$$

with $u \in [0,1]$, which can be simplified to the following form

$$C(t) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ -m & m & 0 & 0 \\ 2m-3 & -2m & m & 3-m \\ 2-m & m & -m & m-2 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} = u \cdot CB \cdot P.$$

The CB is known as the Bézier-like cubic basis coefficient matrix. The inverse of CB has the following form and exists for $m \neq 0$,

$$CB^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & \frac{1}{m} & 0 & 0 \\ 1 & \frac{m-1}{m} & \frac{m-2}{m} & \frac{m-3}{m} \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

If the final block has fewer than four elements, the corresponding lower-dimensional Bézier coefficient matrix such as quadratic Bézier, QB, and linear Bézier, LB, is applied based on the remaining block size.

$$QB = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix}, \quad LB = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$$

We partition the first ciphertext into partitions of four as tabulate in Table 6. The last block length only has four possibilities: 4, 3, 2 or 1. Nevertheless, we will perform the matrix multiplication with the corresponding coefficient matrix as far as the dimension is concerned.

Table 6: Bézier Coefficient Matrix Encryption Simulation with Shape Parameter m

C_1	826	923	856	959	843	...	...
	c_1	c_2	c_3	c_4	c_5	...	...
Partition	Block 1				Block 2		
	$CB \cdot \{c_1, c_2, c_3, c_4\}$ $= \{826, \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$ $= \{c_1', c_2', c_3', c_4'\}$				...	...	

We will now securely embed the shape parameter, m, to the ciphertext. This can be done by using the symmetric key of the secured channel. In this example, we use

$$m = \frac{750009}{500000}$$

Therefore,

$$\text{Encrypted } m = m' = m \cdot \frac{x}{y} = \frac{185252223}{64750000}$$

The result will be the second ciphertext, $C_2 = \{c_1', c_2', c_3', c_4', \ldots, m'\}$.

Finally, the third encryption is performed without any key exchange across any channel. User A and User B will both decide on their own respective lock list with the condition below: $L = \{l_1, l_2, l_3, \ldots\}$ such that $|L| = \lceil \frac{|C_2|}{4} \rceil$. In this context, we use the following:

$$\begin{aligned} L_A = \{ &-9330860753, 6324793423, 938297001, -762216994, -6392925642, \\ &-8168600495, 3933117563, -3065494685, -2239012305, -5818016097, \\ &-1861601546, -1359107190, -6528677145, 5365188670, -5400430920, \\ &-226370303, -9050628932, -802006521, 8531425370, 5154605360, \\ &-7245412736, 9338608994, -6632663059, 925798801\} \end{aligned}$$

$$\begin{aligned} L_B = \{ &-2808974576, -137616513, -2401273125, -8761714860, 8882263079, \\ &8302888096, 1838549596, 8814470296, 7427201951, 2376695553, \\ &-167063560, -6609009886, -5296838654, -8240971550, 2490464969, \\ &-2076127474, 1649987499, 6811531458, -9210938282, 9676691483, \\ &-2525792149, 8410817196, -119990116, 6757761239\} \end{aligned}$$

Partition C_2 into blocks of four and User A will "lock" the message by adding each element in L_A to the first element in each block of the partition. The intermediate encrypted ciphertext will be sent to User B and the same operation will be performed using L_B resulting in the final ciphertext, C_F and sent back to User A as shown in Table 7. This double action of adding lock by both users are known as the "Double-Lock" Encryption.

Table 7: Double-Lock on Text Encryption Simulation

C_2	Block 1	Block 2	...
Partition	$\{826, \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	...	...
Add L_A	$\{826 + L_A[[1]], \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	...	...
Add L_B	$\{826 + L_A[[1]] + L_B[[1]], \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	...	...
C_F	$c_1'' = \{-12139834503, \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	c_2''	...

This is the final complete ciphered text formed by the given scheme: $C_F = \{c_1'', c_2'', c_3'', \dots\}$.

2.5 Decryption Method

Suppose User B wants to decrypt the final ciphered text, C_F. The ciphered text needs to be "unlocked" first by User A followed by User B using their respective lock defined earlier. This is made possible due to the associative laws of addition. We will first partition C_F into blocks of four as in Table 8.

Table 8: Double-Lock on Text Decryption Simulation

C_F	Block 1	Block 2	...
Partition	$\{-12139834503, \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	...	...
Remove L_A	$\{-12139834503 - L_A[[1]], \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	...	...
Remove L_B	$\{-12139834503 - L_A[[1]] - L_B[[1]], \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	...	...
D_1	$d_1 = \{826, \frac{72750873}{500000}, -\frac{23252673}{500000}, \frac{85009}{2500}\}$	d_2	...

This will result in our first decrypted ciphertext, D_1. The removal of both locks will reveal the encrypted m appended at the last element of the ciphered text. User B will need to retrieve the original

m using the symmetric key as such:

$$m' = m = m' \cdot \frac{y}{x} = \frac{750009}{500000}$$

The shape parameter, m, is needed to continue the decryption process of the Bézier Coefficient Matrix. We define the inverse coefficient matrix as follows.

$$CB^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\frac{750009}{500000} & \frac{750009}{500000} & 0 & 0 \\ \frac{9}{250000} & -\frac{750009}{250000} & \frac{750009}{500000} & \frac{749991}{500000} \\ \frac{249991}{500000} & \frac{750009}{500000} & -\frac{750009}{500000} & -\frac{249991}{500000} \end{bmatrix}^{-1}$$

$$QB^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix}^{-1}, \quad LB^{-1} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}^{-1}$$

Similar to encryption, we will partition into groups of four and perform the matrix multiplication of the respective inverse Bézier Coefficient Matrix with each partition of the first decrypted ciphertext, D_1 as tabulate in Table 9.

Table 9: Bézier Coefficient Matrix Decryption Simulation with Shape Parameter m

$\boldsymbol{D_1}$	d_1				d_2	...
	826	$\frac{72750873}{500000}$	$-\frac{23252673}{500000}$	$\frac{85009}{2500}$	...	...
Partition	Block 1				Block 2	...
	$ICB \cdot d_1$ $= \{826, 923, 856, 959\}$ $= d'_1$				d'_2	...

This will result in the second decrypted ciphertext, $D_2 = \{d'_1, d'_2, d'_3, \dots\}$. Table 10 shown the last decryption process is to perform MMVECC decryption which can be done as such: $D_3 = \{d''_1, d''_2, d''_3, \dots\}$ where $d''_1 = (d'_1 - x - y) \pmod p$ and $d''_2 = (d'_2 - d'_1) \pmod p$.

Table 10: MMVECC Decimal ASCII Simulation Decryption

$\boldsymbol{D_2}$	826	923	856	959	...
	d'_1	d'_2	d'_3	d'_4	...
$\boldsymbol{D_3}$ **Decimal ASCII**	84	97	114	103	...
	d''_1	d''_2	d''_3	d''_4	...

Finally, to obtain the decrypted plaintext, we convert the ASCII decimal format to characters as in Table 11.

Table 11: Decimal ASCII to Plaintext Simulation Conversion Table

D_3 Decimal ASCII	84	97	114	103	...
	d''_1	d''_2	d''_3	d''_4	...
Plaintext	T	a	r	g	...

3 Result and Discussion

The proposed DL-BMVECC scheme was evaluated using key space analysis, key sensitivity analysis, information entropy, and NIST statistical testing. The purpose of the evaluation is to determine whether the proposed layered structure improves unpredictability, resistance to key modification, and statistical randomness.

3.1 Key Space Analysis for Text Encryption

A large key space is important for resisting brute-force attacks. In the proposed text encryption scheme, the key space depends on the elliptic curve prime number p, the Bézier shape parameter m, and the double-lock values.

Let t be the bit length required to represent p, where

$$t = \lceil \log_2(p) \rceil. \tag{8}$$

Let n be the length of the message. The minimal key space search for DL-BMVECC is represented as

$$2^{17n+t+35}. \tag{9}$$

For example, when $p = 3023$, $n = 6$, and $t = 12$, the key space becomes

$$2^{17(6)+12+35} = 2^{149}. \tag{10}$$

Since

$$2^{149} > 2^{100}, \tag{11}$$

the proposed scheme provides a sufficiently large key space for resisting exhaustive key search in the simulated setting.

3.2 Key Sensitivity Analysis for Text Encryption

Key sensitivity analysis was performed by attempting to decrypt ciphertext using slightly modified keys. The results show that even a small change in the private key or double-lock key produces significantly different decrypted output. Suppose unauthorized third parties managed to guess all parameters except the private key. Considering the worst-case scenario, in which they are of 1 integer difference from the true private key, the resulting decrypted text is illustrated in Figure 2

Łǻ○ōčŕ̥ōŠŮ□ǬčȀã□šǻÕ□ŜÕ̊ȿHȚŞ□Ôčǣ|8ŕǬ□Ǭő□ҧǬťt

Figure 2: Decrypted text produced using a modified private key

and the corresponding distance analysis is tabulated in Table 12 as follows.

Table 12: Private Key Sensitivity for Text Encryption

Metric	Value
Hamming Distance	41
Levenshtein Distance	41
Damerau-Levenshtein Distance	41
Smith-Waterman Similarity	1

Suppose unauthorized third parties managed to guess all parameters except the double-lock key. Considering the case where all the entries are of 1 integer difference from the true lock matrix, the resulting decrypted text is provided as in Figure 3

ŁōčŕōŠŮčãšÕŜÕỌṢÔčæ|8ŕött

Figure 3: Decrypted text produced using modified double-lock values

Table 13: Double-Lock Key Sensitivity for Text Encryption

Metric	Value
Hamming Distance	21
Levenshtein Distance	21
Damerau-Levenshtein Distance	21
Smith-Waterman Similarity	2

The sensitivity to key changes is pronounced, with all distance metrics in Table 13showing high value suggesting that any alteration in the private key leads to a substantially different ciphertext. The low Smith-Waterman similarity score further confirms that the ciphertexts are highly dissimilar.

Both analyses demonstrate that the encryption algorithms are highly sensitive to key changes. This sensitivity is crucial for ensuring the security of encrypted data, as even minor alterations in the key produce significantly different ciphertexts, thereby enhancing the encryption strength and reducing the likelihood of successful attacks.

3.3 Information Entropy for Text Encryption

Information entropy was used to evaluate the distribution of symbols in the encrypted text. In the text encryption simulation, the ciphertext contained 37 distinct characters. Therefore, the theoretical maximum entropy is

$$H_{\max} = \log_2(37) = 5.2095. \tag{12}$$

The calculated entropy was

$$H = 4.4077. \tag{13}$$

Table 14: Information Entropy for Text Encryption

Measure	Value
Maximum Entropy	5.2095
Calculated Entropy	4.4077
Absolute Difference	0.8018
Percentage Difference	15.39%

The calculated entropy in Table 14 is below the theoretical maximum. This indicates that the ciphertext has some random characteristics, but the current text encryption output does not reach the ideal uniform distribution. Therefore, additional diffusion should be considered to improve the statistical distribution of the final ciphertext.

3.4 NIST Statistical Test for Text Encryption

The NIST statistical test suite was used to evaluate randomness in the encrypted text output as reported in Table 15 and 16. The results show mixed behaviour. Some tests, including the Discrete Fourier Transform test, Linear Complexity test, Random Excursions test, and Random Excursions Variant test, indicated random behaviour. However, other tests, including the Frequency test, Runs test, Binary Matrix Rank test, and several template matching tests, indicated non-random behaviour.

Table 15: NIST Test Suite for Text Encryption

	Type of Test		P-Value	Conclusion
1	Frequency (Monobit) Test		0.0000	Non-Random
2	Frequency Test within a Block		9.1835e-128	Non-Random
3	Runs Test		0.0000	Non-Random
4	Test for the Longest Run of Ones in a Block		4.4015e-56	Non-Random
5	Binary Matrix Rank Test		0.0000	Non-Random
6	Discrete Fourier Transform (Spectral) Test		2.5498e-02	Random
7	Non-overlapping Template Matching Test		4.8347e-16	Non-Random
8	Overlapping Template Matching Test		1.8704e-07	Non-Random
9	Maurer's "Universal Statistical" Test		0.0000	Non-Random
10	Linear Complexity Test		5.2375e-01	Random
11	Serial Test		0.0000	Non-Random
12	Approximate Entropy Test		0.0000	Non-Random
13	Cumulative Sums Test (Forward)		0.0000	Non-Random
14	Cumulative Sums Test (Backward)		0.0000	Non-Random
15	Random Excursions Test			
	State	**Chi Squared**	**P-Value**	**Conclusion**
	-4	0.6301	9.8657e-01	Random
	-3	2.2401	8.1502e-01	Random
	-2	1.9472	8.5639e-01	Random
	-1	6.2727	2.8058e-01	Random
	+1	1.3636	3.7313e-01	Random
	+2	1.9068	8.6188e-01	Random
	+3	1.0000	9.5926e-01	Random
	+4	8.0745	1.5218e-01	Random

These results suggest that the proposed text encryption scheme has promising random characteristics, but it does not pass all statistical tests. Therefore, further improvement is needed to increase randomness across all NIST test categories. This is an important limitation of the current implementation and should be addressed in future work. The summary analysis of the proposed method is demonstrated in Table 17.

Table 16: Continued

16	Random Excursions Variant Test			
	State	**Counts**	**P-Value**	**Conclusion**
	-9.0	1	6.0599e-01	Random
	-8.0	1	5.8190e-01	Random
	-7.0	2	5.9466e-01	Random
	-6.0	5	6.9972e-01	Random
	-5.0	6	6.2284e-01	Random
	-4.0	10	6.2875e-01	Random
	-3.0	13	8.5036e-01	Random
	-2.0	21	9.0203e-01	Random
	-1.0	11	7.1496e-01	Random
	+1.0	3	6.6914e-01	Random
	+2.0	11	7.0292e-01	Random
	+3.0	4	5.7153e-01	Random
	+4.0	2	5.6968e-01	Random
	+5.0	5	8.5211e-01	Random
	+6.0	3	7.7322e-01	Random
	+7.0	2	6.7046e-01	Random
	+8.0	1	8.7672e-01	Random
	+9.0	8	8.7672e-01	Random

Table 17: Summary of Text Encryption Evaluation

Evaluation Aspect	**Observed Result**	**Interpretation**
Key space	2^{149} for $p = 3023$, $n = 6$	Exceeds the 2^{100} benchmark used in the simulation setting.
Private-key sensitivity	Distance values of 41	A one-integer private-key change produces a substantially different decrypted output.
Double-lock sensitivity	Distance values of 21	A one-integer lock-value change disrupts plaintext recovery.
Information entropy	4.4077 out of 5.2095	Randomness improves compared with plaintext structure, but remains below the ideal distribution.
NIST statistical tests	Mixed pass/fail behaviour	Several foundational randomness tests fail; the current text implementation requires stronger diffusion.

3.5 Security Discussion and Current Limitations

The proposed implementation combines ECC-based key generation, a dynamic Bézier matrix transformation, and transaction-specific double-lock values. These layers increase the number of transaction-

dependent parameters and produce strong key-sensitivity behaviour in the tested setting. However, the current text-encryption construction still contains algebraic and additive components, including the recursive MVECC step, matrix multiplication by the Bézier coefficient matrix, and additive double-lock operation.

These approach may preserve statistical structure if no additional diffusion layer (double-lock) is applied to the final ciphertext. The NIST results confirm this limitation. Although several tests indicate random behaviour, the failure of baseline tests such as Frequency, Runs, Block Frequency, Serial, and Approximate Entropy shows that the current ciphertext should not yet be claimed to be statistically indistinguishable from random data. Instead of using a double lock framework, integrating a robust pseudorandom number generator (PRNG), a stronger key derivation function (KDF), or an additional diffusion layer, such as a substitution permutation network or a chaotic map, could be a more effective alternative for future work. Consequently, the present work does not claim formal resistance against chosen plaintext attacks (CPA) or known plaintext attacks (KPA), particularly since the encryption output shows limitations in some baseline NIST statistical tests.

The results show that DL-BMVECC provides several advantages in the tested setting. First, the key-space analysis suggests resistance against exhaustive key search for the selected parameters. Second, the key-sensitivity analysis shows that slight changes in the private key or double-lock key produce significantly different outputs.

However, the text encryption component still requires improvement. Although the entropy and some NIST tests suggest partial random characteristics, several NIST tests produced non-random results. Therefore, the text encryption process should be further improve by adding or redesigning the diffusion layer.

4 Conclusion

This paper introduces DL-BMVECC, a multi-layered encryption framework that combines Menezes-Vanstone ECC, dynamic Bézier matrices, and a "double-lock" mechanism. By using unique, one-time parameters for every transaction, the system reduces repeated-parameter exposure and increases key sensitivity in the tested setting. The key-space and key-sensitivity analyses show promising behaviour for the selected simulation parameters. However, the mixed NIST test results indicate that the randomness of the current text encryption output still requires improvement. The double-lock feature adds a transaction-specific layer beyond standard ECC, but the system still relies on ECC components. Future work will focus on adding or redesigning a keyed diffusion layer, rerunning the NIST and entropy analyses, testing larger datasets, and analyzing computational efficiency.

References

[1] Jacobson Jr, M., Menezes, A. & Stein, A. (2004) Hyperelliptic curves and cryptography. *High Primes And Misdemeanours: Lectures In Honour Of The 60th Birthday Of Hugh Cowie Williams*. **41** pp. 255-282

[2] Koblitz, N. (1987) Elliptic curve cryptosystems. *Mathematics Of Computation*. **48**, 203-209

[3] Li, K., Zou, S. & Xv, J. (2008) Fast Parallel Molecular Algorithms for DNA-Based Computation: Solving the Elliptic Curve Discrete Logarithm Problem over GF (2n). *BioMed Research International*. **2008**, 518093

[4] Malik, M. (2010) Efficient implementation of elliptic curve cryptography using low-power digital signal processor. *2010 The 12th International Conference On Advanced Communication Technology (ICACT).* **2** pp. 1464-1468

[5] Ghadi, D. M., & AL-Rammahi, A. (2020). Improvement of Menezes-Vanstone Elliptic Curve Cryptosystem Based on Quadratic Bézier Curve Technique. *Journal of Computer Science*, **16**, 715–722. https://doi.org/10.3844/jcssp.2020.715.722

[6] Ismail, N. & Misro, M. (2022) An improved image encryption algorithm based on Bézier coefficients matrix. *Journal Of King Saud University-Computer And Information Sciences.* **34**, 10056-10067

[7] Ghadi, D. M. (2022). Based on IoT and PGP: A new secure framework using cubic bézier technique and El Gamal elliptic curve cryptosystem for information exchange. *AIP Conference Proceedings*, **2394**, 070008. https://doi.org/10.1063/5.0121088

[8] Gupta, V., Gupta, S., Chang, S., & Stebila, D. (2002). Performance analysis of elliptic curve cryptography for SSL. *Proceedings of the 1st ACM workshop on Wireless security*, 87–94. https://doi.org/10.1145/570681.570691

[9] Khan, M. R., Upreti, K., Alam, M. I., Khan, H., Siddiqui, S. T., Haque, M., & Parashar, J. (2023). Analysis of Elliptic Curve Cryptography & RSA. *Journal of ICT Standardization.* https://doi.org/10.13052/jicts2245-800x.1142

[10] Bhatnagar, N. (2018). *Mathematical Principles of the Internet: Engineering Fundamentals.* CRC Press.

[11] Das, S. K., & Dhara, B. C. (2023). A non-chaotic image encryption method using Bezier curves. *Multimedia Tools and Applications*, **82**, 23595–23620. https://doi.org/10.1007/s11042-022-14068-1

[12] Dawahdeh, Z., Yaakob, S., & Sagheer, A. (2016). Modified ElGamal Elliptic Curve Cryptosystem using Hexadecimal Representation. *International Journal of Computer Applications*, **138**(14), 1–7.

[13] Ismail, N. & Misro, M. (2022). Bézier Coefficients Matrix for ElGamal Elliptic Curve Cryptosystem. *Malaysian Journal of Mathematical Sciences*, **16**(3), 483–499.

Proceedings of the 10th International Cryptology and Information Security Conference 2026
CRYPTOLOGY2026, pp. 495–515.

Torus: a novel block cipher via power mappings over pseudo-Mersenne toroidal groups

Yiwen Zhu[1,2]

[1] Yunnan-Malaya Institute, Yunnan University, Yunnan Province, China
zhuyuanzhang1000@gmail.com
[2] Universiti Malaya, Kuala Lumpur, Malaysia zhuyuanzhang1000@gmail.com

Abstract. This paper introduces Torus, a novel block designed to use power mappings over a pseudo-Mersenne prime field $\mathbb{F}_{2^{32}-5}$ to achieve high security and performance. We demonstrate that the cubic power mapping $x^3 \pmod{2^{32}-5}$ possesses low differential uniformity and high nonlinearity. Torus leverages the number-theoretic identity $2^{32} \equiv 5 \pmod{2^{32}-5}$ to induce an end-around carry feedback mechanism. This inherent toroidal diffusion facilities a rapid increase in algebraic degree across the bit-vector space, a property often lacking in pure GF(2^n) power functions. Security evaluations reveal that the algebraic incompatibility between modular addition and bitwise operation effectively thwarts higher-order differential and interpolation attacks by inducing exponential growth in polynomial complexity. Furthermore, cube power mapping over $\mathbb{F}_{2^{32}-5}$ has a high resistance to differential and linear cryptanalysis. The 8-round Torus provides a significant security margin, as 5-round Torus can not only resist differential analysis and linear analysis, but also algebraic attacks. Performance benchmarks indicate that Torus achieves superior throughput on modern superscalar processors.

Keywords: pseudo-Mersenne prime · ARX cipher · power mappings · block cipher

1 Introduction

In the realm of power mappings, the cubic function $f(x) = x^3$ is notable for its Almost Perfect Nonlinear (APN) and Almost Bent (AB) characteristics over binary fields $\mathbb{F}_{2^n}$. It was used in some early block ciphers such as KN [23]. However, since the algebraic degree of the functions is limited to a degree of 2, it remains susceptible to higher-order differential[11] and interpolation attacks[10], necessitating the use of complex algebraic transformations.

In this paper, we propose Torus, a high-throughput block cipher based on power mapping over prime field. Departing from the flat vector space of $\mathbb{F}_{2^n}$, Torus anchors its round function within the pseudo-Mersenne prime field $\mathbb{F}_{2^{32}-5}$ and modular addition over $\mathbb{Z}_{2^{32}}$. It also uses the XOR operation to leverage the algebraic incompatibility of heterogeneous operations, disrupting the algebraic structural symmetry inherent in single-domain cipher. By utilizing the number-theoretic identity $2^{32} \equiv 5 \pmod{2^{32}-5}$, the arithmetic carry is not merely a one-way chain but a toroidal feedback mechanism that reinjects entropy from the most significant bit (MSB) back into the least significant bit (LSB) [14].

The pseudo-Mersenne toroidal group yields two significant cryptographic advantages. First, the cyclic arithmetic carry acts as a "degree booster", forcing the algebraic degree of the cubic mapping to saturate rapidly within the Boolean domain. The algebraic incompatibility between modular addition and XOR prevents the construction of unified polynomial systems, shielding the cipher from algebraic solvers. Therefore, Torus has high resistance to

higher-order differential cryptanalysis [11] and interpolation attacks [10]. Secondly, modular arithmetic over pseudo-Mersenne prime fields facilitates efficient implementation [19].

The remainder of this paper is organized as follows. Section 2 provides the mathematical background on APN, AB mappings and pseudo-Mersenne prime fields. Section 3 describes the design specifications of Torus. In Section 4, we provide a security analysis focusing on its resistance to differential or linear cryptanalysis and algebraic attacks. Section 5 presents performance benchmarks with software implementation.

2 Mathematical foundations and the properties of x^3 over $\mathbb{F}_{2^{32}-5}$

2.1 Almost Perfect Nonlinear (APN) and Almost Bent (AB) functions

For a function $f : \mathbb{F}_{2^n} \to \mathbb{F}_{2^n}$, the differential uniformity δ_f is defined as the maximum number of solutions to the equation

$$f(x \oplus \alpha) \oplus f(x) = \beta$$

for any $\alpha \in \mathbb{F}_{2^n} \setminus \{0\}$ and $\beta \in \mathbb{F}_{2^n}$, and where $\oplus$ denotes bitwise XOR (Boolean addition).

A function is classified Almost Perfect Nonlinear (APN) if $\delta_f = 2$, which is the theoretical minimum for functions over fields of even characteristic. The cubic mapping $f(x) = x^3$ is a well-known APN function over any $\mathbb{F}_{2^n}$ where n is an odd number [21].

While APN properties defend against differential attacks, linear cryptanalysis targets the correlation between input and output bits. This resistance is measured by the Walsh-Hadamard Transform (WHT):

$$W_{f(a,b)} = \sum_{x \in \mathbb{F}_{2^n}} (-1)^{b \cdot f(x) \oplus a \cdot x}$$

The maximum magnitude of these coefficients determines the linearity L_f. A function is Almost Bent (AB) if its Walsh spectrum $W_{f(a,b)}$exclusively takes values from the set $\{0, \pm 2^{\frac{n+1}{2}}\}$. Notably, AB functions are always APN [6].

2.2 The pseudo-Mersenne prime field $\mathbb{F}_{2^{32}-5}$

2.2.1 Cyclic carry property (end-around carry)

A defining characteristic of pseudo-Mersenne fields is the modular reduction identity

$$2^n \equiv c \pmod{2^n - c}.$$

Specifically, for the Torus, where $n = 32$ and $c = 5$, for any $x \in \mathbb{Z}^+$, the reduction can be performed via a bitwise shift and weighted addition[19]:

$$x \pmod{2^{32} - 5} = (x \pmod{2^{32}}) + 5 \cdot \lfloor x/2^{32} \rfloor.$$

In the implementation, this iterative mechanism manifests itself as an end-around carry (EAC), where the overflow bits of the MSB are multiplied by 5 and fed back into the LSB [14]. This provides accelerating diffusion properties in the cipher structure.

2.2.2 Carry-induced algebraic degree escalation

While x^3 in a Boolean field is strictly quadratic (degree 2), its representation in the pseudo-Mersenne field involves a chain of carry bits c_i, where each c_i is a nonlinear Boolean function of all preceding bits:

$$c_{i+1} = (x_i \cdot y_i) \oplus (x_i \oplus y_i) \cdot c_i.$$

Due to the addition carry chain of the cube mapping over$\mathbb{F}_{2^{32}-5}$, output bits rapidly attain a high algebraic degree and dense Algebraic Normal Form (ANF)[8], providing robust immunity against algebra solvers. We discuss this property in detail in Section 4.3.1.

2.2.3 Cryptographic properties of x^3 over $\mathbb{F}_{2^{32}-5}$

Unlike the unidirectional carry propagation in finite rings, $x^3 \pmod{2^{32}-5}$ leverages the EAC mechanism [14]. A single-bit difference in the input triggers a carry chain that propagates through the entire 32-bit word, which ensures rapid diffusion within one round.

Due to the incorporation of round-key addition in modular domain, the difference here is often referred to as an additive difference:

$$f(x+a) - f(x) = b.$$

For $f(x) = x^3 \pmod{p}$, where $P > 3$ is a prime, expanding the equation gives

$$3ax^2 + 3a^2x + (a^3 - b) \equiv 0 \pmod{p}.$$

Since this equation has at most two solutions, the mapping exhibits a differential uniformity of $\delta = 2$, analogous to the APN property but over the prime field instead of the Boolean field.

Although $x^3 \pmod{2^{32}-5}$ is an "arithmetic APN" function, it does not have the APN property when the difference is XOR-based. However, it still has high XOR differential uniformity. The EAC mechanism inherent in pseudo-Mersenne reduction ensures that any bitwise difference δ rapidly diffuses into a dense arithmetic distribution. This causes XOR-based differential paths to be quickly disrupted, making it difficult to construct a high-probability differential path in XOR-based differential analysis. We will provide further details in Section 4.

It should be noted that the cubic mapping $f(x) = x^3 \pmod{2^{32}-5}$ is a permutation over the field $\mathbb{F}_{2^{32}-5}$ because $\gcd(3, 2^{32}-6) = 1$. However, to improve performance, we adopt lazy reduction in the implementation of Torus. This causes the maximum output value to be $2^{32}-1$ instead of $2^{32}-6$, and results in a pseudo-zero point. In Sections 4 and 5, we argue that this does not affect the security and statistical indistinguishability of the ciphertext.

3 The design specifications of Torus

Based on the theoretical foundations established in Section 2, we present Torus, a 128-bit block cipher with a modified parallel Lai-Massey-like structure.

We adopt the following notation to describe the data path and operations.

- X_1, X_2, X_3, X_4: The internal round state, consisting of four 32-bit words, where $X_i \in \mathbb{Z}_{2^{32}}$.
- $\oplus$: Bitwise XOR (Boolean addition).
- $\boxplus$: Addition modulo 2^{32}.
- $\lll s$: Bitwise rotation to the right/left by s bits.
- $x^3 \pmod{2^{32}-5}$: The cubic power mapping (Lazy Reduction)

3.1 Torus-128

The block length of Torus is 128 bits, and its overall structure is a Lai-Massey structure similar to IDEA [15]. The recommended round number of Torus-128 is 8.For Encryption and Decryption Test Vector of Torus-128, please see Appendix I

3.1.1 *F*-box

The round function of Torus can be expressed as a single formula:

$$F(X, K) = [(X \boxplus K)^3 \pmod{2^{32} - 5}]$$

where X is a 32-bit input, and K is a 32-bit subkey. The input range of this function is $[0, 2^{32} - 1]$ and is implemented by Lazy Reduction. Therefore, its outputs are not strictly the result of the cube mapping over $\mathbb{F}_{2^{32}-5}$ for a negligible number of special input values. In Section 4, we argue that this implementation method does not affect security and is conducive to improving performance. Furthermore, this avoids conditional branches and enhances resistance to timing-based side-channel attacks. (For details, see Appendix A.)

3.1.2 *G* function

The function G, defined here, uses low-cost nonlinearity and left rotation to enhance diffusion and block the propagation of differential and linear characteristics. This function is defined by

$$(X_1, X_2, K_1, K_2) \in (\{0,1\}^{32})^4 \longrightarrow (Y_1, Y_2) \in (\{0,1\}^{32})^2$$

$$t_1 = X_1 \boxplus K_1$$
$$Y_2 = X_2 \oplus (t_1 \lll 7 \boxplus t_1)$$
$$t_2 = Y_2 \boxplus K_2$$
$$Y_1 = X_1[\oplus(t_2 \lll 2) \boxplus (t_2 \lll 15)]$$

The inverse transformation G^{-1} is given by

$$(Y_1, Y_2, K_1, K_2) \in (\{0,1\}^{32})^4 \longrightarrow (X_1, X_2) \in (\{0,1\}^{32})^2$$

$$t_1 = Y_2 \boxplus K_1$$
$$X_1 = Y_1 \oplus [(t_1 \lll 2) \boxplus (t_1 \lll 15)]$$
$$t_2 = Y_1 \boxplus K_2$$
$$X_2 = Y_2 \oplus (t_2 \lll 7 \boxplus t_2)$$

3.1.3 Encryption algorithm of Torus-128

First, split the 128-bit block into four 32-bit parts (X_1, X_2, X_3, X_4) and perform whitening

$$(X_1, X_2, X_3, X_4) \leftarrow (X_1 \oplus K_1, X_2 \oplus K_2, X_3 \oplus K_3, X_4 \oplus K_4).$$

Then, apply the transformation G:

$$(X_1, X_2) \leftarrow G(X_1, X_2, K_5, K_6);\ (X_3, X_4) \leftarrow G(X_3, X_4, K_7, K_8).$$

Each round takes as input $X = (X_1, X_2, X_3, X_4) \in (\{0,1\}^{32})^4$, and the round keys $(K_{r1}, K_{r2}) \in (\{0,1\}^{32})^2$ To obtain the output $(Y_1, Y_2, Y_3, Y_4) \in (\{0,1\}^{32})^4$ from the inputs, Then execute the following operations for N rounds.

1. F-box execution:

$$Z_1 = F(X_1 \oplus X_3, K_{r1}); \quad Z_2 = F(X_2 \oplus X_4, K_{r2})$$

2. Cross-addition (Lai-Massey Layer):

$$(Y_1, Y_2, Y_3, Y_4) \leftarrow (X_1 \oplus Z_2, X_2 \oplus Z_1, X_3 \oplus Z_2, X_4 \oplus Z_1)$$

3. Branch rotation:

$$Y_1 = Y_1 \lll 13; \quad Y_4 = Y_4 \lll 17$$

4. Round swapping:

$$(Y_1, Y_2, Y_3, Y_4) \longleftarrow (Y_1, Y_3, Y_2, Y_4)$$

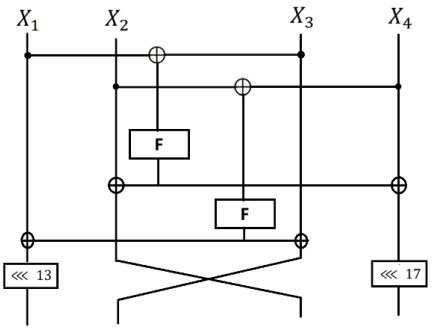


Figure 1: The structure of Torus.

In the final round, round swapping and branch rotation are not performed. Instead, the whitening is performed:

$$(Y_1, Y_2, Y_3, Y_4) \longleftarrow (Y_1 \oplus K_i, Y_2 \oplus K_{i+1}, Y_3 \oplus K_{i+2}, Y_4 \oplus K_{i+3}).$$

This step is followed by G^{-1} transformation:

$$(Y_1, Y_2) \leftarrow G^{-1}(Y_1, Y_2, K_{i+4}, K_{i+5}); \; (Y_3, Y_4) \leftarrow G^{-1}(Y_3, Y_4, K_{i+6}, K_{i+7}).$$

For the case of 8 rounds, $i = 32$.

3.1.4 Torus-256

Torus-256 uses a 256-bit key scheduling algorithm and increases the number of rounds to 10 and the other steps are the same as those in Torus-128.

3.1.5 Decryption algorithm of Torus

Because encryption and decryption are symmetric, decryption follows the same procedure as encryption except that the order of the subkeys is reversed, and left rotations are changed to right rotations.

3.2 Key scheduling algorithm

The key scheduling algorithm consists of a feedback shift register (FSR) and a counter. In the 128-bit key version, the key is split into four 32-bit register parts (R_1, R_2, R_3, R_4). In the 256-bit version, the key is split into eight 32-bit register parts $(R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8)$. The function F used in the key scheduling algorithm below is the same as the round function

$$F(X, K) = (X \boxplus K)^3 \pmod{2^{32} - 5}.$$

The pseudocode for the each round of 128-bit key scheduling algorithm is as follows, use the output R_4 as a subkey,

$$\text{output}(R_4)$$

$$Z \leftarrow R_4 \oplus \text{counter}; \quad \text{counter} \leftarrow \text{counter} + 1$$

$$R_2 \leftarrow R_2 \oplus F(Z, R_3); \quad R_1 \leftarrow R_1 \oplus (Z \lll 13)$$

$$(R_1, R_2, R_3, R_4) \leftarrow (Z, R_1, R_3, R_3).$$

for the 256-bit key scheduling algorithm is as follows , use the output R_8 as a subkey,

$$\text{output}(R_8)$$

$$Z \leftarrow R_8 \oplus \text{counter}; \quad \text{counter} \leftarrow \text{counter} + 1$$

$$R_4 \leftarrow R_4 \oplus F(Z, R_5); \quad R_2 \leftarrow R_2 \oplus (Z \lll 13)$$

$$(R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8) \leftarrow (Z, R_1, R_2, R_3, R_4, R_5, R_6, R_7),$$

where the size of the counter is an 8-bit register and the initial value is 0.

Before generating subkeys, the feedback shift register in the 128-bit version runs idle for 8 rounds, discarding all outputs, before using each output as a subkey. In the 256-bit version, the feedback shift register runs idle for 16 rounds.

4 Security analysis

Traditional automated search tools (e.g. MILP/SAT) are computationally infeasible for Torus due to combinatorial explosion, as the $2^{32} - 5$ state-space and highly complex carry-dependencies of the pseudo-Mersenne toroidal group leads to high computational complexity beyond the capabilities of modern CPUs. [18, 4] In this section, we adopt a conservative wide-trail strategy and a heuristic extrapolation based on a reduced field $\mathbb{F}_{2^8-5}$ for security evaluation. We also use a genetic algorithm with multi-start search and pruning (see Appendix B for details) to estimate the maximum differential probability and linear bias. By assuming every active F-function achieves its estimated theoretical maximum differential probability and linear bias, we derive a conservative upper bound on the total characteristic probability. This analytical approach ensures a robust security margin that remains valid even against the most optimal paths that heuristic tools might fail to identify.

Because the implementation of Torus differs from that of the standard cubic mapping, all subsequent experiments are based on the variant implemented in Torus. Unless otherwise specified, all functions $f(x) = x^3 \pmod{2^n - 5}$ mentioned here refer to the variant used in Torus with lazy reduction.

4.1 Differential analysis

Differential cryptanalysis, first formally introduced by Biham and Shamir [3], remains one of the most powerful tools for evaluating the security of block ciphers. The core of this method lies in tracking the evolution of input differences δ_x through non-linear transformations of the cipher to find a specific output difference δ_y that occurs with a probability $P > 2^{-(n+1)}$, where n is the block length of the cipher. Traditional differences are defined as $\Delta x = x \oplus x'$ [3], while in Torus, modular addition is used with subkeys. Therefore, in this section we discuss both the traditional XOR difference and the additive difference $\Delta x = x - x'$.

Before our security analysis of Torus, we first consider a weakness in the Lai-Massey-like structure. In this structure, differences all enter the F-box in the form of XORs like $X_1 \oplus X_2$. Therefore, the ability to maintain the differential state (a, a, a, a) from start to finish would pose a significant threat to this structure. However, in Torus, each round performs different and coprime rotations on the first and fourth branches, making it impossible for any non-zero or all-ones differential (a, a, a, a) to remain unchanged after a round. Moreover, the G layer also disrupts this symmetric differential pattern, so if an all-ones differential is used, it will be completely diffused after passing through the G layer.

Next, we explore the differential properties of F-box in detail. In Section 2.2.2, the mapping $f(x) = x^3$ over $\mathbb{F}_{2^{32}-5}$ exhibits a differential uniformity of $\delta = 2$ when the difference is defined as $f(x + a) - f(x) = b$. Therefore, in this case, the optimal differential probability of the function $f(x) = x^3$ is about 2^{-30}. However, in the practical implementation of software, x is defined over the cyclic group $\mathbb{Z}_{2^{32}}$, which is slightly larger than the primary field $\mathbb{F}_{2^{32}-5}$. This domain mismatch implies that the differential space exceeds the algebraic field's boundary, leading to a subtle degradation of the ideal differential properties. Specifically, the arithmetic difference $\Delta y = (y_2 - y_1) \pmod{2^{32}}$ fails to perfectly preserve the algebraic structure of $\mathbb{F}_{2^{32}-5}$. Our experimental analysis on miniaturized models (e.g. $\mathbb{F}_{251}$ vs. $\mathbb{Z}_{256}$) shows that this expansion of the differential space introduces "boundary collisions". For instance, we observed a maximum differential hit of 12 in an 8-bit model. Nevertheless, as the word size n is increased to 32, this boundary effect is exponentially diluted by the massive output space. We also used a genetic algorithm with multi-start Search and pruning to verify this conclusion. The algorithm showed that no differential characteristic with a probability exceeding 2^{-29} was found in 100 generations of iteration, and there was no significant difference in differential probabilities between the early and the last few generations. Based on this, we conservatively estimate that the additive differential upper bound of this function does not exceed 2^{-25}. Furthermore, replacing addition differences with XOR differences yielded similar results except that one XOR difference 0xFFFFFFFB $\rightarrow$ 0xFFFE0003 had a probability of 2^{-16}, since the input domain $[0, 2^{32}-1]$ is larger than the prime modulus $p = 2^{32}-5$. However, this path does not constitute a threat because the difference 0xFFFFFFFB has a very high Hamming weight, with 30 active bits. So whether it remains unchanged after key modular addition with itself, or is produced by other differences after modular addition, the corresponding probability is less than 2^{-29} [16]. After passing through x^3, the probability of this characteristic drops to 2^{-45}, which is lower than any other differential characteristic.

No probabilities exceeding $2^{-26.9}$ were found among 100 generations of offspring. Similarly, there is no noticeable difference in the probability distribution between the initial several generations and the last few generations. Based on this, we estimate that the upper bound for XOR differential probability is also 2^{-25}, though further investigation is necessary.

Notably, the lazy reduction implementation of the function $f(x) = x^3 \pmod{2^8 - 5}$ is, in some aspects, superior to the strictly implemented version in its XOR differential properties over $\mathbb{F}_{2^8-5}$. For example, the maximum differential probability of this function is about $2^{-3.8}$, lower than 2^{-3} of the original function.

Furthermore, because the lazy reduction implementation results in five duplicate points, this function is not a permutation. Mathematically, this occurs when $(x \oplus a)^3 \equiv x^3 \pmod{2^{32}-5}$. Our experiment shows that for a fixed a, the above equation has at most 24 solutions. Therefore, the maximum probability of differential collision is $\frac{24}{2^{32}-5} \approx 2^{-27.4}$. This can be exploited to a construct a 2-round iterative differential path assuming the branch rotation and G are bypassed:

$$(a,0,a,0) \overset{P=1}{\longrightarrow} (a,a,0,0) \overset{P=2^{-54.8}}{\longrightarrow} (a,0,a,0),$$

where a is a non-zero difference.

In this case, we consider an input difference $\Delta X = (a,0,a,0)$, where $a \in \mathbb{F}_{2^{32}-5}$ is a non-zero differential value. The input to the first F-box is calculated as $(\Delta X_1 \oplus \Delta X_3)$ or $(\Delta X_2 \oplus \Delta X_4)$. In this specific configuration, the differences cancel each other out, resulting in a zero difference to the input of the F-box. Consequently, the F-box outputs a zero difference with probability $P = 1$. After the branch exchange, the state becomes $(a,a,0,0)$. In round 2, The difference will enter two F functions, at which point we introduce a zero difference, assuming that the output differences of both F-boxes are zero. According to the previous derivation of the probability of zero differences, the probability of this step is $2^{-27.4}$. Then, the state becomes $(a,0,a,0)$ again, so the probability of this differential characteristic is about $(2^{-27.4})^2 = 2^{-54.8}$, ignoring the effect of modulo addition and branch rotation. However, this characteristic is not feasible in practice because the branch rotation and G will rapidly change a to some other difference.

We derived another 3-round XOR iterative difference characteristic

$$(a,0,a,0) \to (a,a,0,0) \to (b,c,d,e) \to (f,0,f,0).$$

For the characteristic diagram, see Appendix C. If we assume that the differential probability of each active F-box reaches its estimated maximum 2^{-25} and bypasses the effect of modulo addition and the G layer, the probability of this characteristic is 2^{-100}.

As before, we consider an input difference $\Delta X = (a,0,a,0)$, where $a \in \mathbb{Z}_{2^{32}}$ is a non-zero differential value. The input to the first F-box is calculated as $(\Delta X_1 \oplus \Delta X_3)$ or $(\Delta X_2 \oplus \Delta X_4)$. Due to the XOR-sum property of the branches, the inputs to both F-boxes cancel out to zero. Consequently, the F-boxes output a zero difference with probability $P = 1$, maintaining the state structure for the subsequent rounds. After the branch exchange and rotation, the state becomes $(a',a,0,0)$. In round 2, the two differences will each enter an F-box, yielding as output the differences e and d with certain probability. We assume that this step reaches the theoretical maximum differential probability for the F-box with $P = (2^{-25})^2 = 2^{-50}$. Then, according to the characteristics of the Lai-Massey structure, the state becomes (b,d,c,e). In the final stage of this iterative segment, the differences undergo another cancellation. The differences $(b \oplus c)$ and $(d \oplus e)$ each enter an F-box. We assume that the differences x and y that result in $b \oplus y = (f \ggg 13)$, $c \oplus y = 0$ and $d \oplus x = f$, $e \oplus x = 0$ are output with the maximum differential probability $P = (2^{-25})^2 = 2^{-50}$. Therefore, after the branch exchange and rotation, the state becomes $(f,0,f,0)$, thus returning to the same form as the initial state: $(a,0,a,0)$. The total probability of this differential characteristic is $P^2 = 2^{-100}$.

The 3-round or 4-round differential path constructed basing this differential characteristic is as follows:

$$(a,0,a,0) \overset{P=1}{\longrightarrow} (a',a,0,0) \overset{P=2^{-50}}{\longrightarrow} (b,d,c,e) \overset{P=2^{-50}}{\longrightarrow} (f,0,f,0) \overset{P=1}{\longrightarrow} (g,h,i,j).$$

The estimated probabilities of this 3- or 4-round differential path are both 2^{-100}. With 5-round, the probability drops to $2^{-150} < 2^{-128}$. If additive differences are used instead of XOR differences, the probability drops further because the additive differences of the F-box are more uniform.

For all the construction methods used here, the actual probability of the differential characteristic in practical situations is much lower than the conservative estimates we provide here. Firstly, the differential cannot reach the estimated extreme probability $P = 2^{-25}$ of the F-box at every step. Secondly, the problem of XOR and modular addition incompatibility, which is ignored here, actually causes the differential probability to decrease rapidly in practice [16]. The nonlinear logic of the G layer can also hinder the construction of the initial differential. Specifically, given the bit-rotations $(\lll 5, \lll 7)$, the active bits are reshuffled, preventing any high-probability differential clusters. This ensures that the G layer provides an additional security margin by effectively truncating the differential trails. Therefore, the probability of 3-round or 4-round differential paths will be far lower than 2^{-100}.

Furthermore, we consider the differential hull effect, where multiple differential characteristics share the same input and output differences [7]. This effect typically enhances the attacker's advantage by summing the probabilities of all possible internal paths. However, because the probability of any single differential characteristic is extremely low, their contributions to each hull are negligible. In addition, during our multi-start search with pruning, the probabilities of almost all discovered differential characteristics fell within the interval $(2^{-28}, 2^{-27})$, indicating that the differential distribution is uniform. This suggests that the clustering of high-probability paths is prevented, violating the implementation conditions of the differential hull. Therefore, the differential hull does not cause marginal degradation to security.

In summary, Torus does not have exploitable differential paths through 4 or more rounds in practical scenarios.

4.2 Linear analysis

Linear cryptanalysis, originally proposed by Matsui [17], is a powerful known-plaintext attack that exploits high-probability linear approximations between input and output bits of a cipher. For a cryptosystem to be secure, the linear probability (LP), often measured by the linear bias ϵ, must be sufficiently low. A block cipher is theoretically susceptible to threats from linear analysis when $\epsilon > 2^{-\frac{n}{2}}$ where n is the block length of cipher.

To evaluate the linear properties of $f(x) = x^3 \pmod{2^n - 5}$ over the field $\mathbb{F}_{2^8-5}$, we conducted an exhaustive search over all possible inputs $\alpha \in \{1, 2, \ldots, 2^8 - 1\}$ and output masks $\beta \in \mathbb{F}^*_{2^8-5}$, where * indicates that the zero element is excluded. The experimental results reveal that the maximum linear bias is $\epsilon_{\max} = 0.152344 \approx 2^{-2.7}$. In addition, we adopt a genetic algorithm with multi-start search and pruning to estimate the maximum linear bias of $x^3 \pmod{2^{32}-5}$, where $x \in 1, 2, \ldots, 2^{32} - 5$. The results show that, after the initial few generations, the maximum linear bias consistently remained at approximately 2^{-15}. Based on the above results, we conservatively estimate that the maximum bias of $f(x)$ will not exceed $2^{-13.5}$. Specifically, we identify an efficient 2-round iterative linear characteristic for the Torus. Let the linear mask be denoted by $\Gamma = (\gamma_1, \gamma_2, \gamma_3, \gamma_4)$. The linear characteristic trail is

$$\Gamma_{\text{in}}(a, 0, b, 0) \to \Gamma_{\text{mid}}(c, b, b, d) \to \Gamma_{\text{out}}(e, 0, f, 0),$$

where a, b, c, d represent non-zero masks in $\{1, 2, \ldots, 2^{32} - 1\}$. The corresponding schematic diagram is shown in Appendix D. This iterative characteristic activates a total of three F-boxes. We calculate the linear bias of each F-box based on the estimated peak value $\epsilon_{\max} = 2^{-13.5}$. According to the Matsui's Piling-up Lemma [16], the total linear bias is

$$\epsilon_{total} = 2^{n-1} \prod_{i=1}^{n} \epsilon_i = 2^{-38.5}.$$

According to the structural characteristics, the initial mask $\Gamma(a, 0, b, 0)$ will generate a mask: $a \oplus b$, and it becomes b after passing through the F-box by using linear approximation:

$$(a \oplus b) \cdot X \oplus b \cdot F(X) = 0$$

Then the state becomes$\Gamma(a \oplus b = c', b, b, b)$ and branch rotation transforms it into $\Gamma(c, b, b, d)$. This step activates an F-box whose bias is assumed to be the peak value $\epsilon_{\max} = 2^{-13.5}$. Then, we try to return these masks to their initial state. We use the linear approximation $(c \oplus b) \cdot X \oplus d \cdot F(X) = 0$ for the first F-box, and the linear approximation $(b \oplus d) \cdot X \oplus b \cdot F(X) = 0$ for the second. Using the properties of masks, $d \oplus d$ and $b \oplus b$ cancel out, yielding 0, while $c \oplus b$ and $b \oplus d$ produce new non-zero masks e' and f. Therefore, after the F-box of the second round, the state $\Gamma(c, b, b, b)$becomes $\Gamma(c \oplus b = e', b \oplus d = f, b \oplus b = 0, d \oplus d = 0)$. Branch rotation and round swapping transform the state to $\Gamma_{\text{out}} = (e, 0, f, 0)$. This step will active two F-box, bias is $\epsilon = 2^{-27}$. The total linear bias is

$$\epsilon_{total} = 2^2 \cdot 2^{-27} \cdot 2^{-13.5} = 2^{-38.5}$$

Using this 2-round iterative linear feature, a 3-round linear approximation:

$$\Gamma(a, 0, b, 0) \overset{\epsilon=2^{-13.5}}{\longrightarrow} \Gamma(c, b, b, d) \overset{\epsilon=2^{-27}}{\longrightarrow} \Gamma(e, 0, f, 0) \overset{\epsilon=2^{-13.5}}{\longrightarrow} \Gamma(g, f, f, h)$$

of the Torus can be constructed. According to Matsui's Piling-up Lemma [17], the maximal linear bias is

$$\epsilon_{\text{total}} = 2^{n-1} \prod_{i=1}^{n} \epsilon_i = 2^3 \cdot (2^{-13.5})^2 \cdot 2^{-27} = 2^{-51}.$$

The data complexity of attacks using the above linear discriminator is $\frac{1}{\epsilon_{\text{total}}^2} \approx 2^{102}$. If there is one more round, the total linear bias becomes $2^{-76} < 2^{-64}$,making the discriminator fail completely.

As in the case of differential cryptanalysis, the actual data complexity of the aforementioned attack is far higher than the conservative estimate given, because our estimate ignores the effects of the nonlinear bitwise logic of the G layer and key modular addition. These components cause extremely rapid decay of linear bias. Furthermore, the linear bias at each step is assumed to be at its maximum value, which is virtually impossible to achieve in practice.

In addition, we also considered linear hull effects [22]. In our genetic algorithm with multi-start search, biases of all tested linear approximation fell within the interval $(2^{-16}, 2^{-15})$, which shows that the correlation spectrum of $f(x) = x^3 \pmod{2^{32} - 5}$ is uniform, and the biases are generally quite low. Therefore the improvement provided by linear hulls on linear characteristics is negligible.

In conclusion, linear cryptanalysis does not pose a threat to Torus beyond 4 rounds, in practice.

4.3 Algebraic-type attacks

4.3.1 Mathematical properties of the F-box

We tested the Boolean algebraic properties of the core mapping of the F-box $f(x) = x^3 \pmod{2^{32} - 5}$, $x \in [0, 2^{32} - 1]$. We found that the algebraic degree of all bits reached 31 or 32 after one round of mapping, and the polynomial density of all bits remained around 0.5. For detailed data tables, please refer to Appendix E.

While the Boolean algebra properties of this function are very complex, its arithmetic algebraic degree is only 3. However, after cross-domain operations and iterations, both the arithmetic algebraic degree and the polynomial density quickly saturate due to the

cross-domain operations. We tested the algebraic properties of its iterative function $f(f(x) \oplus x)$ and found that its algebraic degree reached 32, and the polynomial density was approximately 0.9999.

4.3.2 Higher-order differential analysis and the interpolation attack

Higher-order differential attacks exploit the low algebraic degree of a cipher's round function [11]. Our experimental results demonstrate that the algebraic degree of output bits of the F-box reaches $d = 32$ or 31 within a single round. This rapid saturation of algebraic degree ensures that the complexity of constructing a successful higher-order differential exceeds the brute-force threshold within 2 rounds of the cipher.

Interpolation attacks are effective when the cipher can be represented by a polynomial with a relatively small number of non-zero terms or a polynomial of low algebraic degree [4]. Our experiment shows that each output bit's ANF contains approximately 2,147,000,000 non-zero terms, achieving a density of around 0.5. This high-density property, coupled with the maximum algebraic degree, renders interpolation attacks computationally infeasible within ony a few round.

When using arithmetic interpolation and high-order differences, this is still not feasible due to the rapid saturation of the arithmetic algebraic degree within two iterations as our experiment shows.

4.3.3 Integral attacks

Traditional integral cryptanalysis [13] relies heavily on the bijective nature of the component functions to maintain balance. However, the F-box is an approximately quasi-bijective function, making it impossible to establish a deterministic integral distinguisher over the entire set. Furthermore, the algebraic degree of output bits of F-box exceed the condition limit of high-order integral attacks [13] within a single round. Consequently, the integral distinguishers are disrupted and become ineffective after the first or second rounds.

4.4 Other attacks

4.4.1 Impossible differential analysis

The efficacy of an impossible differential attack relies on finding a differential path with a probability of exactly zero over several rounds [12].

The end-around carry (EAC) induces a cyclic dependency that interlinks the MSB and LSB, causing any local bit-difference to avalanche into a global arithmetic perturbation. Furthermore, the Rotation and modular addition cause the differential to spread rapidly, further increasing the difficulty of finding impossible differentials (ID).

We consider a 16-bit mini Torus with 4 rounds based on the function

$$F(X, K) = (X + K \pmod{256})^3 \pmod{2^8 - 5},$$

with a specific key. Details of this specification are given in Appendix F. By exhaustively listing all impossible differential pairs of this 16-bit mini Torus, we verify whether they hold under different independent keys. The results show that for each additional key, the number of impossible differential pairs decreases by approximately 60% (For detailed data, see Appendix G). The number of impossible differential pairs decreases from approximately 2,600,000,000 at a single key to 0 when 45 independent keys are added (ignoring the trivial case). Furthermore, we found that while there are approximately 1,570,000,000 additively impossible differential pairs under a single key, there are no such pairs under 23 independent keys. This demonstrates 4-round Mini Torus does not have structurally impossible differentials independent of the key, indicating that components of a scaled-down

Torus have strong impossible differential resistance. However, as the bit size increases, the addition carry chain becomes longer, and diffusion becomes more uniform. Therefore, components of the same nature with larger sizes have stronger resistance to impossible differentials. In conclusion, impossible differential analysis does not pose a threat to high-round Torus.

4.4.2 Zero-correlation linear analysis references

To successfully launch a zero-correlation linear attack, an adversary must satisfy two stringent criteria: the existence of non-zero linear mask pairs (α, β) such that the correlation C of the linear approximation $\alpha \cdot P \oplus \beta \cdot C = 0$ is exactly zero, and the universality of these masks across the entire key space [5].

We exhaustively list all zero-correlation linear pairs of the Mini Torus under a specific key. These pairs were then tested with different independent keys to verify their key-dependency. Experimental results demonstrate that while approximately 13,400,000 zero-correlation (ZC) candidate pairs exist for a single key, these pairs do not persist across independent keys. The number of common ZC pairs rapidly converges to 0 as more keys are introduced; specifically, our observations show that 5 independent keys to reduce the number of ZC pairs to zero, suggesting the absence of structural zero-correlation.This indicates that even in cases where the bit size is reduced, the addition carry chain is shorter, and diffusion is less uniform, components of Torus still possess excellent resistance to zero-correlation linear analysis. In summary, zero-correlation linear analysis does not pose a threat to high-round Torus.

4.4.3 Related-key analysis

Unlike standard attacks in which the key is fixed, this attack exploits vulnerabilities in the key schedule to identify patterns or to leak information about the secret key [2]. The Torus key schedule effectively disrupts the propagation of differences between two related keys K and K' by incorporating a monotonic 8-bit counter into the transformation. This injection of a time-varying constant breaks the structural symmetry of the key expansion, ensuring that even if an adversary introduces a fixed difference ΔK, the resulting subkey differences ΔZ will vary across each step. This mechanism specifically neutralizes slide attacks and prevents the construction of long-range related-key differential paths.

Furthermore, the non-linear transformation of Key scheduling algorithm: $R_2 = R_2 \oplus F(Z, R_3)$ and the cyclic rotation $R_1 = R_1 \oplus (Z \lll 13)$ ensure rapid diffusion of key differences. The function F acts as a non-linear filter that prevents the linear cancellation of differences, while the word-level shuffling ensures that every bit of the master key influences the entire subkey state within a few iterations. Consequently, the high algebraic degree of the transformation and the elimination of periodic structures ensures that the subkeys are statistically independent, providing robust immunity against related-key analysis.

4.5 Statistical and diffusion tests

Torus has passed fifteen statistical tests of the NIST SP 800-22 [1]. For details please see the Appendix H. We tested the avalanche effect of Torus, and the results show that when one bit of the plaintext is inverted, the probability of each bit of the ciphertext being inverted is 50.17%. When one bit of the key is inverted, the probability of each bit of the ciphertext being inverted is 49.86%. This indicates that Torus satisfies the avalanche effect.

5 Performance testing

5.1 Throughput testing of the software implementation

To study the performance of the Torus software implementation, we chose AES [20], RC6 [24], and SIMON [9] for a throughput comparison. AES is currently the most widely used encryption algorithm, while RC6 represents high performance in software-interleaved implementations, and SIMON is representative of ARX-type ciphers. All the above algorithms were implemented in software during the tests, and AES was implemented without hardware acceleration. The results show that although Torus128 has a lower speed than AES128 without interleaving, its throughput exceeds both AES and SIMON in 2-way and 4-way interleaving, and comes close to RC6 in 4-way interleaving. The throughput of Torus256 significantly surpasses that of AES256(software-only) indicating that Torus has high software performance. The specific data is shown in Figure 2.

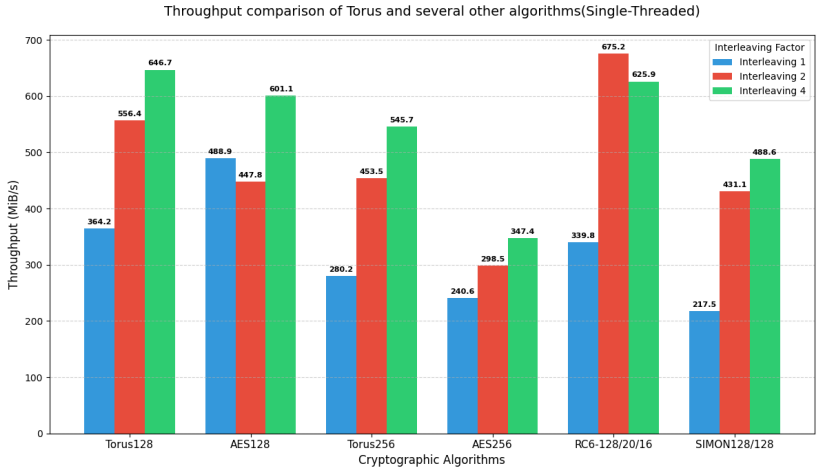


Figure 2: Throughput testing of the Torus software implementation.

These benchmarks were performed on the following hardware and software: AMD Ryzen 7 H255, 32GB LPDDR5x-7500 memory, Windows 11. Compilation was via GCC 13 (MinGW 11.0) with -O3 -march=native. All results are single-threaded throughput in MiB/s.

6 Conclusion

In this paper, we presented Torus, a novel block cipher leveraging cubic power mappings over the pseudo-Mersenne prime field $\mathbb{F}_{2^{32}-5}$. The core of our design lies in the exploitation of the number-theoretic identity $2^{32} \equiv 5 \pmod{2^{32}-5}$, which facilitates a unique toroidal diffusion mechanism that significantly accelerates the bit-level dependency across the state.

Our security analysis demonstrates that Torus provides a robust security margin. Specifically, we identified a 4-round differential characteristic based on Genetic algorithm and wide-trail strategy, which confirms that the full 8-round configuration offers substantial resistance against differential cryptanalysis and linear cryptanalysis. Furthermore, the inherent algebraic incompatibility between modular arithmetic and bitwise XOR operations ensures that the algebraic degree saturates rapidly, effectively thwarting higher-order differential and interpolation attacks within the early stages of the cipher.

Performance evaluations show that Torus is highly optimized for modern superscalar architectures, achieving a superior throughput-to-security ratio.The combination of its strong

mathematical foundation and high implementation efficiency makes Torus a compelling candidate for high-speed cryptographic applications.

Acknowledgements

I would like to express my sincere gratitude to Dr. Ian from the University of Malaya for his invaluable time and effort in proofreading and editing this manuscript. His meticulous feedback and insightful suggestions have significantly enhanced the clarity and quality of my work. I truly appreciate his generosity and expertise, which have been a great support throughout this process.

Appendices

A Lazy Reduction of $F(X, K) = X^3 \pmod{2^{32} - 5} \boxplus K$

The following code implements the lazy reduction of the power mapping used in the main text.

```
inline uint32_t fast_reduce(uint64_t z) {
    uint64_t z_lo = (uint32_t)z;
    uint64_t z_hi = z >> 32;
    uint64_t r = (z_hi << 2) + z_hi + z_lo;
    r = (r >> 32) * 5 + (uint32_t)r;
    return (uint32_t)r;

}

inline uint32_t cube(uint32_t x, uint32_t k) {
    uint64_t sq = x * x;
    uint64_t sq_red = fast_reduce(sq);
    uint32_t cb = fast_reduce(sq_red * x);
    return cb+k;
}
```

Lazy Reduction of Torus

B Genetic algorithm with multi-start search and pruning

This multi-start genetic algorithm with adaptive pruning is designed to identify optimal differential characteristics for cryptographic functions through a hybrid heuristic-stochastic search. The process begins with a population of 128 input differentials: 32 are manually curated "weak" differentials (including single-bit active, all-ones, alternating bit patterns, and boundary points), while 96 are generated randomly. During each generation, the algorithm performs an initial full-space traversal for each input difference to identify the top five candidates based on probability, and then generates variants via bit-flip mutation and cyclic shifts. Reproduction occurs via two distinct crossover operators. To maintain computational efficiency, a pruning strategy is applied to the offspring: each candidate is first subjected to a statistical sampling of 4 million random points, and only those with a differential uniformity count greater than 2 proceed to exhaustive full-space verification. After identifying the top five candidate, a new generation is produced based on the previous variations and crossbreeding. This iteration is repeated 100 times. This "fast-fail" mechanism ensures that computational resources are concentrated on high-probability candidates. Over 100 generations, the algorithm iteratively refines the population, retaining only superior offspring to converge on the maximum differential characteristic.

C Iterative differential characteristic

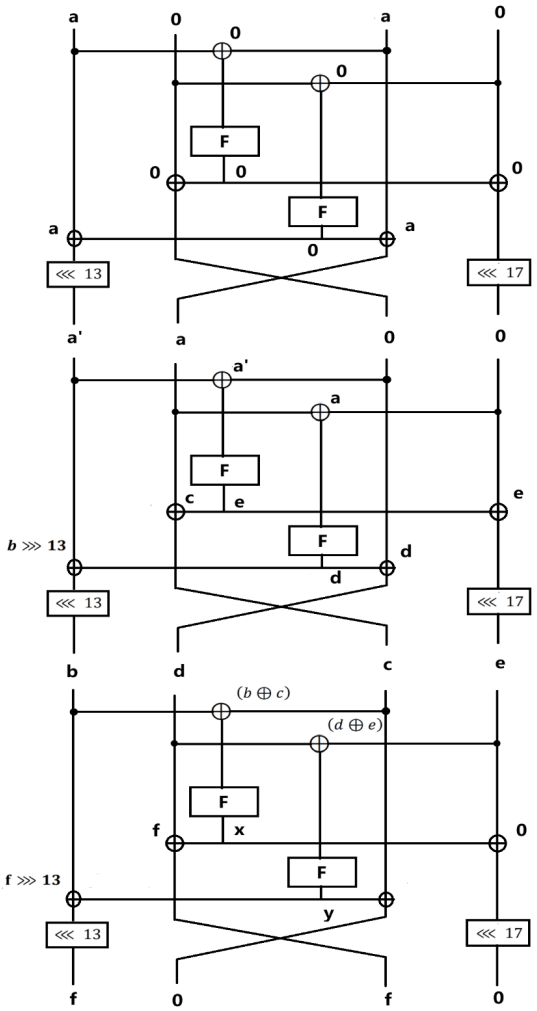


Figure 3: Iterative differential characteristic.

D Linear characteristic

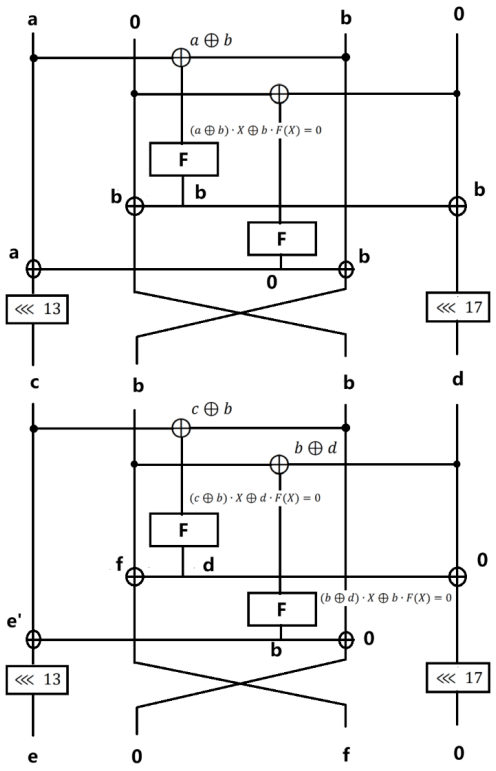


Figure 4: Iterative linear characteristic.

E The algebraic properties of $x^3 \pmod{2^{32}-5}$ used in Torus

Table 1: Iterative linear characteristic.

Bit Index	Algebraic Degree	Non-zero Terms	Polynomial Density
0	32	2147441992	0.4999903012
1	31	2147484058	0.5000000955
2	31	2147491661	0.5000018657
3	32	2147446221	0.4999912858
4	31	2147517890	0.5000079726
5	31	2147475505	0.4999981041
6	31	2147481162	0.4999994212
7	31	2147503338	0.5000045844
8	32	2147478268	0.4999987474
9	31	2147488766	0.5000011916
10	31	2147521886	0.5000089030
11	31	2147450446	0.4999922696
12	31	2147522290	0.5000089970
13	32	2147460254	0.4999945532
14	31	2147491902	0.5000019218
15	32	2147470254	0.4999968815
16	31	2147482518	0.4999997369
17	31	2147523778	0.5000093435
18	31	2147474656	0.4999979064
19	32	2147467830	0.4999963171
20	31	2147468500	0.4999964731
21	31	2147425562	0.4999864758
22	31	2147497486	0.5000032219
23	32	2147504532	0.5000048624
24	32	2147454756	0.4999932731
25	32	2147491012	0.5000017146
26	32	2147511448	0.5000064727
27	32	2147412224	0.4999833703
28	32	2147445022	0.4999910067
29	32	2147474146	0.4999977876
30	31	2147422784	0.4999858290
31	32	2147500392	0.5000038985

F 4-round Mini-Torus

Mini-Torus is a block cipher with a 16-bit block length and 4 rounds. It is based on a double-branch Lai-Massey structure. Its F-box is

$$F(X,K) = (X + K \pmod{256})^3 \pmod{2^8 - 5},$$

where $X, K \in [0, 255]$. Its G layer is the same as Torus, except that all 32-bit data is changed to 8-bit data. Mini Torus, like Torus, performs the G transformation at the beginning of the first round and at the end of the last round. The pseudo-code of its round structure is as follows:

$$\begin{aligned} Z &\leftarrow F(L \oplus R, K) \\ L &\leftarrow (L \oplus Z) \lll 2 \\ R &\leftarrow R \oplus Z. \end{aligned}$$

G Convergence Rate of Impossible Differential Pairs

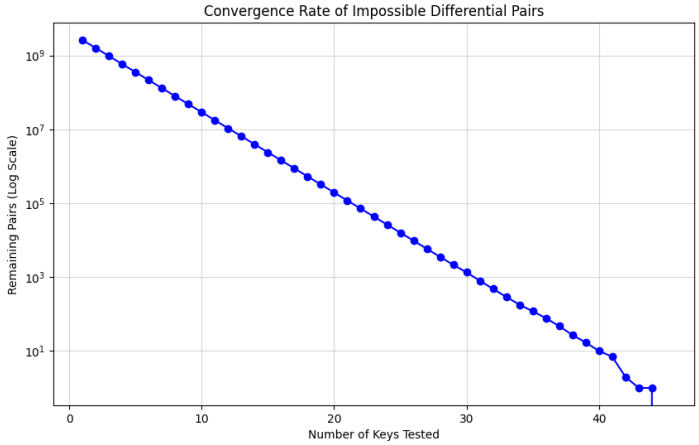


Figure 5: Convergence Rate of Impossible Differential Pairs

H Statistical Test Results

Table 2: NIST SP800-22 Statistical Test Result

NIST SP 800-22 Statistical Test Results for Torus-128

No.	NIST Statistical Test	P-Value	Result
01	Frequency Test (Monobit)	0.0455	**Pass**
02	Frequency Test within a Block	0.7308	**Pass**
03	Run Test	0.0757	**Pass**
04	Longest Run of Ones	0.2326	**Pass**
05	Binary Matrix Rank	0.4350	**Pass**
06	Discrete Fourier Transform	0.5326	**Pass**
07	Non-Overlapping Template	0.4225	**Pass**
08	Overlapping Template	0.1292	**Pass**
09	Maurer's Universal Statistical	0.5808	**Pass**
10	Linear Complexity	0.0248	**Pass**
11	Serial Test (P1/P2)	0.7456 / 0.3086	**Pass**
12	Approximate Entropy	0.2154	**Pass**
13	Cumulative Sums (Forward)	0.0835	**Pass**
14	Cumulative Sums (Reverse)	0.0731	**Pass**
15	Random Excursions (Min P)	0.0665	**Pass (8 sub)**
16	Random Excursions Var (Min P)	0.0812	**Pass (18 sub)**

I Encryption and Decryption Test Vector of Torus-128

Plaintext: ['0x11111111', '0x22222222', '0x33333333', '0x44444444']
Key: ['0x1234567', '0x89abcdef', '0xfedcba98', '0x76543210']

the state of every step:
Initial whitening: ['0x4000e08d', '0x29ad7cf1', '0xea870fd9', '0xa739d23']
Round 1 : ['0xdef165a7', '0xb0a41358', '0xd15e6e8e', '0x2a72b980']
Round 2 : ['0x5e3a1660', '0x91465a38', '0xa3a92a32', '0xb7cae27b']
Round 3 : ['0xc59f2303', '0x26da432e', '0x83ea2701', '0x71bab2e']
Round 4 : ['0xeecc4244', '0x381389a5', '0x28c8d517', '0xe5137409']
Round 5 : ['0xc3753a9d', '0x49b9ff0b', '0x35b9c0ae', '0x22855970']
Round 6 : ['0xdb91d7c6', '0x149a5f9c', '0xa5bea5ab', '0x44f2ac36']
Round 7 : ['0xdc150744', '0x37fea6b9', '0x4ea53447', '0x3346855']
Round 8 : ['0x5c3ebc28', '0x6c782b98', '0xce8e8f2b', '0x58b2e574']
Final ciphertext: ['0x1810ece9', '0x7c7c3d6', '0x6fdd5b4d', '0xfe124fb']

32 subkey (K0-K31)
K00: 76543210 K01: FEDCBA98 K02: 3F99FB25 K03: 85907281
K04: 3E1D34E6 K05: ED731489 K06: ECFD2880 K07: 743FA319
K08: 6E45336F K09: F96C8510 K10: A97857C9 K11: 5BB867C9
K12: 6B3CDE81 K13: A61D3AE6 K14: 38F4DE56 K15: DC9BA46F
K16: 6B725FA4 K17: 1A714524 K18: F04041A7 K19: DBEF3987
K20: B2B95319 K21: 5DB39FBB K22: D3152023 K23: C56BD263
K24: 442E50C1 K25: 6BBFE84E K26: A153D466 K27: 5753C18F
K28: 0E6F9E61 K29: ABEC1ABA K30: 00291E44 K31: 0874AD41

References

[1] Lawrence E. Bassham, Andrew Rukhin, Juan Soto, James Nechvatal, et al. A statistical test suite for random and pseudorandom number generators for cryptographic applications. Technical Report Special Publication 800-22 Revision 1a, National Institute of Standards and Technology (NIST), 2010.

[2] Eli Biham. New types of cryptanalytic attacks using related keys. *Journal of Cryptology*, 7(4):229–246, 1994.

[3] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of Cryptology*, 4(1):3–72, 1991.

[4] Alex Biryukov and Vesselin Velichkov. Automatic search for differential trails in arx ciphers. In Josh Benaloh, editor, *Topics in Cryptology – CT-RSA 2014*, pages 227–250, Cham, 2014. Springer International Publishing.

[5] Andrey Bogdanov and Vincent Rijmen. Zero-correlation linear cryptanalysis of block ciphers. *IACR Cryptol. ePrint Arch.*, 2011:123, 2011.

[6] Florent Chabaud and Serge Vaudenay. Links between differential and linear cryptanalysis. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 356–365. Springer, 1994.

[7] Joan Daemen. Cipher and hash function design strategies based on linear and differential cryptanalysis. 1995.

[8] S. M. Dehnavi and Alireza Rahimipour. Algebraic properties of modular addition modulo a power of two. *IACR Cryptol. ePrint Arch.*, 2014. Report 2014/127.

[9] ISO/IEC. ISO/IEC 29192-2:2019. Information security — Lightweight cryptography — Part 2: Block ciphers, 2019.

[10] Thomas Jakobsen and Lars R Knudsen. The interpolation attack on block ciphers. In *International workshop on fast software encryption — FSE 1997*, pages 28–40. Springer, 1997.

[11] Lars R Knudsen. Truncated and higher order differentials. In *International Workshop on Fast Software Encryption — FSE 1994*, pages 196–211. Springer, 1994.

[12] Lars R Knudsen. DEAL - a 128-bit block cipher. Technical report, NIST AES report, 1998.

[13] Lars R Knudsen and David Wagner. Integral cryptanalysis. In *International Workshop on Fast Software Encryption — FSE 2002*, pages 112–127. Springer, 2002.

[14] Donald Ervin Knuth. *The Art of Computer Programming, Volume 2: Seminumerical Algorithms*. Addison-Wesley, 3rd edition, 1997.

[15] Xuejia Lai and James L Massey. A proposal for a new block encryption standard. In *Workshop on the Theory and Application of Cryptographic Techniques*, pages 389–404. Springer, 1990.

[16] Helger Lipmaa and Shiho Moriai. Efficient algorithms for computing differential properties of addition. In *International Workshop on Fast Software Encryption*, pages 336–350. Springer, 2002.

[17] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In *Advances in Cryptology — EUROCRYPT '93*, pages 386–397. Springer, 1994.

[18] Nicky Mouha, Qingju Wang, Dawu Gu, and Bart Preneel. Differential and linear cryptanalysis using mixed-integer linear programming. In Chuan-Kun Wu, Moti Yung, and Dongdai Lin, editors, *Information Security and Cryptology*, pages 57–76, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[19] Kaushik Nath and Palash Sarkar. Efficient arithmetic in (pseudo-)mersenne prime order fields. *Advances in Mathematics of Communications*, 16(2):303, 2022.

[20] National Institute of Standards and Technology (NIST). *FIPS PUB 197: Advanced Encryption Standard (AES)*, 2001.

[21] Kaisa Nyberg. Differentially uniform mappings for cryptography. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 55–64. Springer, 1993.

[22] Kaisa Nyberg. Linear approximation of block ciphers. In Alfredo De Santis, editor, *Advances in Cryptology — EUROCRYPT'94*, pages 439–444, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[23] Kaisa Nyberg and Lars R Knudsen. Provable security against a differential attack. *Journal of Cryptology*, 8(1):27–37, 1995.

[24] Ronald L Rivest, Matthew J B Robshaw, Raymond Sidney, and Yiqun Lisa Yin. The RC6 block cipher. In *First advanced encryption standard (AES) conference*, 1998.

Cryptology 2026

10th International Cryptology and
Information Security Conference 2026

Research. Innovation. Security.