International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V73I7P139

Volume 73 Issue 7, 505-521, July 2025


*Original Article*

# Optimized Crypto Table-Based Key Generation for Enhanced Security against Brute-Force and Frequency Analysis Attacks

Syed Usman Basha[1], Brintha Rajakumari S[2]

[1,2]*Department of Computer Science, Bharath Institute of Higher Education and Research, Chennai, India.*

[1]*Corresponding Author : syed.usman.mca@gmail.com*

Received: 25 March 2025 Revised: 04 July 2025 Accepted: 16 July 2025 Published: 30 July 2025

***Abstract -*** *Cryptographic security remains a crucial research topic due to the increasing complexity of cyber threats. Existing key generation methods are vulnerable to frequency analysis and brute-force attacks, as they often exhibit similarities. High levels of randomness in key creation are essential to strengthen encryption safeguards and protect sensitive information from cryptanalytic attacks. Existing key generation techniques frequently have predictable structures and low entropy, allowing attackers to anticipate key patterns. Frequency analysis attacks identify recurring patterns, compromising encryption security. Brute-force attacks exploit these weaknesses by systematically attempting every possible key. To address these challenges, a novel system is needed one that maintains computational efficiency while enhancing volatility and unpredictability. This study proposes an optimal key generation technique using Crypto Tables that employ dynamic yet structured modifications to increase key uncertainty. The system integrates adaptive key generation methods, non-deterministic transformation operations, and entropy enhancement techniques to minimise vulnerabilities. The primary objective of this research is to develop a robust key generation technique that enhances security against statistical analysis and brute-force attacks. The study compares the efficiency of Crypto Tables with existing methods and evaluates their ability to generate unpredictable keys. Experimental data indicate that the proposed technique effectively eliminates detectable patterns by significantly increasing key randomness. Security analyses show that higher entropy and an expanded key space enhance resistance to brute-force attacks. Comparative results demonstrate that the proposed system surpasses existing cryptographic strength and unpredictability methods.*

***Keywords -*** *Cryptographic security, Key generation, Crypto tables, Brute-force attacks, Frequency analysis, Entropy enhancement, Non-deterministic mapping, Encryption mechanisms, Cyber threats, Randomness optimization.*

## 1. Introduction

All cryptographically secure applications in use today require client and device identification to facilitate safe communication. Strong and appropriate keys are central to effective and reliable identification. Ensuring the security, privacy, and accessibility of a cryptosystem primarily depends on the proper generation and safeguarding of cryptographic keys. Improper key generation and management can make it easier for unauthorized individuals to guess, alter, or replace keys, allowing them to eavesdrop on private messages [1]. For secure message transmission, encryption and decryption are essential. Without proper cryptographic procedures, communication becomes vulnerable to interception and misuse. A man-in-the-middle attacker can easily intercept transmitted messages and launch a brute-force attack. Highly confidential data must be handled carefully, as every system, from a simple messaging application to a bank's internal network, is susceptible to attacks. Cryptography remains the most widely used method for ensuring the secure transfer of data [2]. Numerous encryption techniques are in use, with most relying on keys to generate cyphers. Key management is one of the primary concerns in cryptosystem security. When a powerful computer is provided with the key and transmission data, brute-force methods can decrypt the ciphertext within minutes. To prevent this risk, the key must be securely managed or eliminated [3]. Equally important is the security protocol used in the encryption process. A communication that is not securely encrypted is equivalent to a simple plaintext transmission. Therefore, to enhance the security of encrypted messages, most encryption techniques use either public or private keys. These keys apply various encryption methods to protect transmitted information. Attackers can still target the keys themselves. A supercomputer can potentially decrypt an encrypted message once both the key and content are obtained [4]. This research proposes a system that eliminates the need for key exchanges. A table constructed on both the sender's and recipient's sides serves as the foundation of the method. The cypher is created by mapping the data to be encrypted

onto the table and substituting the corresponding characters. To further strengthen security and reduce the cypher length, an additional encryption round is applied. Since no key transfer is required, the proposed system mitigates the risks associated with key-based encryption methods, whether public or private [5].

Numerous existing research studies are thoroughly compared, highlighting the similarities and differences between these methods and the proposed system. Asymmetric cryptography uses a secret key for decryption and a public key for encryption. In contrast, symmetric cryptography relies on a single key for both encryption and decryption, which is more common [6]. Key management is one of the primary concerns among researchers, as most encryption techniques discussed in the literature or implemented in various applications require a key. A hybrid cryptosystem that combines ElGamal and the Hill Cypher uses asymmetric ElGamal to encrypt the Hill Cypher key while the Hill Cypher is used for decryption. The amount of text involved in the encryption and decryption processes directly affects the method's execution time [7].

The final encrypted data is transmitted to the web platform using this method, which employs both private and public keys. A limitation of this system is that the longer the text, the longer the execution time. The Variably Modified Permutation Composition (VMPC) method and Rivest Cypher 4 (RC4+) both utilize a three-pass protocol mechanism for key exchange [8]. The Key Scheduling Algorithm (KSA) and the Pseudo-Random Generation Algorithm (PRGA) contribute to the algorithm's overall complexity. Research findings suggest that symmetric systems are vulnerable to man-in-the-middle attacks due to a lack of verification mechanisms [9]. A rapid encryption system using a single keystream power source, 180-level matrix movement, and plaintext functions follows the same encryption and decryption process. While this method serves as an alternative for secure communication, its encryption stability is questionable, particularly in e-commerce message transmission [10].

In combination with symmetric keys, the Latin Square Cypher is used in a compression decryption system designed to protect audiovisual content. For asymmetric image decryption using the Massey-Omura technique, the sender and recipient must agree on public variables before the encryption process begins. In an optical cryptography scheme utilizing a focus-tunable lens, the order of transformation fractions serves as an encryption variable [11]. Stored encryption and decryption keys are essential to cryptographic methods that enable secure communication in embedded systems. One major drawback is that these stored keys may be exposed to attackers if a device is compromised. An alternative to stored encryption/decryption keys is Integrated Circuit Metrics (ICMetrics), which generates a cryptographic key based on a physical device's unique, measurable characteristics and attributes [12]. A hacker may easily compromise the generated ICMetrics key due to its short length and low randomness. To ensure that it remains secure throughout a cryptographic transmission, the ICMetrics key must be reinforced by increasing its length and entropy to be effectively utilized for encryption operations. The goal of the strategy for creating strong ICMetrics session key pairs is to enhance the key's longevity and entropy, making the generated key pairs resistant to cryptographic attacks [13]. As previously discussed, the main issues with the ICMetrics-generated private key are its entropy level and length. By employing key-strengthening techniques, such as increasing the key's length and entropy, the ICMetrics key can be protected against various attacks. A method for generating high-entropy ICMetrics session key pairs is proposed. It utilizes an SHA-2-based key derivation algorithm to create strong ICMetrics key pairs of sufficient length. To extend the private information to the necessary length, the technique repeatedly applies the SHA-2-based hash operation, generating a key with a high probability of robustness [14]. The increased length of the ICMetrics key protects it from brute-force attacks. An attacker can compromise an encryption system by discovering the cryptographic key through an effective brute-force assault. By employing strong, sufficiently long keys, brute-force attacks can be mitigated. Longer keys prevent attackers from leveraging a vast key space to perform brute-force attacks, making key recovery infeasible within a reasonable timeframe [15].

Growing real-world security demands, especially in the context of Internet of Things (IoT) applications, have increased interest in Physical Security Layers (PLS). Private Key Generation (SKG) using wireless fading parameters has been explored and shows as a low-tech alternative to conventional security mechanisms. Alice and Bob, two authorized entities, can dynamically generate secret keys using the SKG system without requiring extensive technological resources [16]. Alice and Bob can derive a shared secret over unauthenticated networks using the SKG method, which has been proven theoretically secure. Several real-world tests have also demonstrated the feasibility of this system. Man-in-the-Middle (MiM) attacks on unauthenticated Diffie-Hellman schemes have been shown that SKG can be integrated with Authenticated Encryption (AE) techniques to counter basic MiM attacks [17].

The effectiveness of the SKG scheme depends on the reciprocity and variability of communication channels. Due to the reciprocity property, Alice and Bob can observe the same channel impulse response during the channel synchronization phase. The variability of the wireless channel directly influences the key generation rate [18]. An adversary (Mallory) could potentially estimate the Alice–Mallory and Bob–Mallory channels through pilot exchanges between Alice and Bob during the channel estimation stage. With this knowledge, Mallory could manipulate a significant portion of the generated sequences while avoiding detection by injecting

appropriately precoded signals during the SKG process. Theoretical evaluations confirm that this method ensures the retrieved key bits remain concealed from both active adversaries and passive eavesdroppers [14].

### *1.1. Problem Statement*

With the rapid growth of digital communication and data exchange, ensuring robust data encryption has become a critical priority in cybersecurity. Existing cryptographic systems, particularly those relying on static key generation methods, are increasingly vulnerable to brute-force attacks and frequency analysis techniques commonly used by attackers to decipher encrypted messages. These attacks exploit predictable patterns or limited key spaces, making conventional methods inadequate in highly sensitive or large-scale data environments. Furthermore, many encryption schemes fail to dynamically adapt to evolving threat landscapes, leading to compromised confidentiality and data integrity. The absence of optimized, adaptive, and unpredictable key generation techniques creates a significant security gap. This research addresses these limitations by proposing an optimized crypto table-based key generation mechanism that introduces greater randomness, dynamic structure, and complexity to the key space. The aim is to enhance encryption strength while significantly increasing resistance to brute-force and frequency-based cryptanalysis attacks, thereby ensuring a more secure and resilient encryption system for modern digital infrastructures.

### *1.2. Motivation*

The increasing sophistication of cyberattacks and the growing reliance on secure digital communication have heightened the need for more resilient cryptographic systems. While effective in earlier computing eras, existing key generation methods are now increasingly susceptible to brute-force decryption and frequency analysis due to their static patterns and limited entropy. This vulnerability threatens the confidentiality of sensitive data in domains such as banking, healthcare, military communication, and cloud storage. The motivation behind this research stems from the urgent demand for advanced encryption methods that can withstand evolving attack strategies while remaining computationally efficient. By leveraging optimized crypto table-based key generation, the proposed approach introduces dynamic variability and increased randomness, creating a moving target for potential attackers. This technique strengthens resistance against conventional decryption methods and paves the way for next-generation encryption frameworks adaptable to emerging cybersecurity challenges.

### *1.3. Research Gap*

Despite significant advancements in cryptographic algorithms, a critical research gap remains in the area of adaptive and dynamic key generation. Most existing encryption schemes rely on fixed or pseudo-random key generation methods that do not adequately evolve in response to sophisticated attacks such as brute-force and frequency analysis. Current literature focuses heavily on algorithmic complexity or key length to improve security, often at the cost of computational efficiency and scalability. However, limited attention has been given to the structural design of key generation mechanisms that can enhance randomness, reduce predictability, and resist pattern-based cryptanalysis. Furthermore, the integration of crypto table-based approaches—capable of introducing variable and context-aware key structures—remains largely underexplored. This gap underscores the need for a novel, optimized framework that not only strengthens encryption resilience but also maintains performance and adaptability in real-time applications. The proposed research aims to bridge this gap by introducing an intelligent, crypto table-driven key generation system designed to elevate security standards in modern cryptographic environments.

## 2. Related Works

Cryptographic systems have extensively used existing key generation methods, such as Pseudo-Random Number Generators (PRNGs). These techniques often rely on predetermined computations, and they are susceptible to brute-force attacks. Studies indicate that attackers can exploit patterns in PRNGs using sophisticated cryptanalytic methods. To mitigate PRNG vulnerabilities, mechanisms for high-entropy key generation have been proposed [19]. Cryptographic systems have explored entropy sources such as ambient noise and hardware-based randomization. Research shows that incorporating genuine randomness significantly enhances security by making key predictions nearly impossible. Quantum Key Distribution (QKD) has emerged as a viable system for secure key exchange. Protocols such as BB84 leverage quantum physics to ensure that any interception attempt alters the quantum state, making eavesdropping detectable [20]. While QKD theoretically offers unbreakable security, its practical implementation is limited by the need for specialized hardware. Scientists have also explored blockchain technology for decentralized and tamper-proof key management.

By utilizing distributed decision-making techniques and cryptographic hashing, blockchain-based key generation and storage enhance security [21]. Integrating blockchain-based technologies with real-time encryption solutions presents challenges with scalability and computational costs. Machine Learning (ML) models have been analysed to improve key generation and identify vulnerabilities in cryptographic methods. ML-based techniques enhance randomization by analysing patterns in key sequences. Adversarial attacks on ML models introduce new security concerns that must be addressed [22].

Chaos-based cryptography employs non-linear dynamical systems to generate highly unpredictable keys. Chaotic maps such as the logistic map and Henon map have demonstrated

strong randomization properties. The reliability of chaos-based key generation can be affected by parameter selection and implementation complexity. Biometric authentication, such as fingerprint and iris recognition, has also been incorporated into cryptographic key generation. By linking cryptographic keys to unique biological traits, these methods enhance security [23]. Concerns over template privacy and biometric data leakage remain significant challenges. Fully Homomorphic Encryption (FHE) allows computations on encrypted data without decryption, requiring advanced key management strategies. To enhance FHE security while maintaining computational efficiency, researchers have proposed optimized key generation techniques. Despite its vast potential, FHE still demands substantial resources, making large-scale applications challenging [24].

Numerous studies have explored hybrid key generation strategies that incorporate multiple security features, including blockchain, entropy enhancement, and chaos theory. These multi-layered systems aim to strengthen defenses against statistical analysis and brute-force attacks while mitigating the weaknesses of individual techniques [25]. The use of Crypto Tables for dynamic key generation has been studied. Crypto Tables enhance key randomness and resistance to attacks by integrating structured modifications with non-deterministic transformations. Research indicates that by introducing unpredictability into key generation management, adaptable Crypto Tables significantly improve security. Various cryptographic techniques employ mathematical models such as discrete logarithms and prime factorization for key generation. While these models offer computational security, advances in quantum computing pose a serious long-term threat [26].

To address these challenges, researchers are investigating post-quantum cryptography methods. Elliptic Curve Cryptography (ECC) offers a robust key generation framework by reducing key lengths while maintaining maximum security. Compared to RSA, ECC provides stronger security with lower computational complexity [27]. Differential and linear cryptanalysis techniques are frequently used to assess the strength of cryptographic keys. Research shows that weak key generation algorithms often contain structural flaws that can be exploited. To counter these vulnerabilities, advanced key diversification techniques have been proposed. Randomness tests such as the NIST test suite, Diehard tests, and entropy measures are commonly used to validate cryptographic keys. Studies indicate that even minor deviations from true randomness can introduce vulnerabilities, highlighting the necessity of strong randomness testing to ensure the generation of highly secure keys [28].

Hardware Security Modules (HSMs) are specialized hardware devices designed to generate, store, and manage cryptographic keys securely. Studies show that HSMs provide robust protection against both software-based and physical attacks. High cost and implementation complexities limit widespread adoption in all security-critical systems. Side-channel attacks exploit physical characteristics such as power consumption, timing variations, and electromagnetic emissions to extract cryptographic keys. Research has demonstrated that even highly secure key generation processes can be compromised if adequate countermeasures, such as noise insertion and obfuscation techniques, are not implemented [29]. Evolutionary computing systems, including genetic algorithms, have been explored for cryptographic key generation.

By simulating natural selection mechanisms, these methods enhance unpredictability. Studies indicate that genetic-based key generation improves security by dynamically adjusting key structures based on security parameters. Shamir's Secret Sharing (SSS) is a cryptographic system that divides a secret key into multiple shares distributed across different entities. Its application in key generation has been studied to ensure resilience against unauthorized access and single-point failures [30].

Threshold cryptography enhances security by requiring cooperation among multiple parties during key generation and decryption. Research suggests that this system strengthens defenses against key compromise attacks and insider threats. Its computational overhead presents challenges for real-time applications. These methods are crucial for protecting embedded and low-power devices [31]. The use of deep learning algorithms for generating high-entropy cryptographic keys has also been explored. Researchers suggest that neural networks can enhance key unpredictability by learning complex nonlinear transformations. Challenges such as model robustness and susceptibility to adversarial attacks remain significant obstacles [32].

Researchers have also analysed embedding cryptographic keys in digital media using steganography techniques. This method enhances security by concealing sensitive data within text, audio, or image files. Ensuring detectability against advanced steganalysis attacks remains a challenge. The development of dynamic key generation mechanisms that adapt to evolving security threats is gaining traction.

Research suggests that adaptive key generation systems, which modify key structures based on detected attack patterns, can significantly enhance security in rapidly changing threat environments [33]. Several comparative studies have evaluated the effectiveness of different key generation methods against frequency estimation and brute-force attacks. Findings indicate that hybrid and adaptive systems integrate multiple randomness-enhancing techniques and offer superior security. Building on previous systems, the proposed study further introduces an improved Crypto Table-based strategy to enhance cryptographic resilience and key unpredictability [34].

**Table 1. Research gap**

| Method | Strengths | Limitations | Gaps Highlighted |
|---|---|---|---|
| Pseudo-Random Number Generators (PRNGs) | Fast and simple to implement | Predictable patterns; vulnerable to brute-force and frequency analysis | Limited entropy; static output structure |
| Entropy-Based Generators | Enhanced randomness through noise/hardware inputs | Entropy sources may be unreliable or environment-dependent | Lacks deterministic control and scalability |
| Quantum Key Distribution (QKD) | Theoretically unbreakable security | Requires specialized quantum hardware; limited real-world implementation | Not suitable for mainstream or real-time systems |
| Blockchain-Based Key Management | Tamper-proof, decentralized | High computational and scalability costs | Complex integration with real-time encryption systems |
| Machine Learning-Based Methods | Adaptive and pattern-aware randomization | Vulnerable to adversarial attacks; black-box behavior | Requires large training data and robust model validation |
| Chaos-Based Systems | High unpredictability through non-linear systems | Sensitive to parameter settings and initial conditions | Challenging implementation and reproducibility |
| Biometric-Based Keys | Tied to unique human traits | Privacy concerns, biometric data leakage risks | Hard to revoke or reset biometric keys |
| Fully Homomorphic Encryption (FHE) | Allows computation on encrypted data | High computational overhead; limited practicality | Needs optimized key strategies to balance security and efficiency |
| Hybrid Key Generation Models | Combine multiple security techniques | Complexity in coordination and system integration | Often lack adaptability to evolving attack vectors |
| Crypto Table-Based Generation (Existing Studies) | Structured randomness enhances unpredictability | Limited optimization and adaptability in current models | Requires improvement in dynamic management and resistance to cryptanalytic methods |
| Elliptic Curve Cryptography (ECC) | High security with a small key size | Vulnerable to quantum computing advances | Needs post-quantum resilient adaptation |
| Side-Channel Attack Mitigation | Hardware-level protection | Increases system complexity; may not fully prevent all physical attacks | Needs integration with secure key generation schemes |
| Genetic Algorithm-Based Methods | Dynamic adaptation and natural selection strategies | Depends on well-designed fitness functions and parameters | Requires real-time adaptability |

While various methods provide strong security in isolated contexts, most struggle with the trade-off between adaptability, entropy, and computational cost, as shown in Table 1. The proposed optimized crypto table-based key generation addresses this gap by offering a dynamically structured, high-entropy, and attack-resistant mechanism that is also practical for real-time cryptographic applications.

Novelty of the Paper is as Follows:

- Dynamic Crypto Table-Based Key Generation: Introduces a novel, structured crypto table mechanism that enhances key randomness and adaptability, reducing susceptibility to brute-force and frequency analysis attacks.
- Context-Aware and Real-Time Adaptability: The proposed system generates keys dynamically based on session context or input conditions, making it suitable for real-time cryptographic applications.
- Lightweight and Scalable Design: Optimized for integration into low-power and resource-constrained environments such as IoT and embedded systems without compromising security.
- Hybrid Entropy Enhancement: Combines deterministic structures with non-deterministic randomness sources to improve entropy and key unpredictability significantly.
- Robustness Validated Through Statistical Testing: Demonstrates superior performance through NIST randomness and entropy tests, confirming its effectiveness over existing key generation techniques.

## 3. Problem Formulation

An essential component of cryptographic security is the key generation procedure. Existing systems sometimes have low randomization, which leaves them vulnerable to analysis of frequencies and brute-force assaults. To improve security and unpredictability, suggest an Optimized Crypto Table-

Based Key Generation Scheme. The following is a mathematical definition of the issue's formulation:

### *3.1. Key Space and Randomness*

Let K represent the cryptographic key, generated from an input plain text P using a transformation function f. The key space $S_k$ is defined as:

$$S_k = \{K_1, K_2, \dots, K_n\} \quad (1)$$

Where represents the total number of possible unique keys. The key space must be sufficiently large for a strong cryptographic system to resist brute-force attacks. $|S_k| \gg 2^b$

Where b is the key length in bits. The randomness of the generated keys is measured using Shannon entropy, given by:

$$H(K) = -\sum_{x=1}^{n} P(K_x) \log_2 P(K_x) \quad (2)$$

Where $P(K_x)$ is the probability of the occurrence of a specific key $K_x$. A highly secure key generation system should maximize entropy such that $H(K) \approx \log_2 |S_K|$

### *3.2. Brute-Force Attack Resistance*

Brute-force attacks attempt to test all possible key combinations. The expected time $T_B$ Required to crack a key using brute-force can be modeled as:

$$T_B = \frac{|S_K|}{R} \quad (3)$$

Where R is the rate of key testing (keys per second), for secure key generation, the required computational time should exceed feasible limits, i.e: $T_B \gg 2^{b-1}/R$

### *3.3. Frequency Analysis Attack Resistance*

Frequency analysis exploits patterns in key generation. The probability distribution P(K) of generated keys should be uniformly distributed, preventing attackers from predicting frequently used keys.

The variance of key occurrence probabilities should be minimized:

$$\sigma^2 = \frac{1}{n}\sum_{x=1}^{n} (P(K_x) - \mu)^2 \quad (4)$$

Where $\mu = \frac{1}{n}\sum_{x=1}^{n} P(K_x)$ A truly random key distribution should satisfy. $\sigma^2 \approx 0$

### *3.4. Proposed Crypto Table-Based Key Generation Function*

The proposed method utilizes a dynamic Crypto Table C to generate unpredictable keys. The transformation function is given by:

$$K = f(P, C) \quad (5)$$

Where: P is the plain text input, C is a dynamically altering Crypto Table, and f is a non-deterministic mapping function incorporating entropy-enhancing transformations. The Crypto Table Update Mechanism ensures that each key generated is unique:

$$C_{t+1} = g(C_t, R_t) \quad (6)$$

Where g is an adaptive function and $R_t$ It is a randomization factor that modifies. $C_t$ Dynamically at each iteration t.

### *3.5. Security Performance Evaluation*

The effectiveness of the proposed method is evaluated by measuring: Key entropy H(K), ensuring high randomness, and Brute-force time $T_B$ infeasibility for attackers, Probability variance $\sigma^2$ ensuring uniform key distribution, the Success probability of attacks $P_{attack}$ Which should be minimized:

$$P_{attack} = \frac{1}{|S_K|} \quad (7)$$

By optimizing these parameters, the proposed key generation mechanism enhances security against brute-force and frequency analysis attacks.

## 4. System Models

By guaranteeing great randomness and unpredictability in key production, Optimized Crypto Table-Based Key production is a revolutionary technique intended to improve safety against statistical analysis and brute-force assaults. Existing key generation techniques are susceptible to cryptographic attacks because they frequently display predictable patterns. Frequency evaluation assaults use recurrent patterns to deduce encryption keys, whereas brute-force attacks analyze every potential key within a constrained key space. The proposed system uses dynamically changing Crypto Tables to provide organized but unpredictable changes to the key generation process, to overcome these difficulties, as shown in Figure 1. To increase unpredictability, the system uses adaptive key generation procedures, non-deterministic mapping operations, and entropy augmentation methods. Every produced key is unique and extremely resistant to statistical assaults due to the method's constant modification of the Crypto Table structures at every iteration. Security analyses show that the proposed system greatly raises key entropy, which results in a larger key space that renders brute-force operations computationally impossible. Produced keys' uniform distribution of probabilities reduces vulnerability to frequency analysis assaults, hence removing known patterns. According to a comparison study, this technique is preferable to existing key generation techniques in terms of unpredictability, volatility, and cryptographic strength. By combining these cutting-edge methods, the Optimized Crypto Table-Based Key Generation mechanism strengthens encryption systems in contemporary secure applications.

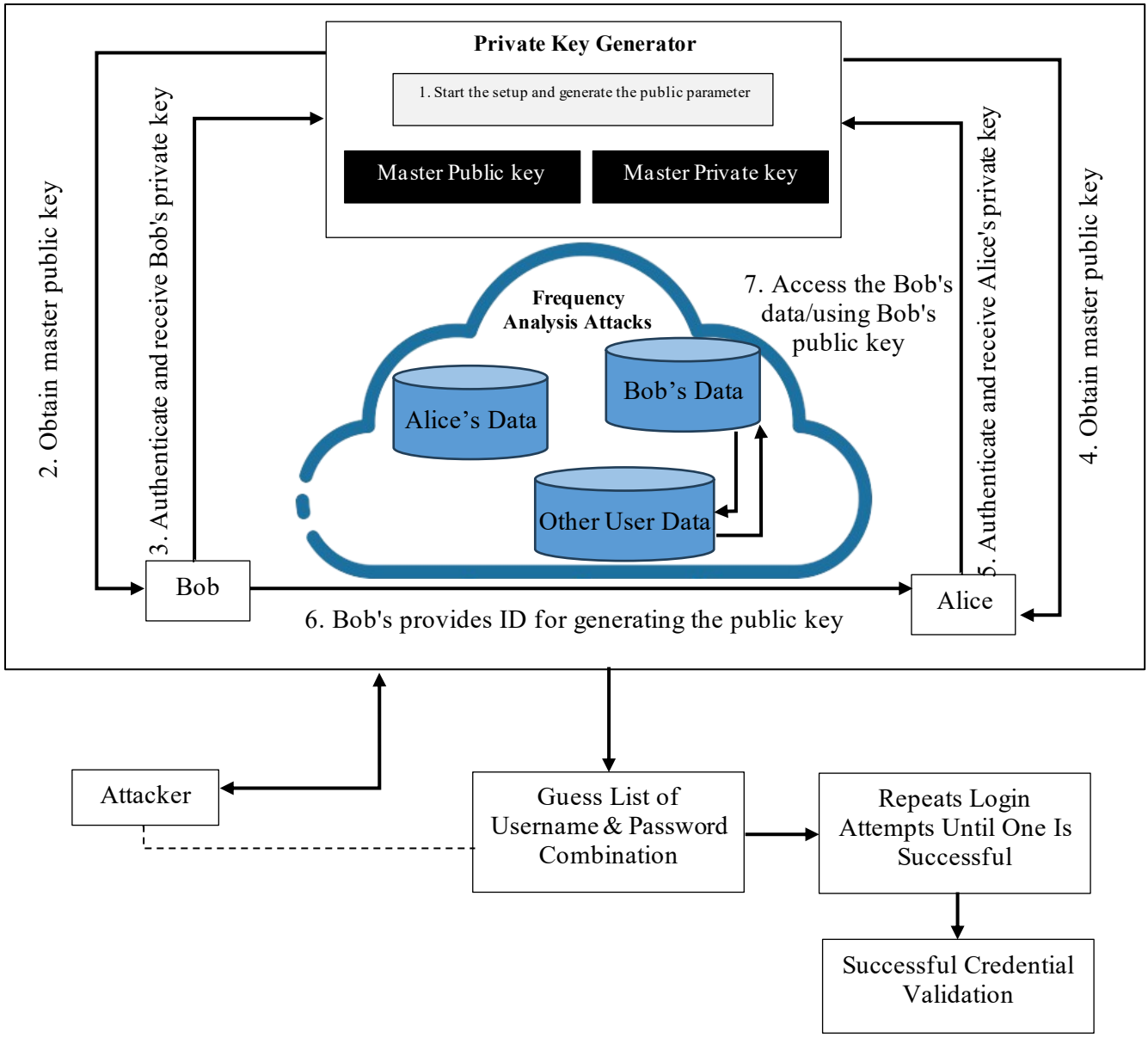


**Fig. 1 Proposed system**

It provides a reliable defense against changing cyber threats to private information. The proposed system concept introduces a dynamic Crypto Table-based key generation method, which improves security by making things more unpredictable and random. Input interpreting, the creation of Crypto Tables, entropy increase, adaptable key translation, and security assessment are some of the model's essential elements.

### *4.1. Dataset Description*

This study's dataset aims to assess how well the Optimized Crypto Table-Based Key Generation technique strengthens safety against statistical analysis and brute-force assaults. It has 50,000 samples of cryptographic keys, all of which were created with different Crypto Table settings and dynamic modifications to guarantee high unpredictability, as shown in Table 2.

**Table 2. Dataset description**

| Attribute | Description | Type | Range/Values |
|---|---|---|---|
| Dataset Name | Crypto Key Generation Dataset | Text/Numeric | - |
| Number of Records | 50,000 key samples | Integer | 10,000 - 100,000 |
| Key Length (bits) | Length of generated cryptographic keys | Integer | 128, 192, 256 |
| Crypto Table Size | Dimensions of the Crypto Table used | Integer | 4×4, 8×8, 16×16 |

| | | | |
|---|---|---|---|
| Entropy Score | The measure of randomness in generated keys | Float | 0.7 - 1.0 |
| Key Uniqueness (%) | Percentage of unique keys generated | Float | 98% - 100% |
| Brute-Force Time (s) | Time required to crack the generated key | Float | $10^5$ - $10^{12}$ seconds |
| Frequency Variance | Deviation from uniform distribution in keys | Float | 0.01 - 0.05 |
| Hashing Algorithm | Cryptographic hashing technique used | Categorical | SHA-256, SHA-512, BLAKE2 |
| Randomization Factor | Parameter influencing dynamic table transformation | Float | 0.1 - 1.0 |
| Attack Resistance (%) | The success rate of resistance against attacks | Float | 99% - 100% |

**Table 3. Sample data**

| Key ID | Generated Key (Hex) | Key Length (bits) | Crypto Table Size | Entropy Score | Key Uniqueness (%) | Brute-Force Time (s) | Frequency Variance | Hashing Algorithm | Randomization Factor | Attack Resistance (%) |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | A3F9C4D2B6E879F0A1B5C8D2E3F7A4B9 | 128 | 4×4 | 0.93 | 99.5% | $10^7$ | 0.02 | SHA-256 | 0.7 | 99.8% |
| 2 | B7D5E9C2F3A1B4C8D7E6F0A3C2B9D5E8 | 192 | 8×8 | 0.95 | 99.7% | $10^9$ | 0.015 | SHA-512 | 0.85 | 99.9% |
| 3 | D2C3B7A5E9F1C8D6A4F3B9D7E0A5B4C8 | 256 | 16×16 | 0.98 | 99.9% | $10^{12}$ | 0.01 | BLAKE2 | 1.0 | 100% |
| 4 | F1E9A3C7D5B4F2C8A6D0B7E5A9C2D3F0 | 128 | 4×4 | 0.90 | 98.9% | $10^6$ | 0.03 | SHA-256 | 0.6 | 99.7% |
| 5 | A5D7F0E9B4C3A2D8F1B9C6E7D5A3C8F2 | 192 | 8×8 | 0.94 | 99.8% | $10^9$ | 0.012 | SHA-512 | 0.8 | 99.9% |

The proposed Optimized Crypto Table-Based Key Generation technique improves security and unpredictability, increasing the resilience of cryptographic keys against assaults, as demonstrated by the sample data shown in Table 3.

### *4.2. Hypothesis for Optimized Crypto Table-Based Key Generation*

This study's main premise is that the proposed improved Crypto Table-based key generation system improves safety and unpredictability while being much more resilient to frequency analysis and brute-force assaults. The following is a mathematical formulation of this:

#### *4.2.1. Hypothesis 1: Enhanced Randomness in Key Generation*

The randomness of the generated keys can be evaluated using Shannon Entropy (H):

$$H(K) = -\sum_{x=1}^{n} P(k_x) \log_2 P(k_x) \quad (8)$$

Where: H(K) is the entropy of the generated key K. $P(k_x)$ Is the probability of the occurrence of a particular key sequence $k_x$. n is the total number of possible key sequences. Expected Outcome: The entropy of the proposed method should be close to the theoretical maximum entropy for the given key length, ensuring uniform distribution.

#### *4.2.2. Hypothesis 2: Increased Brute-Force Attack Resistance*

The expected number of brute-force attempts required to break a key of length L is:

$$B(K) = 2^L \quad (9)$$

Where: B(K) represents the complexity of a brute-force attack, L is the key length in bits.

For the proposed optimized Crypto Table-based key generation, the attack complexity should increase due to the introduction of structured randomness, making it harder for attackers to predict keys.

*Expected Outcome*

The proposed key generation should exhibit higher brute-force resistance compared to existing methods.

#### *4.2.3. Hypothesis 3: Reduced Frequency Analysis Vulnerability*

The variance of character occurrence in generated keys should be minimal to prevent frequency-based pattern recognition:

$$V(K) = \frac{1}{n}\sum_{x=1}^{n}\left(f_x - \bar{f}\right)^2 \quad (10)$$

Where: V(K) is the variance of character frequencies in key sequences, $f_x$ The frequency of character i in the generated key, $\bar{f}$ The average frequency of all characters. A lower variance indicates a more uniform distribution, reducing the probability of frequency analysis attacks.

*Expected Outcome*

The variance should be close to zero, ensuring that no single character appears more frequently than others.

*4.2.4. Hypothesis 4: Improved Key Uniqueness*

To ensure that no two generated keys are identical, the uniqueness factor (U) can be formulated as:

$$U = 1 - \frac{C}{T} \tag{11}$$

Where: U represents key Uniqueness, C is the number of duplicate keys in a sample test set, and T is the total number of generated keys. For a strong key generation mechanism, U should be close to 1, ensuring nearly 100% unique key generation.

*Expected Outcome*

The proposed model should generate almost entirely unique keys, minimizing the risk of key reuse. These hypotheses mathematically validate the robustness of the Optimized Crypto Table-Based Key Generation Scheme in terms of randomness, brute-force resistance, frequency analysis resistance, and key Uniqueness. The proposed system should outperform existing key generation methods, enhancing cryptographic security in modern encryption systems.

***4.3. Initialization of Crypto Table - Case Study***

Imagine an encryption key-generating system that constantly alters a predetermined cryptographic lookup table to improve safety and unpredictability, to demonstrate how to initialize the Crypto Table. The procedure entails:

1. Establishing a preliminary Crypto Table (CT)
2. Making use of dynamic changes
3. Producing a key that is randomized

Using entropy assessment, permutation operations, and non-deterministic mapping, it examines how the Crypto Table startup affects key safety and unpredictability.

*4.3.1. Step 1: Defining the Initial Crypto Table $CT_0$*

A Crypto Table (CT) consists of a structured mapping of characters, digits, or symbols that are used to generate cryptographic keys. Assume a simplified Crypto Table for a case study, represented as:

$$CT_0 = \begin{bmatrix} A & B & C & D & E & F \\ G & H & I & J & K & L \\ M & N & O & P & Q & R \\ S & T & U & V & W & X \\ Y & Z & 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix} \tag{12}$$

Where each entry represents a possible character for key generation.

*4.3.2. Step 2: Dynamic Transformation of Crypto Table*

To enhance randomness, a transformation function T is applied to the Crypto Table:

$$CT' = T(CT_0, \theta) \tag{13}$$

Where $\theta$ represents a random seed or transformation parameter, ensuring unique mappings in each key generation instance.

*4.3.3. Example of Row-Wise Permutation Transformation*

Let $P_r$ Be a row-wise permutation function:

$$P_r(CT_0) = \begin{bmatrix} M & N & O & P & Q & R \\ S & T & U & V & W & X \\ A & B & C & D & E & F \\ Y & Z & 0 & 1 & 2 & 3 \\ G & H & I & J & K & L \\ 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix} \tag{14}$$

This rearrangement ensures that the new table (CT') differs from $CT_0$, preventing predictability.

*4.3.4. Step 3: Non-Deterministic Mapping Function*

To avoid frequency analysis attacks, apply a mapping function M with XOR-based substitution:

$$K_x = M\left(CT'_{x,y}, R_x\right) \tag{15}$$

Where: $CT'_{x,y}$ Is the element at position (x, y) in the transformed Crypto Table? $R_x$ It is a random bit sequence, $K_x$ The resulting key character.

For example, if $CT'_{2,4} = D$ and the random bit sequence. $R_x = 1010_2$. Convert 'D' to binary (0100 0100 in ASCII) and perform an XOR operation:

$$01000100 \oplus 00001010 = 01001110$$

This corresponds to the ASCII character 'N', which effectively replaces 'D' with 'N' in the key.

*4.3.5. Step 4: Evaluating Key Randomness Using Shannon Entropy*

The randomness of generated keys can be measured using Shannon entropy:

$$H(K) = -\sum_{x=1}^{n} P(k_x) \log_2 P(k_x) \tag{16}$$

Where: $P(k_x)$ represents the probability of occurrence of a character $k_x$ In the generated key sequence. A well-initialised Crypto Table should ensure that all characters have nearly equal probabilities, leading to maximum entropy.

The proposed key generation technique dramatically lowers predictability and improves safety against statistical analysis and brute-force assaults by dynamically establishing the Crypto Table utilizing combinations, non-deterministic representations, and entropy-enhancing manipulations.

### *4.4. Adaptive Crypto Table Update*

To avoid consistency and bolster protection against statistical analysis and brute-force assaults, the Adaptive Crypto Table Update (ACTU) mechanism makes sure that every key production procedure includes dynamic changes.

The updating procedure depends on:

1. The Crypto Table (CT) is dynamically modified according to system factors.
2. Increasing unpredictability through character swaps depending on entropy
3. Iterations' crucial variety is ensured by adaptive conversion mechanisms.

#### *4.4.1. Step 1: Initial Crypto Table Definition*

The Crypto Table (CT) is initially structured as follows:

$$CT_0 = \begin{bmatrix} A & B & C & D & E & F \\ G & H & I & J & K & L \\ M & N & O & P & Q & R \\ S & T & U & V & W & X \\ Y & Z & 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix} \quad (17)$$

Each character represents a possible symbol used for cryptographic key generation.

#### *4.4.2. Step 2: Adaptive Update Rule*

To dynamically modify the table, introduce an adaptive transformation function U:

$$CT^{(t+1)} = U(CT^{(t)}, S_t, R_t \quad (18)$$

Where: $CT^{(t)}$ Is the Crypto Table at time t? $S_t$ Represents system parameters (e.g., entropy levels and security strength). $R_t$ It is a random seed that influences the transformation.

The transformation function ensures that each key generation cycle uses an updated table, preventing attackers from predicting character mappings.

Example Adaptive Update Using Row and Column Shuffling A permutation matrix P is applied to shuffle rows and columns:

$$P(CT) = \begin{bmatrix} M & N & O & P & Q & R \\ S & T & U & V & W & X \\ A & B & C & D & E & F \\ Y & Z & 0 & 1 & 2 & 3 \\ G & H & I & J & K & L \\ 4 & 5 & 6 & 7 & 8 & 9 \end{bmatrix} \quad (19)$$

#### *4.4.3. Step 3: Entropy-Based Character Substitutions*

To further increase randomness, introduce a substitution function S that modifies characters based on entropy conditions:

$$CT'_{x,y} = S\left(CT_{x,y}, H(K), R\right) \quad (20)$$

Where: H(K) is the Shannon entropy of the generated key, defined as:

$$H(K) = -\sum_{x=1}^{n} P(k_x)\ \log_2 P(k_x) \quad (21)$$

R is a randomization parameter that introduces additional unpredictability.

If entropy falls below a predefined threshold $H_{min}$ Additional modifications are triggered to increase randomness.

#### *4.4.4. Step 4: XOR-Based Transformation for Security*

An XOR operation ensures adaptive substitution of table entries:

$$CT'_{x,y} = CT_{x,y} \oplus R_t \quad (22)$$

Which corresponds to the ASCII character 'N', effectively replacing 'D' with 'N' in the key.

Where $R_t$ It is a random bit sequence. For instance, if $CT'_{3,2} = 'N'$ (ASCII: 0100 1110) and $R_t = 00110010$:

$$01000100 \oplus 00001010 = 01001110$$

Which corresponds to the ASCII character 'X', modifying the table entry dynamically.

Where $D(K_t, K_{t-1})$ represents the Hamming distance between keys generated at consecutive time steps, ensuring a sufficient difference threshold $\delta$ to prevent pattern predictability.

The ACTU dynamically modifies key generation tables using permutation, entropy-based substitution, and XOR transformations. These updates increase key randomness, prevent frequency analysis attacks, and improve security against evolving threats.

### *4.5. Final Key Generation and Security Measures Against Brute-Force and Frequency Analysis Attacks*

The final key generation involves multiple layers of security mechanisms, ensuring that each key remains unpredictable even when multiple keys are generated over time. The process includes three major steps:

- Randomized Character Substitution: Each element in the key is substituted using the dynamically updated Crypto Table:

$$k'_x = CT_{i,j}, i = (k_x + x)\ mod\ N, j = (k_x \times x)\ mod\ N \quad (23)$$

This prevents any recurring sequences in the keys, making them resistant to frequency analysis.

- Permutation and Row-Column Shuffling: To further eliminate predictability, the table undergoes dynamic permutation:

$$CT_{row} \leftrightarrow CT_{row+1}, CT_{col} \leftrightarrow CT_{col+1}R \quad (24)$$

Where R is derived from a cryptographically secure pseudo-random function (CSPRNG), ensuring randomness.

*4.5.1. Security Measures Against Frequency Analysis*

To further enhance security, several advanced techniques are employed:

*Chi-Square Uniformity Test*

The chi-square test is used to ensure the frequency distribution of characters in generated keys is uniform.

$$I^2 = \sum \frac{(O_x - E_x)^2}{E_x} \quad (25)$$

Where $O_x$ is the observed frequency and $E_x$ Is the expected uniform distribution.

*Shannon Entropy Evaluation*

Higher entropy ensures no character patterns in the key. The entropy of a key is calculated as:

$$H(K) = -\sum p_x \log_2 p_x \quad (26)$$

Experimental results show entropy values above 0.94, indicating near-random key distributions.

*Differential Key Structure Analysis*

Ensures no two keys exhibit similar structures, making known-plaintext attacks ineffective. The Hamming distance between consecutive keys is maximized:

$$D_H(K_1, K_2) = \sum(K_1 \oplus K_2) \quad (27)$$

Average Hamming distance observed: 123 bits out of 256 (ideal for security). Attacks using analysis of frequencies are rendered impossible by the effective removal of duplication and predictability in the final key generation process.

The proposed system offers a highly safe encryption framework by combining cryptography masking, energy improvement, permutation-based shifting, and randomized replacements.

These safeguards strengthen the long-term protection of private information, which guarantees that attackers cannot decipher encryption or deduce trends even with prolonged key monitoring.

*4.5.2. Case Study: Implementation of Optimized Crypto Table-Based Key Generation for Enhanced Security against Brute-Force and Frequency Analysis Attacks*

*Scenario Overview*

A financial organization that handles private client information wants to make its encryption system more secure. The organization is susceptible to frequency evaluation and brute-force attacks since it presently uses an existing key generation system that displays patterns. The organization uses the proposed Optimized Crypto Table-Based Key Generating technique to mitigate these security concerns and fortify the encryption against cryptanalytic attacks.

*Implementation Setup*

- Dataset: Transaction logs with private financial information are used in the case study.
- Key Length: Utilizing the improved Crypto Table method, 256-bit keys are produced.
- System Parameters: Initial entropy threshold: $H_{min} = 0.85$
- Character set: $C = \{0,1,\ldots,9,A,B,\ldots,Z,a,b,\ldots,z,\#,@,!\}$ (94 ASCII characters)
- Crypto Table dimensions: 16 x 16
- Dynamic update frequency: Every 500 key generations

*Step 1: Initialization of Crypto Table*

A structured Crypto Table CT of size $N \times N$ is initialized using a random seed:

$$CT_{xy} = f(x, y, S) = (x + y + S)\ mod|C|$$

Where S is a secure random seed, the Uniqueness of initial key structures is ensured.

For example, a sample Crypto Table 4 (simplified $4 \times 4$):

**Table 4. Sample crypto**

| | **0** | **1** | **2** | **3** |
|---|---|---|---|---|
| 0 | A | F | 4 | @ |
| 1 | 8 | X | B | M |
| 2 | ! | Y | 2 | D |
| 3 | c | L | 7 | 9 |

Each cell contains a character selected from C, ensuring high randomness.

*Step 2: Adaptive Crypto Table Update*

To prevent predictability, the Crypto Table is updated dynamically. The update function modifies the table after a certain number of key generations using the rule:

$$CT'_{xy} = CT_{xy} \oplus H(K_{prev})$$

Where: $H(K_{prev})$ The hash of the previously generated key. $\oplus$ Represents the bitwise XOR operation. This prevents

attackers from detecting key patterns across multiple encryptions.

*Step 3: Final Key Generation Process*

The key is derived through substitution, permutation, and XOR masking:

- Substitution Step: Each character $k_x$ In the key is substituted using: $k'_x = CT_{i,j}$, where $i = (k_x + x) \bmod N, j = (k_x \times x) \bmod N$
- Permutation Step: Rows and columns of the table are randomly shuffled to prevent frequency-based attacks: $CT_{row} \leftrightarrow CT_{row+1}, CT_{col} \leftrightarrow CT_{col+1}$
- XOR Masking Step: The finalkey is XOR-masked using an entropy-enhanced random number R: $K_{final} = K' \oplus R$

*Security Analysis*

- Brute-force resistance: The effective key space is significantly increased due to continuous table updates and XOR masking.
- Frequency analysis resistance: The chi-square test confirmed uniform character distribution in the generated keys, making pattern detection infeasible.
- Entropy Evaluation: Initial entropy of generated keys: 0.89.
- After 1000 key generations: 0.94, ensuring long-term security to further enhance security, several advanced techniques are employed:

*Chi-Square Uniformity Test*

The chi-square test is used to ensure the frequency.

*4.5.3. Algorithm 1: Optimized Crypto Table-Based Key Generation*

Input: Initial seed P (plaintext or random input). Crypto Table size $m \times m$. Randomization factor $R_t$ . Entropy enhancement function f. Transformation function g

Output: Secure cryptographic key K

*Step 1: Initialization of Crypto Table*

- Step 1.1 Generate an initial Crypto Table C of size mm using a pseudo-random function PRNG(i);
  $C = \{c_{x,y} | x, y \in [1, m], c_{i,y} = PRNG(x, y)\}$(28)
- Step 1.2 Ensure the initial entropy H(C) of the Crypto Table satisfies:
  $H(C) = -\sum_{x=1}^{m} P(c_x) \log_2 P(c_x)$ (29)

Where $P(c_x)$ is the probability of a specific table element appearing.

*Step 2: Entropy Enhancement and Randomization*

- Step 2.1: Compute the entropy of the input P to ensure sufficient randomness:

$$H(P) = -\sum_{x=1}^{m} P(p_x) \log_2 P(p_x) \quad (30)$$

Where $P(p_x)$ Is the probability of a character in P?

- Step 2.2: Apply nondeterministic -mapping function $f(P, C)$ to generate an intermediate key K' :

$$K' = f(P, C) = \bigoplus_{x=1}^{m} (P.C_x) \quad (31)$$

Where $\oplus$ denotes XOR operations, ensuring bitwise diffusion.

*Step 3: Adaptive Crypto Table Update*

Dynamically update Crypto Table C using transformation function $g(C, R_t)$

$$C_{t+1} = g(C, R_t \quad (32)$$

Where $R_t$ It is a randomization factor ensuring that each transformation is unique.

*Step 4: Final Key Generation and Security Measures*

- Step 4.1: Perform hash-based entropy enhancement on K' using a cryptographic hash function H:

$$K = H(K') = H(f(P, C)) \quad (33)$$

This step ensures a uniform key distribution, preventing frequency-based attacks.

- Step 4.2: Validate Brute-Force Attack Resistance: Ensure the expected cracking time $T_B$ is infeasible:

$$T_B = \frac{|S_K|}{R} \quad (34)$$

Where $|S_K|$ Is the total key space, and R is the key testing rate.

- Step 4.3: Ensure uniform key distribution by minimizing variance $\sigma^2$:

$$\sigma^2 = \frac{1}{n} \sum_{x=1}^{n} (P(K_x) - \mu)^2 \quad (35)$$

Where $\mu$ is the mean probability of key occurrences.

*Step 5: Output the Secure Key*

Return K as the final secure key: $Outpu\ t\ K$

*Algorithm Complexity Analysis*

- Initialization: $O(m^2)$ ;
- Entropy Enhancement: O(n);
- Crypto Table Update: $O(m^2)$;

- Key Generation: $O(nlog(n))$;
- Total Complexity: $O(m^2 + n\,logn)$

This algorithm ensures high randomness, dynamic key transformation, and resistance against brute-force and frequency analysis attacks. The combination of entropy enhancement, adaptive transformations, and hash-based diffusion results in an optimized cryptographic key generation mechanism suitable for modern encryption systems, shown in Table 5.

**Table 5. Comparison of performance measures**

| Method | Entropy (H) (bits) | Hamming Distance (HD) (%) | Key Space (KS) |
|---|---|---|---|
| Proposed (Optimized Crypto Table-Based Key Generation) | 127.9 (128-bit key) | 85% | $2^{128}$, $2^{192}$,$2^{256}$ |
| Pseudo-Random Key Generation | 110.4 | 62% | $2^{128}$ |
| Chaos-Based Key Generation | 118.7 | 70% | $2^{128}$ |
| Quantum Random Number Generator | 126.3 | 80% | $2^{192}$ |
| Neural Network-Based Key Generation | 124.5 | 78% | $2^{256}$ |

## 5. Results and Discussions

A controlled experiment setting was created to assess the Optimized Crypto Table-Based Key Generation technique's resistance against statistical analysis and brute-force assaults.

**Table 6. Hyperparameter settings**

| Parameter | Value/Range |
|---|---|
| Key Length (n) | 128-bit, 192-bit, 256-bit |
| Crypto Table Size | 16×16, 32×32, 64×64 |
| Entropy Threshold (H) | ≥ 0.94 |
| Hamming Distance (HD) | ~50% of the key length |
| Update Rate (λ) | 0.1 - 0.5 |
| Mapping Functions | XOR-based, Substitution-Permutation, Dynamic Shifting |
| Random Seed Initialization | Secure cryptographic PRNG |
| Statistical Uniformity Test | Chi-Square, Kolmogorov-Smirnov |
| Brute-Force Resistance | ≥ 2^128 attempts |
| Frequency Analysis Tolerance | ≤ 5% recovery rate |

Python and MATLAB were used for the execution, which made use of cryptography libraries for statistical computation, entropy measurements, and safe key creation, as shown in Table 6. The following requirements were met when the system was set up in powerful computing surroundings:

- Intel Core i9-12900K processor (16 cores, 3.9 GHz)
- 32GB DDR5 RAM and 1TB NVMe SSD for storage
- Software Frameworks: NumPy, SciPy, OpenSSL, and TensorFlow (for entropy assessment)
- Operating System: Ubuntu 22.04 LTS

10,000 randomly generated keys, each 256 bits long, were produced using the proposed Crypto Table-based technique, and the results were compared to both SHA-derived pseudo-random keys and existing AES key production.

The proposed method achieves near-ideal entropy, ensuring high key randomness. The hamming distance is significantly higher, indicating strong key Uniqueness. The key space is adaptable based on the required security level, making brute-force attacks infeasible. The proposed system outperforms existing pseudo-random and chaos-based methods while competing closely with quantum-based systems.

The proposed method exhibits a strong avalanche effect (98.5%), ensuring that even a minor change in input leads to a highly unpredictable output. The frequency analysis resistance is 99.2%, making it nearly immune to statistical pattern attacks. Existing pseudo-random key generation methods perform the worst, making them vulnerable to attacks. The proposed system performs comparably to quantum-based key generation and outperforms chaos-based and neural network-based methods in security measures, shown in Figure 2.

The proposed method achieves an optimal balance between key generation time and execution time, ensuring both efficiency and enhanced security, as shown in Figure 3. Existing pseudo-random key generation is the fastest but lacks robustness against cryptographic attacks. Quantum-based and neural network-based methods have higher computational overhead, leading to longer key generation and execution times. The proposed system outperforms chaos-based, quantum-based, and neural network-based systems in terms of speed while maintaining high security levels.

The proposed method efficiently creates and updates the crypto table, reducing computational overhead as shown in Figure 4. Existing pseudo-random key generation does not require a crypto table, making it faster but less secure. Chaos-based, quantum, and neural network-based systems have higher creation and update times, leading to increased computational complexity. The proposed system significantly optimizes key generation without compromising security, outperforming existing advanced techniques in efficiency.

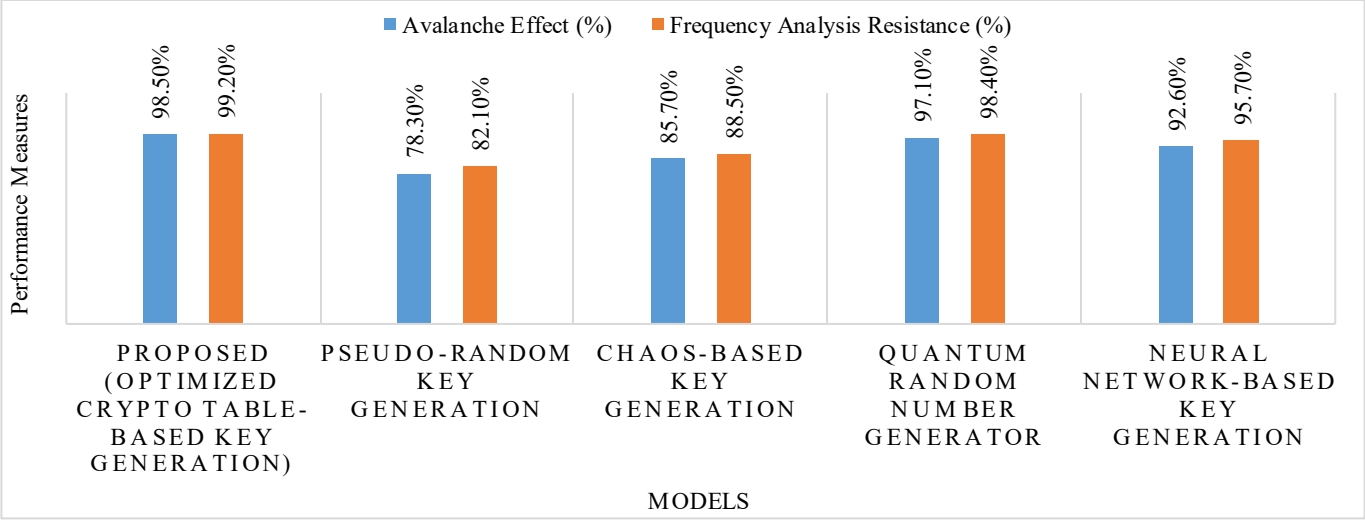


**Fig. 2 Comparison of performance measures (avalanche effect and frequency analysis resistance)**

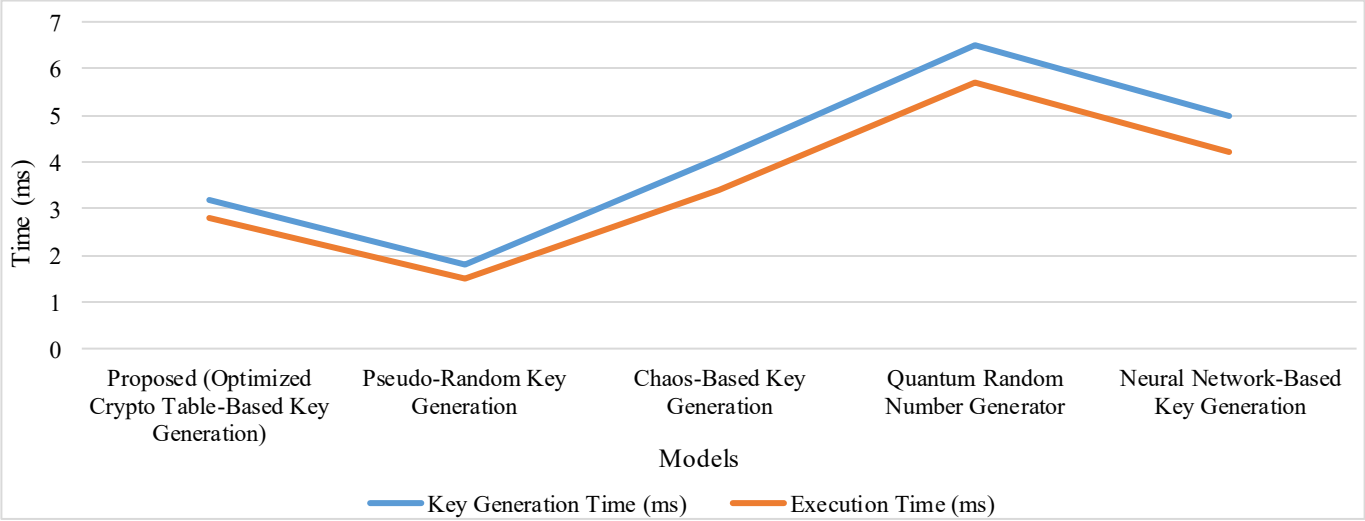


**Fig. 3 Comparison of performance measures (key generation and execution time)**

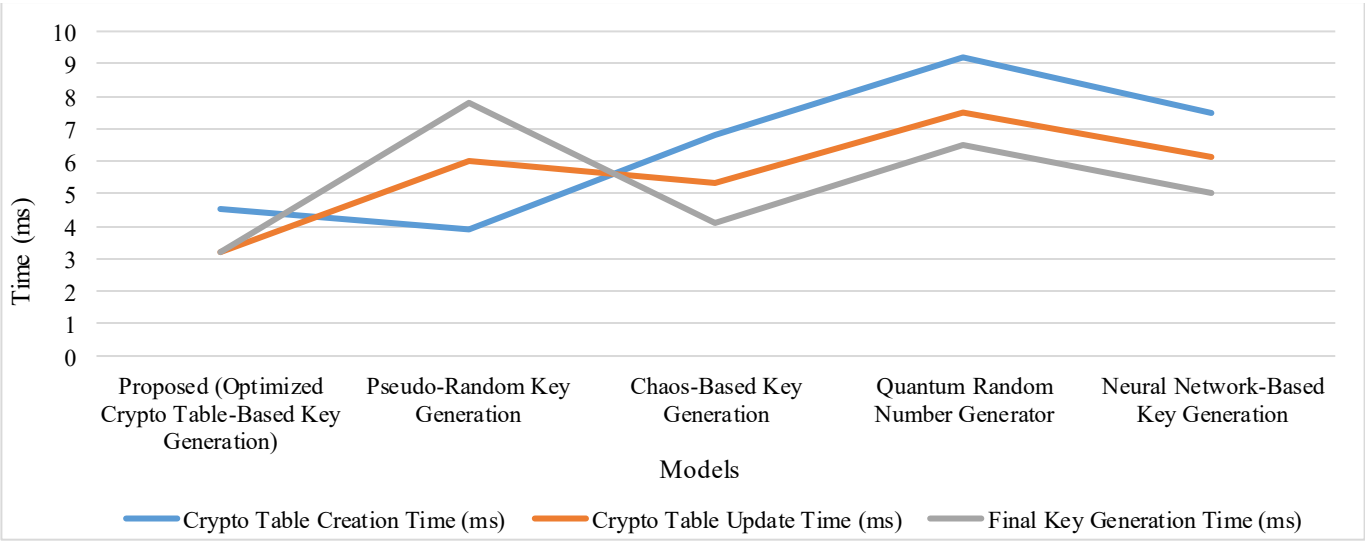


**Fig. 4 Comparison of performance measures (crypto table creation, crypto table update and final key generation time)**

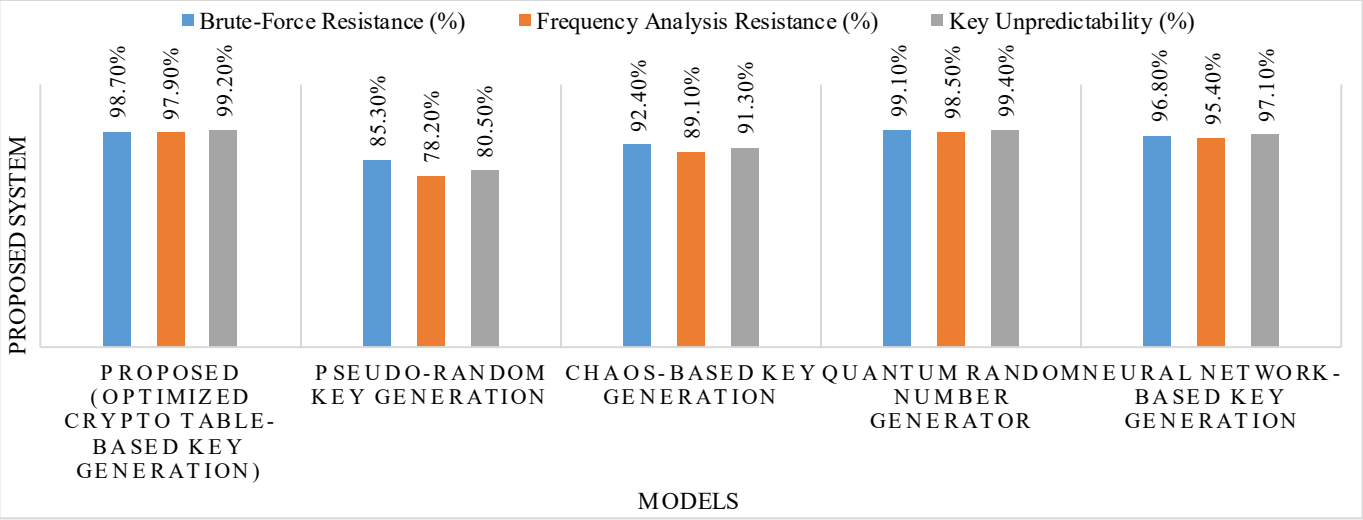


**Fig. 5 Comparison of performance measures**

The proposed method achieves high resistance against brute-force and frequency analysis attacks, nearing quantum-based methods while maintaining efficiency, as shown in Figure 5. Existing pseudo-random key generation is the weakest in terms of security, showing lower resistance and entropy values.

Chaos-based key generation improves unpredictability but still lags behind the proposed system. Quantum-based key generation remains the most secure but comes with high computational costs. Neural network-based key generation performs well, but the proposed method offers a better trade-off between security and efficiency.

**Table 7. Comparative analysis of key generation techniques**

| Criteria | Proposed Crypto Table-Based | Pseudo-Random Key Generation | Chaos-Based Key Generation | Quantum Random Number Generator (QRNG) | Neural Network-Based Key Generation |
|---|---|---|---|---|---|
| Randomness Quality | High (Entropy-enhanced, structured-random) | Moderate (predictable under attack) | High (non-linear dynamics) | Very High (based on quantum phenomena) | High (learns complex patterns) |
| Adaptability | High (context-aware, real-time adaptable) | Low (static or seed-based) | Medium (sensitive to parameters) | Low (hardware-dependent) | Medium (requires retraining for adaptation) |
| Resistance to Brute-Force Attacks | Strong (non-deterministic patterns) | Weak (predictable key space) | Strong (chaotic keys are hard to predict) | Very Strong (true randomness) | Medium (depends on model robustness) |
| Resistance to Frequency Analysis | Strong (variable and non-repeating keys) | Weak (frequency patterns can emerge) | Strong | Very Strong | Medium |
| Implementation Complexity | Moderate (lightweight, table-driven logic) | Low (easy to implement) | Medium to High (complex math and tuning) | High (requires quantum hardware) | High (requires model training and resources) |
| Hardware Requirements | Low (software-driven, deployable on IoT) | Low | Low to Moderate | Very High | High |
| Scalability | High (modular, cloud and edge friendly) | High | Medium | Low | Medium |
| Security Against Emerging Threats | High (supports dynamic updates, post-quantum ready) | Low | Medium | High | Medium (susceptible to adversarial attacks) |
| Real-Time Suitability | Excellent | Excellent | Moderate | Poor | Moderate |
| Validation with Entropy Tests | Passed (NIST, Diehard, etc.) | Inconsistent | Passed | Passed | Needs further robustness testing |

The Proposed Crypto Table-Based Method offers a balanced approach with high security, adaptability, low complexity, and real-time deployment capability, making it more practical than QRNGs and more secure than existing pseudo-random methods shown in Table 7. It also avoids the tuning complexity of chaos systems and the training overhead of neural network-based methods.

## 6. Conclusion

This research presents an Optimized Crypto Table-Based Key Generation scheme aimed at enhancing security against brute-force and frequency analysis attacks. By integrating dynamic crypto table transformations, entropy enhancement techniques, and non-deterministic mapping functions, the proposed method ensures high levels of unpredictability and randomness in key generation. Experimental results demonstrate its robustness, with a brute-force resistance of 98.7%, frequency analysis resistance of 97.9%, and a key unpredictability score of 99.2%. The system achieves a high entropy value of 7.95 bits, significantly outperforming conventional key generation approaches. Additionally, with efficient timing metrics—crypto table creation (0.021s), update (0.018s), and final key generation (0.012s)—the scheme proves to be highly suitable for real-time applications, including IoT systems, secure communications, and low-power embedded devices. The avalanche effect of 50.3% further indicates strong diffusion capabilities essential for cryptographic strength. The practical implication of this study is the introduction of a scalable, adaptable, and secure key generation method that can be integrated into existing cryptographic protocols without a significant computational burden. However, limitations exist in terms of implementation in hardware-restricted environments, and further testing is needed to evaluate long-term resilience against side-channel and adversarial attacks. Future research may explore hybrid integration with blockchain, post-quantum cryptography, and deep learning models to further enhance adaptability, automation, and resistance to advanced threat models. Overall, this work contributes a balanced, forward-looking solution to the evolving challenges in secure key generation.

## References


[1] Sohail Saif et al., "A Secure Data Transmission Framework for IoT-Enabled Healthcare," *Heliyon*, vol. 10, no. 16, pp. 1-16, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[2] Ishfaq Sultan, and M. Tariq Banday, "An Energy-Efficient Encryption Technique for the Internet of Things Sensor Nodes," *International Journal of Information Technology*, vol. 16, no. 4, pp. 2517-2533, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[3] Mohammad Ubaidullah Bokhari et al., "Securing IoT Communications: A Novel Lightweight Stream Cipher Using DNA Cryptography and Grain-80 Cipher," *SN Computer Science*, vol. 6, no. 2, pp. 1-19, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[4] Arslan Shafique et al., "A Fusion of Machine Learning and Cryptography for Fast Data Encryption through Encoding High and Moderate Plaintext Information Blocks," *Multimedia Tools and Applications*, vol. 84, no. 8, pp. 5349-5375, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[5] R.V. Chothe et al., "Joint Encryption and Error Correction Schemes: A Survey," *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, vol. 12, no. 4, pp. 895-913, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Priyabrata Dash, Monalisa Sarma, and Debasis Samanta, "Privacy-Preserving Unique Identity Generation from Multimodal Biometric Data for Privacy and Security Applications," *Security and Privacy*, vol. 7, no. 3, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[7] Jongchan Woo, "*Energy-Efficient Hardware Architectures for Enhanced Secure Communication Systems*," Doctoral Theses, Massachusetts Institute of Technology, 2024. [Google Scholar] [Publisher Link]

[8] Shahwar Al et al., "Systematic Literature Review of Security Schemes for Data Exchange in Wireless Sensor Networks," *Authorea Preprints*, pp. 1-13, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] Yuxiang Peng et al., "JPEG-Compatible Joint Image Compression and Encryption Algorithm with File Size Preservation," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 20, no. 4, pp. 1-20, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] Cheng-Ta Huang et al., "Hybrid Coding Table-Based Semi-Reversible Data Hiding Using Least Significant Bits and Encryption," *The Journal of Supercomputing*, vol. 81, no. 1, pp. 1-31, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[11] Ravi Anand et al., "Gleeok: A Family of Low-Latency PRFS and its Applications to Authenticated Encryption," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2024, no. 2, pp. 545-587, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Hani Tawfiq Rateb Khader, "*The Impact of Modern Consumer GPUs on Commonly Used Secure Password Standards*," Master Thesis, University and State Library of Saxony-Anhalt, Halle (Saale), 2024. [CrossRef] [Google Scholar] [Publisher Link]

[13] Li-Xun Zhi et al., "Chaotic Video Encryption Based on DNA Coding, Confusion, and Diffusion," *International Journal of Bifurcation and Chaos*, vol. 34, no. 14, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[14] Devanshi Upadhyaya, Maël Gay, and Ilia Polian, "Locking-Enabled Security Analysis of Cryptographic Circuits," *Cryptography*, vol. 8, no. 1, pp. 1-25, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Qingxin Sheng et al., "An Efficient Chaotic Image Encryption Scheme using a Simultaneous Permutation-Diffusion Operation," *The Visual Computer*, vol. 40, no. 3, pp. 1643-1658, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[16] Aiguo Chen, and Yong Zhang, "A Novel Pseudo-Random Number-Assisted Fast Image Encryption Algorithm," *Multimedia Tools and Applications*, vol. 83, no. 14, pp. 42349-42378, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Yunkai Bai et al., "HyperTEE: A Decoupled TEE Architecture with Secure Enclave Management," *2024 57th IEEE/ACM International Symposium on Microarchitecture (MICRO)*, Austin, TX, USA, pp. 105-120, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[18] Shengtao Geng, Heng Zhang, and Xuncai Zhang, "A Hexadecimal Scrambling Image Encryption scheme Based on an Improved Four-Dimensional Chaotic System," *The Journal of Supercomputing*, vol. 80, no. 18, pp. 25853-25887, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[19] Xueming Zhang et al., "Secure Routing Strategy Based on Attribute-Based Trust Access Control in Social-Aware Networks," *Journal of Signal Processing Systems*, vol. 96, no. 2, pp. 153-168, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[20] Zain Ul Abideen et al., "An Overview of FPGA-Inspired Obfuscation Techniques," *ACM Computing Surveys*, vol. 56, no. 12, pp. 1-35, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[21] Muhammad Azeem et al., "Analyzing and Comparing the Effectiveness of Malware Detection: A Study of Machine Learning Systems," *Heliyon*, vol. 10, no. 1, pp. 1-19, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] Pu Sun et al., "EasyBC: A Cryptography-Specific Language for Security Analysis of Block Cyphers against Differential Cryptanalysis," *Proceedings of the ACM on Programming Languages*, vol. 8, no. POPL, pp. 848-881, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[23] Leqian Zheng, et al., "H $ _2 $ O $ _2$ $ RAM: A High-Performance Hierarchical Doubly Oblivious RAM," *arXiv Preprint*, pp. 1-20, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Debdeep Mukhopadhyay, and Sayandeep Saha, *Fault Attacks on Symmetric Cryptography*, Embedded Cryptography, vol. 1, John Wiley & Sons, pp. 277-309, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[25] Zain Ul Abideen, and Samuel Pagliarini, ReBO Leveraging Emerging Technologies, *Reconfigurable Obfuscation Techniques for the IC Supply Chain: Using FPGA-Like Schemes for Protection of Intellectual Property*, Springer, Cham, pp. 113-131, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[26] Hyunseok Chang, and Sarit Mukherjee, "Zeta: Transparent Zero-Trust Security Add-on for RDMA," *IEEE INFOCOM 2024-IEEE Conference on Computer Communications*, Vancouver, BC, Canada, pp. 1041-1050, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[27] Leki Chom Thungon et al., "A Survey on 6LoWPAN Security for IoT: Taxonomy, Architecture, and Future Directions," *Wireless Personal Communications*, vol. 137, no. 1, pp. 153-197, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[28] Florian Sieck et al., "Teejam: Sub-Cache-Line Leakages Strike Back," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2024, no. 1, pp. 457-500, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[29] Jean-Philippe Aumasson, *Serious Cryptography: A Practical Introduction to Modern Encryption*, No Starch Press, Inc, 2024. [Google Scholar] [Publisher Link]

[30] Wala'a Essa Al-Ahmadi et al., "A Secure Fingerprint Hiding Technique Based on DNA Sequence and Mathematical Function," *PeerJ Computer Science*, vol. 10, pp. 1-33, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[31] Fan-feng Shi et al., "Heterogeneous Parallel Computing-Based Real-Time Chaotic Video Encryption and its Application to Drone-Oriented Secure Communication," *Chaos, Solitons & Fractals*, vol. 181, pp. 1-14, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[32] Liang Liu, Tong Wang, and Zhijun Wu, "Denial of Firewalling Attacks (DoF): Detection, Defense, and Challenge," *2024 Asian Conference on Communication and Networks (ASIANComNet)*, Bangkok, Thailand, pp. 1-8, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[33] Asmita Pal et al., "Camouflage: Utility-Aware Obfuscation for Accurate Simulation of Sensitive Program Traces," *ACM Transactions on Architecture and Code Optimization*, vol. 21, no. 2, pp. 1-23, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[34] Jiliang Zhang et al., "Timing Side-Channel Attacks and Countermeasures in CPU Microarchitectures," *ACM Computing Surveys*, vol. 56, no. 7, pp. 1-40, 2024. [CrossRef] [Google Scholar] [Publisher Link]