A Survey on Symmetric Cryptography: Techniques and Challenges

Ferial Najiantabriz, Amirhossein Arezoumand

Abstract

In the rapidly evolving field of digital security, symmetric cryptography plays a crucial role in protecting sensitive information. This survey paper provides an extensive overview of symmetric encryption algorithms, highlighting key operational mechanisms, historical developments, and advancements in cryptography techniques. We explore a range of symmetric algorithms beyond the well-known Advanced Encryption Standard (AES) and Data Encryption Standard (DES), discussing their respective strengths, vulnerabilities, and the context of their usage. The analysis covers various attack vectors, such as collision, boomerang, and square attacks, demonstrating how these methods exploit algorithmic weaknesses. This study not only compares the security features across different symmetric algorithms but also discusses their practical implications in current systems.

Index Terms: Cryptography, Symmetric Algorithms, Information Security, Encryption, Decryption

1 Introduction

In the rapidly changing world of computer security, the robustness of cryptographic algorithms against a variety of attacks is essential. There are two main types of encoding: symmetric and asymmetric encoding algorithms. Within these categories, various algorithms are utilized. Examples of symmetric encoding algorithms include AES, DES, 3DES, E-DES, Blowfish, RC2, RC4, and RC6, all of which employ bilateral algorithms.

Information security encompasses a series of measures, procedures, and strategies designed to prevent unauthorized access, disruptions, exposure, alterations, and interference with computer network resources. Strengthening the confidentiality, integrity, and availability of data involves continual efforts to fortify existing methods against persistent threats and to develop new techniques that are resistant to various types of attacks. Historical evidence indicates that encryption is one of the most effective methods for securing information, dating back to the Romans who used similar techniques to protect sensitive information and documents.

Data encryption involves converting data into unintelligible symbols using codes. This process, which relies on a single key known as symmetric key cryptography, uses the same key for both encryption and decryption. A secure channel is necessary to transmit the secret key between the sender and the receiver. Symmetric algorithms address two cipher modes: block and stream ciphers. Block ciphers operate on fixed-length strings of bits, called blocks, and process data in multiple similar

rounds where each round involves a substitution on one part of the data followed by a permutation that integrates the two halves. As the key expands, multiple sub-keys are used in each round. Symmetric key cryptography refers to cryptographic algorithms that require two distinct keys: one private and the other public. [27]

Although they are distinct, they are mathematically linked. The public key encrypts plaintext, whereas the private key decrypts ciphertext. As detailed in [15], asymmetric encryption methods are approximately 1,000 times slower than symmetric encoding, rendering them impractical for encrypting large volumes of data. Furthermore, to match the security potency of symmetric algorithms, asymmetric ones often require more robust keys.

Cryptography has always been the foundation of secure communications. As digital communication becomes more popular, the importance of employing robust cryptographic algorithms cannot be underestimated. The Data Encryption Standard (DES) was established in the early 1970s and serves as a fundamental encryption technique. However, its key size of 56 bits was eventually found insufficient against brute force attacks, which led to its decline in favor of more secure alternatives [21]. Despite this, DES's influence persists in the form of Triple DES, which aims to enhance security by increasing the key length.

This paper presents an analysis of various symmetric cryptography models, focusing on their specific vulnerabilities and strengths. We examine a variety of algorithms, including AES, DES, and others. We intend to present a comprehensive overview of how these models perform against different security challenges. Through this analysis, we seek to identify potential improvements and explore possible alternatives that could enhance their security frameworks. Our review aims to contribute significantly to the field of cryptography security.

1.1 Encryption and Decryption

Encryption involves transforming data into a non-recordable format, while decryption reverses this process by converting ciphertext back into plaintext. A cipher consists of two algorithms, responsible for both encoding and decoding. The effectiveness of a cipher hinges on its algorithm and a secret key, which is a concise set of symbols used to decrypt encrypted data [25]. Figure 1, shows how encryption and decryption works.

1.2 Goals of Cryptography

Cryptography serves numerous purposes, some of which are outlined below:

Authentication is the process of offering identity to a person to break special resource using keys.
Confidentiality is the ultimate target of encryption that confirms that only the cipher-key owner receives the message.
Data Integrity is the operation that has the access of modulating the database that belongs to a specific group or person.
Non-Repudiation ensures that both the sender and receiver acknowledge the delivery of the report.
Access Control confirms that only the group with correct authentication is eligible to log into the delivered message.

1.3 Organization

In this survey, we will explore various algorithms within the realm of symmetric cryptography. The initial focus will be outlined in Section 1, where we introduce the main topics of discussion and we follow with preliminaries in section 2. In Section 3, we discuss symmetric cryptography and provide a brief overview of various models within this category. In Sections 4 through 8, we explore various models of symmetric cryptography, each section dedicated to a different algorithm or family of algorithms. This detailed examination provides insights into their operational mechanisms, strengths, and vulnerabilities within the broader framework of cryptography security. In Section 9, we present a comparative analysis of the symmetric algorithms discussed, evaluating their relative merits and drawbacks and in Section 10, we discuss the prospective directions for our research. Finally, in Section 11, we draw conclusions from our survey, offering insights into the current landscape and future directions of symmetric cryptography.

2 Preliminaries

This section introduces key terms and concepts that we'll frequently discuss throughout this survey, making it easier for readers to follow along [3]. Plain Text is just regular text that we intend to send to someone else. Encryption is the method of transforming Plain Text into a form that only the intended sender and receiver can decode. The Encryption/Decryption Key is a secret code that the sender and receiver use to encrypt and later decrypt the message. In symmetric cryptography, the algorithm can be the same for both encrypting and decrypting, or it can be different for each in asymmetric cryptography.

Decryption is the reverse process where the Cipher Text is turned back into the original Plain Text. A Cipher is a specific method or algorithm used to encrypt text. Cipher Text is what you get after the text has been encrypted; it's the encrypted version of Plain Text. Block ciphers are algorithms that encrypt data in fixed-size blocks of bits consistently. It's a fundamental part of symmetric cryptography systems, which include well-known algorithms like AES (Advanced Encryption Standard) and DES (Data Encryption Standard). These systems depend on block ciphers because they can efficiently process large amounts of data while maintaining strong security.

3 Symmetric Cryptography

Cryptography is the practice of converting readable text (plain text) into an unreadable format (cipher text) to ensure data privacy. The term derives from "crypto," meaning hidden, and "graphy," meaning to write. It involves securing information through encryption, authentication, and access control. Cryptography is divided into two main types: Symmetric Key (Secret Key) cryptography and Asymmetric Key (Public Key) cryptography. This brief focuses on Symmetric Key cryptography, which uses a single key for both encrypting and decrypting data, providing a straightforward yet effective security solution. In Figure 2, the architecture for Symmetric Key Cryptography is shown.

In general, as we can see in Figure 2, several symmetric key cryptography algorithms are widely used, including AES, DES, 3DES, RC, and Blowfish. This section provides an overview of these fundamental symmetric key algorithms. We will explain them in details in the sections 4, 5, 6, 7 and 8.

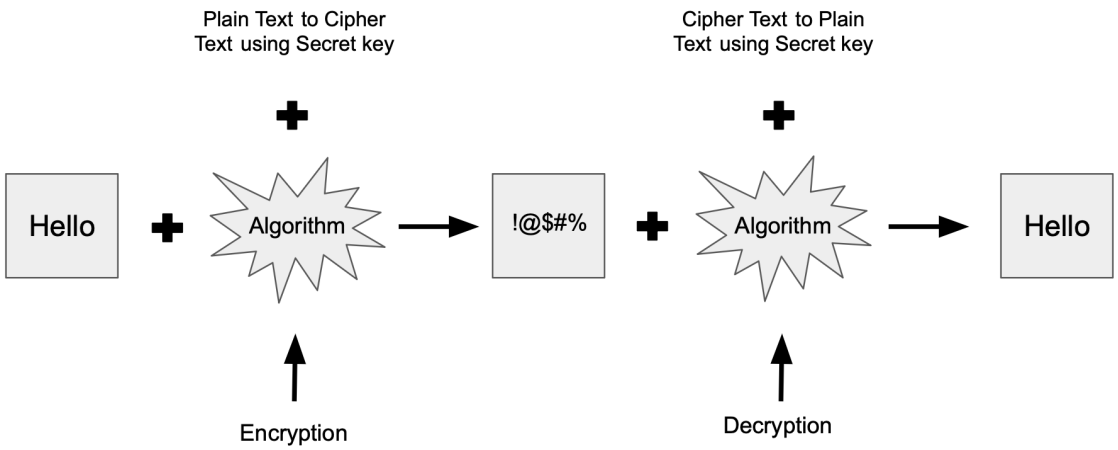

Figure 1: Symmetric Key Cryptography

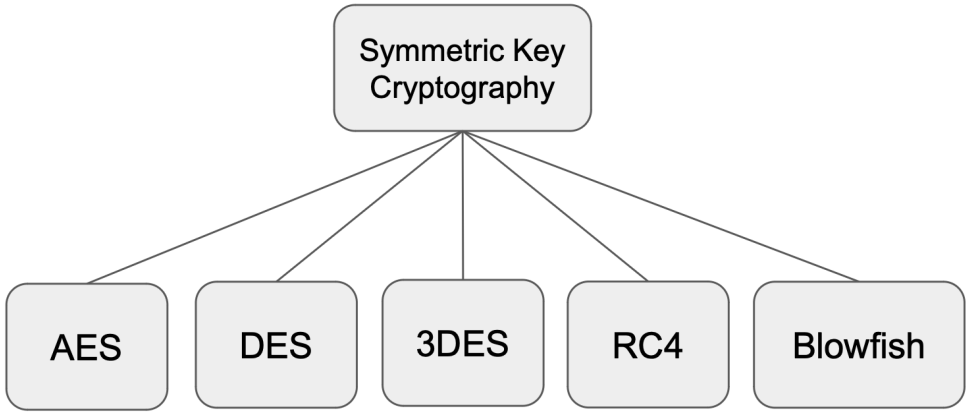

Figure 2: Classification of Symmetric Key Cryptography algorithms

3 - 1: Advanced Encryption Standard (AES)

The Advanced Encryption Standard (AES) was initiated by the National Institute of Standards and Technology (NIST) in January 1997. It is considered more secure than the DES algorithm and operates with a minimum block size of 128 bits for both encryption and decryption. The AES encryption process involves four main steps: byte substitution, row shifting, column mixing, and adding a round key. AES is versatile enough to protect sensitive and unclassified materials effectively. We will explain it in details in section 4.

3 - 2: Data Encryption Standard (DES)

Developed by IBM in 1997, this algorithm operates on a 64-bit block size. The encryption process is structured into 16 stages and incorporates eight S-Boxes. Initially, it shuffles the bits, then performs nonlinear substitutions, and concludes with an XOR operation to produce the final result. Each round's subkey is combined with the result using an XOR operation. The decryption process reverses the

order of the subkeys.

3 - 3: Triple Data Encryption Standard (3DES)
Triple DES is an advanced version of the DES algorithm, noted for its enhanced reliability and a total key length of 192 bits. The key is divided into three 64-bit subkeys. The operational process is similar to that of DES, with the notable distinction that the encryption and decryption sequence is executed three times: the data is encrypted with the first key, decrypted by the second, and re-encrypted with the third. Despite its strengths, Triple DES is not considered sufficiently robust for long-term data protection.

3 - 4: RC4 Algorithm
Developed by Ronald Rivest, this algorithm involves a dynamic exchange of state entries based on a variable key sequence ranging from 1 to 256 bytes in length. It generates a pseudo-random byte stream, which is then XORed with the plaintext to produce ciphertext. Notably, this encryption method is up to 10 times faster than the DES algorithm.

3 - 5: Blowfish Algorithm
Blowfish, designed by Bruce Schneier in 1993, is highly efficient among encryption algorithms. It supports variable key lengths from 32 to 448 bits and operates with a 64-bit block size. The encryption process begins with key expansion, creating an 18-entry P-array and four S-boxes, each with 256 32-bit entries. Data encryption is performed using XOR operations. Blowfish is particularly useful in applications where the key does not frequently change, providing a reliable alternative to other encryption methods. [10]
Many researchers, including Nadeem [23], have recognized the superiority of Blowfish for encryption, citing its numerous advantages over competing algorithms. Nadeem's studies also conclude that AES is significantly more advanced than both DES and 3DES. Furthermore, it was determined that DES outperforms 3DES in terms of processing efficiency, as 3DES requires three times as long to process information.

4 Advanced Encryption Standard (AES)

AES (Advanced Encryption Standard) is a symmetric block cipher that encrypts information in 128-bit blocks and can use keys of 128, 192, or 256 bits. Unlike DES (Data Encryption Standard), AES does not employ a Feistel network, instead using a more basic, yet robust, sequential process. The encryption process in AES is defined by multiple rounds of processing, which are different by key size—10 rounds for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys. Each round of AES encryption includes four distinct levels, enhancing the cipher's security and complexity:

- **SubBytes** — This is a non-linear substitution step where each byte of the data is replaced with another according to a predefined lookup table. This step introduces confusion into the data, making it harder to decipher without the key.
- **ShiftRows** — A transposition step that cyclically shifts the bytes in each row of the data block. The number of positions each row is shifted varies, adding further complexity and diffusion to the encryption.

- **MixColumns** — This mixing operation works on each column of the block, combining the four bytes in each column using a mathematical function. This step enhances the diffusion properties of the cipher.

- **AddRoundKey** — The simplest of the operations, this involves a bitwise XOR (exclusive OR) between the current block of data and a portion of the expanded key. This operation integrates the key into the encrypted data.

AES (Advanced Encryption Standard) is considered highly secure, largely tolerant of all forms of attack except brute force attacks. Such attacks, while potentially possible, would require an impractical amount of computational power with today's technology, rendering them unfeasible. AES also excels in performance; it is optimized for both hardware and software, functioning effectively across various platforms. This makes it faster and more secure than Triple-DES, which was once prevalent in financial services but is now increasingly seen as less efficient and robust by comparison.

In [9] Boura et al. explores into the Advanced Encryption Standard (AES) and its susceptibility to various cryptoanalytic attacks since its standardization in 2001. It highlights the development of attacks that exploit specific properties of the algorithm, primarily targeting reduced-round AES versions. Key to these attacks are distinguishers—non-random properties that identify anomalies in these versions—crucial for breaking block ciphers.

A significant advancement discussed is the discovery of a property within the AES round function R, which leverages well-chosen linear subspaces to create effective 5-round distinguishers. Although these distinguishers have revealed novel characteristics of AES, they were not initially applicable to key-recovery attacks.

The paper's primary goal is to present a general formulation of the mixture-differential distinguisher and the "multiple-of-8" property. These concepts are applied systematically to any cipher based on the Substitution-Permutation Network (SPN) structure. It examines the conditions necessary for these properties to appear. It also examines how the difference between two outputs of the AES round function remains consistent under an equivalence relation between plaintext groups. This analysis has resulted in a straightforward and concise validation of the distinguishing characteristics observed. It demonstrates that these properties do not depend on the branch number of the MixColumns function figure 3 shows AES algorithm.

In [36] authors outlines several protocols designed to boost the security and efficiency of data encryption and identity management within cloud computing systems. Among the proposals is an improved PKC-based certificateless group authenticated key agreement protocol, and an enhanced protocol that adds a signature to strengthen certificateless group key agreements. Furthermore, the paper presents several other protocols aimed at mitigating various security challenges inherent in cloud computing.

Additionally, they propose a modified data encryption algorithm tailored for cloud environments. This involves reevaluating the traditional Advanced Encryption Standard (AES) and suggesting a variant that incorporates random disturbance information to bolster data security. This new protocol is touted as being secure, efficient, and less demanding computationally than existing methods, potentially offering a more robust solution for protecting data in cloud platforms.

In [37] authors examines the Advanced Encryption Standard (AES) and its optimization for graphics processing units (GPUs), noting AES as the most prevalent encryption algorithm today. It details how GPUs, due to their superior parallel processing capabilities compared to central processing units (CPUs), are particularly effective for block cipher encryption like AES. This is crucial for applications

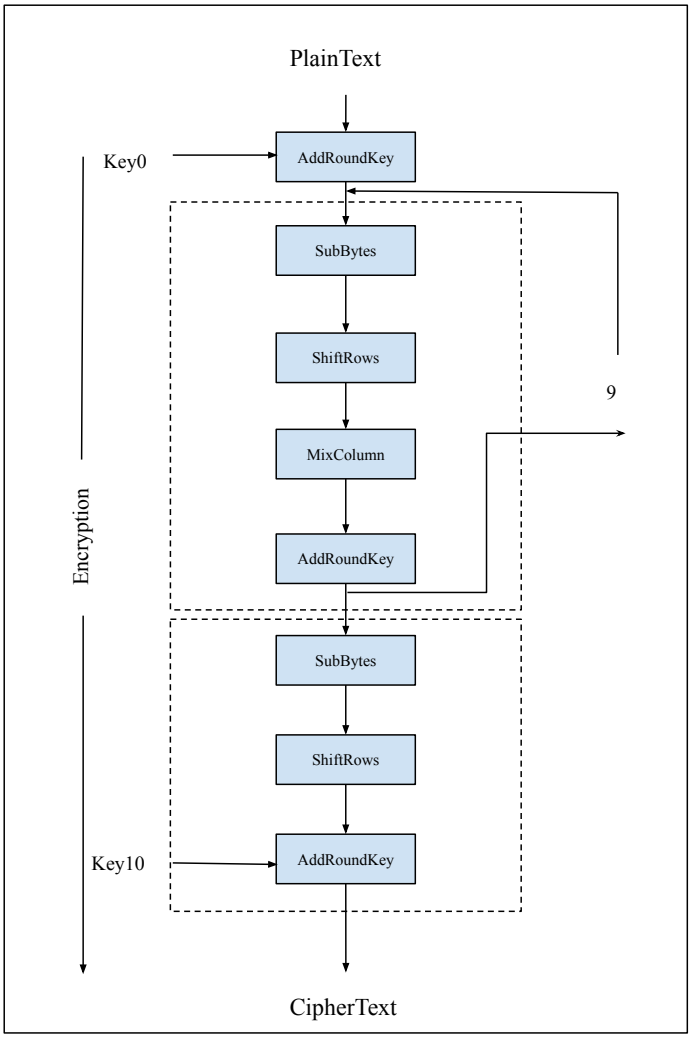

Figure 3: AES Algorithm Process

requiring rapid encryption of large volumes of data. The paper explores three primary techniques for implementing AES on GPUs:

Naive Implementation: This basic method involves a straightforward adaptation of AES to GPU without significant modifications.

Table-Based Implementation: Utilizes lookup tables to speed up the encryption process, trading off memory usage for faster performance.

Bitsliced Implementation: This approach arranges data in a way that allows simultaneous processing of multiple encryption operations, optimizing throughput and efficiency. For each method, the paper presents optimization strategies and performance outcomes, providing valuable insights into

how best to leverage GPU architecture for enhancing AES encryption.

4.1 Attacks on AES

This section offers concise summaries of existing attacks on the Advanced Encryption Standard (AES). It outlines the basic concepts behind each attack and touches on their complexities.

4.1.1 Boomerang

The boomerang attack [41] is a sophisticated technique used in cryptographic analysis to improve upon differential cryptanalysis by targeting the weaker segments of a block cipher. This method divides the cipher into two parts, E_1 and E_2. For each segment, a specific differential is identified—this refers to input differences that result in predictable output differences.

- **Preparation:** The attacker prepares two sets of inputs that will generate the targeted differentials when processed through the cipher.
- **Interaction of Differentials:** These inputs are sent through the cipher, and the way these differentials interact is crucial. The main innovation here is exploiting the cipher's vulnerabilities more efficiently than traditional methods by focusing on how these interactions unfold.

The technique employs what is known as a "boomerang distinguisher." This tool is used to verify if certain differential properties are consistent across both halves of the cipher:

- First, the attacker sends a modified version of the ciphertext back through E_2, similar to a boomerang returning.
- If the outputs match the expected differential from E_1, it confirms a successful exploitation of the cipher's structure.

This approach allows attackers to identify and leverage vulnerabilities by concentrating on the likelihood of differential combinations. It provides a formidable tool against ciphers previously considered resistant to simpler forms of differential attacks.

4.1.2 Collision Attack

Collision attacks [34]in cryptography are critical vulnerabilities where two distinct inputs produce the same hash output from a hash function, compromising its collision resistance property. These attacks pose significant risks to digital security, impacting applications like digital signatures and data integrity checks. The discovery of such vulnerabilities in popular hash functions such as MD5 and SHA-1 has led to widespread reevaluation of cryptographic practices. For instance, researchers demonstrated the practical implications of collision attacks on MD5 in 2004, where they created two different messages with the same MD5 hash, undermining the security of digital signatures based on this hash function

4.1.3 Square Attack

The Square attack [13] is a cryptanalytic method initially designed for the Square cipher, which has proven effective against AES due to its exploitation of AES's byte-oriented operations. This attack leverages the linear transformation layers of AES, focusing on invariant properties maintained across these layers. It involves constructing a set of plaintexts where each byte position holds all possible byte values exactly once. The attackers then partially encrypt these plaintexts through a limited number of rounds, analyzing the intermediate states to infer properties about the encryption process.

Key to the Square attack is its use of statistical analysis on the outputs after partial encryption, identifying patterns that could indicate weaknesses in the cipher's structure or hints towards parts of the encryption key. This technique is particularly potent against reduced-round versions of AES, typically effective up to 7 rounds, depending on the AES version. Although full recovery of the encryption key may require further complex adaptations, the Square attack highlights crucial vulnerabilities in symmetric key ciphers and emphasizes the need for thorough cryptanalysis when designing and evaluating encryption algorithms.

5 Data Encryption Standard (DES)

DES, or the Data Encryption Standard, is a widely recognized cryptographic system available to the public. Developed by IBM in the 1970s, it was subsequently adopted by the National Institute of Standards and Technology (NIST) as Federal Information Processing Standard 46 (FIPS PUB 46). DES operates as a block cipher, designed to encrypt and decrypt data blocks of 64 bits using a 64-bit key. [32, 26]

While the DES algorithm accepts a 64-bit input key, only 56 bits are actively used for encryption. This is because the least significant bit (right-most) in each byte functions as a parity bit, ensuring an odd number of 1s in each byte. These parity bits are not used in the encryption process, leaving only the seven most significant bits of each byte active, thereby reducing the effective key length to 56 bits. The algorithm performs 16 iterations, intertwining blocks of plaintext with key-derived values, transforming 64-bit input into 64-bit output through a series of steps. The same process and key are applied for decryption. Numerous vulnerabilities have been identified in DES, leading to its classification as an insecure block cipher. However, despite concerns about its security, DES continues to be extensively employed by financial services and various industries globally to secure sensitive online applications. [33]

Figure 4 illustrates the flow of the DES Encryption algorithm, which begins with an initial permutation, followed by sixteen rounds of block cipher operations, and concludes with a final permutation, essentially reversing the initial permutation.

Security Analysis on DES
In 1990, Eli Biham and Adi Shamir introduced the concept of differential cryptanalysis, unveiling a chosen-plaintext attack against DES that surpassed brute force in efficiency. The most potent attack against the full 16-round DES necessitates 247 chosen plaintexts, which can be transformed into a known plaintext attack, albeit requiring 255 known plaintexts. This analysis demands 237 DES oper-

ations and heavily relies on the structured optimization of the S-boxes within DES against differential cryptanalysis. Additionally, augmenting the number of rounds can bolster DES's resistance [8].

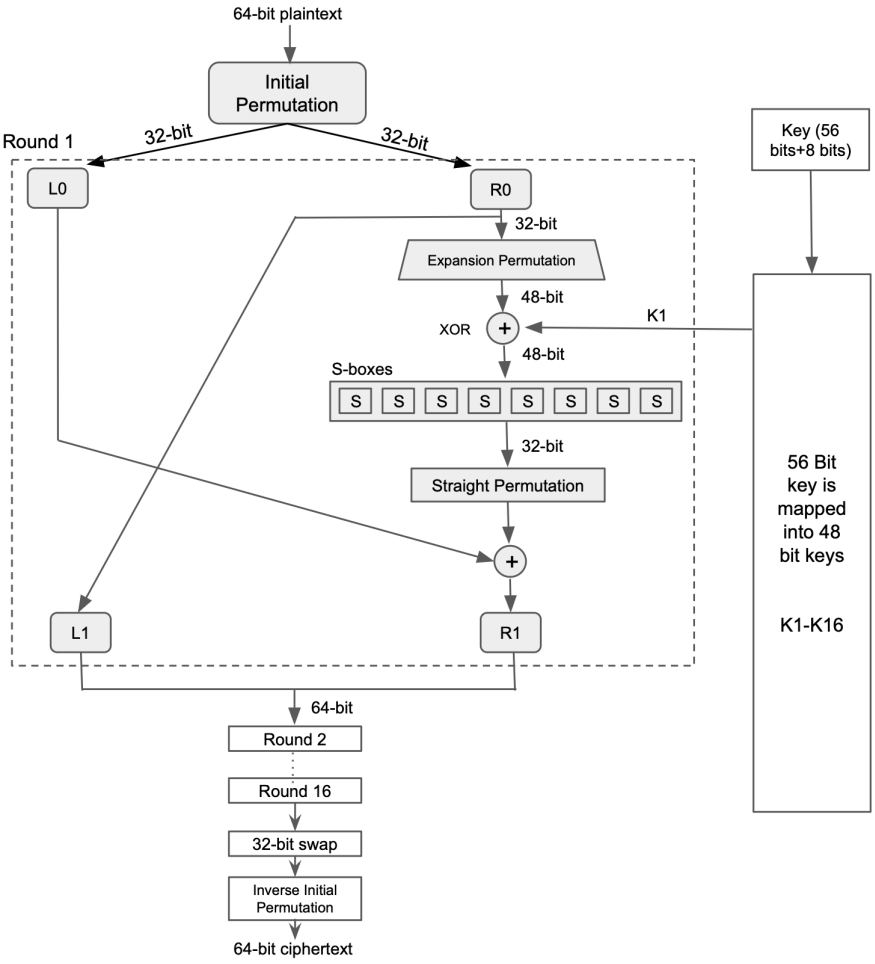

Figure 4: DES Algorithm Process

Linear cryptanalysis, conceived by Mitsuru Matsui, presents another form of cryptanalytic assault, leveraging linear approximations to elucidate the behavior of a block cipher [20]. This method can retrieve the key against the full 16-round DES with an average of 243 known plaintexts. Notably, a software implementation of this attack managed to uncover a DES key in 50 days using 12 HP9000/735 workstations, marking the most formidable attack to date [19]. Linear cryptanalysis, though newer than differential cryptanalysis, exhibits efficacy against reduced-round DES variants.
Based on the preceding analysis, it is evident that DES can offer a degree of security assurance through the optimization of S-box construction.

Educational Data Encryption Standard (E-DES)
The enhancement to DES, referred to as E-DES, involves several key transformations, including an increased key and block size, a refined F function, an advanced key schedule, and more complex permutation tasks [27]. Additionally, E-DES incorporates an element from AES, specifically the

substitution box. It primarily operates on a Feistel network structure with sixteen rounds, starting with an initial permutation of the plaintext. Each round consists of the following steps:

1. The permuted plaintext is split into two halves, left and right.
2. The right half is directly shifted to the left, and the left half undergoes an XOR operation with the output from the F function.

After completing the sixteen rounds, a final inverse permutation is performed to produce the encrypted text block. This process is depicted below in Figure 5. A notable difference between the S-box used in E-DES and that in AES is the tailored S-box recommended for each 8-bit block in E-DES, providing a unique 8-bit output from an 8-bit input, arranged in a 16x16 grid of bytes [27].

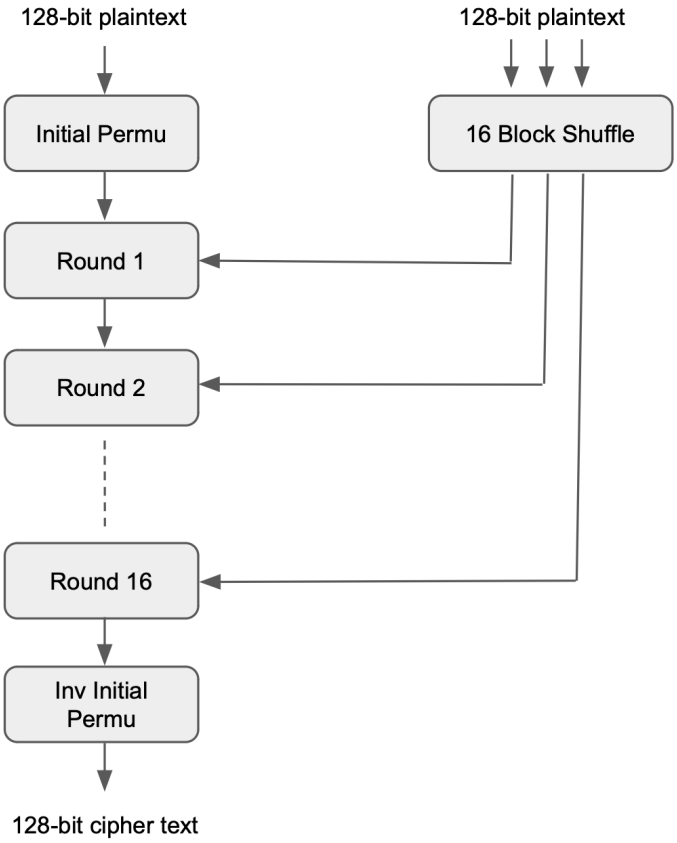

Figure 5: General Encryption Structure

6 Triple DES (3DES)

3DES, also known as Triple Data Encryption Algorithm (TDEA), was developed as an enhancement to DES, rectifying its vulnerabilities without necessitating the creation of an entirely new cryptographic system. DES, utilizing a 56-bit key, was deemed inadequate for securing sensitive data. 3-DES

resolves this by tripling the key size, achieved by consecutively applying the algorithm three times with three distinct keys. This results in a combined key size of 168 bits (3 times 56).

TDEA operates by utilizing three 64-bit DEA keys (K1, K2, K3) in Encrypt-Decrypt-Encrypt (EDE) mode. Here, plaintext is first encrypted with K1, then decrypted with K2, and finally encrypted again with K3. The standard delineates three keying options:

Option 1 the preferred choice, employs three mutually independent keys ($K1 \neq K2 \neq K3 \neq K1$), providing a keyspace of 3 × 56 = 168 bits.
Option 2 utilizes two independent keys and a third key identical to the first ($K1 \neq K2$ and K3 = K1), offering a of 2 x 56 = 112 bits.
Option 3 entails a bundle of three identical keys (K1 = K2 = K3), akin to the DES Algorithm.
These choices mirror the DES Algorithm. The triple iteration in 3-DES augments the encryption strength and average time, albeit at the expense of speed compared to other block cipher methods [16].

7 RC Algorithms

The RC algorithms comprise a series of symmetric-key encryption algorithms developed by Ron Rivest. The acronym ”RC” represent either Rivest's cipher or, more colloquially, Ron's code. Although their names share similarities, the algorithms are largely independent of each other. Six RC algorithms have been published, among which RC2, RC4, and RC6 are the most commonly used.

RC2 is a block encoding algorithm that was introduced all the way back in the year 1987. It is meant to replace the DES.RC2 applies exclusive size key from 1 byte to 128 bytes. Both the input and output block size of 64-bit per one. This algorithm was set to apply on 16-bit microprocessors. In the case having the encoding already done, the algorithm would work twice as fast as the DES on IBM [11].

RC4 The algorithm functions as a stream cipher, employing symmetric key encoding for both encryption and decryption processes. It XORs the database stream with a group of generated keys, and the key stream remains independent of the plaintext at all times. The Vernam stream cipher is widely used due to its simplicity and is employed in protocols such as SSL and WEP (Wireless Equivalent Privacy). WEP utilizes the RC4 algorithm for ensuring confidentiality. Initially considered secure, WEP was later compromised by the BEAST attack [11].

RC6 Introduced in 1997, RC6 is a block cipher with a 128-bit block size and supports key sizes of 128, 192, and 256 bits. In addition to this, RC6 is designed to address the requirements set forth by AES. It has been demonstrated to outperform the RC5 algorithm by providing enhanced security against attacks. RC6 employs four registers and requires fewer rounds while achieving higher throughput [11].

8 Blowfish Algorithm

Bruce Schneier, a prominent figure in cryptography, developed the Blowfish algorithm [29] and released it into the public domain. Blowfish operates with a variable-length key and a 64-bit block

cipher. Introduced in 1993, the algorithm remains unbroken to date. Its compactness allows for optimization in hardware applications.

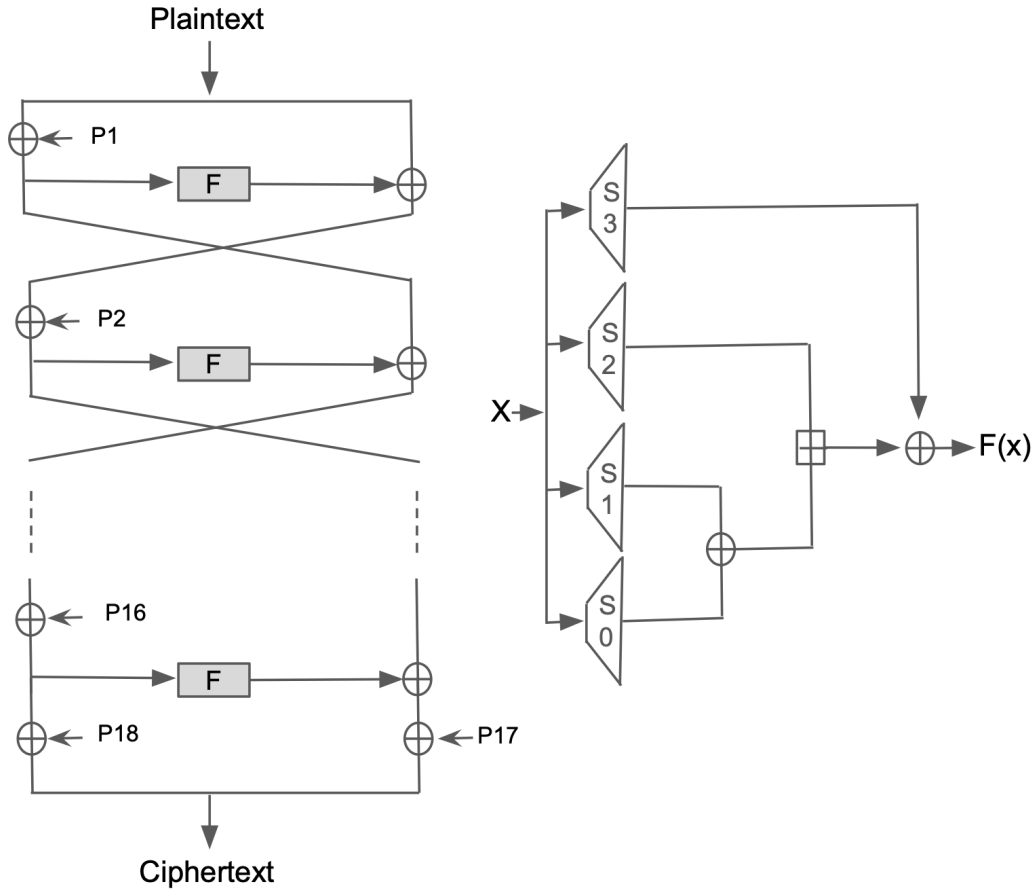

Figure 6: Blowfish Algorithm Process

The algorithm, depicted in Fig 6, comprises two main components: a key-expansion section and a data-encryption section. The key expansion process transforms a key of up to 448 bits into multiple sub-key arrays, resulting in a total of 4168 bytes. Data encryption is conducted through a 16-round network, wherein each round involves a key-dependent permutation and a substitution dependent on both the key and the data. All operations within the algorithm are XORs and additions on 32-bit words, with the only additional operations being four indexed array data lookups per round. Bruce Schneier demonstrated that differential cryptanalysis on Blowfish is feasible either against a reduced number of rounds or with knowledge of the F function. Nevertheless, the S-boxes in Blowfish are meticulously crafted to withstand attacks, as they are randomly generated. To date, no successful cryptanalysis against Blowfish has been reported. [39]

9 Comparison Between Symmetric Cryptography Algorithms

simpler in hardware than software. DES encrypts data in 64 bit block size and uses effectively a 56 bit key. 56 bit key space amounts to approximately 72 quadrillion possibilities. Even though it seems large but according to today's computing power it is not sufficient and vulnerable to brute force attack. Therefore, DES could not keep up with advancement in technology and it is no longer appropriate for security. Because DES was widely used at that time, the quick solution was to introduce 3DES which is secure enough for most purposes today.3DES is a construction of applying DES three times in sequence. 3DES with three different keys (K1, K2 and K3) has effective key length is 168 bits (The use of three distinct key is recommended of 3DES.). Another variation is called two-key (K1 and K3 is same) 3DES reduces the effective key size to 112 bits which is less secure.[5] Two-key 3DES is widely used in electronic payments industry. 3DES takes three times as much CPU power than compare with its predecessor which is significant performance hit. AES outperforms 3DES both in software and in hardware [42, 35]

The Rijndael algorithm, a modified version of which is known as the Advanced Encryption Standard (AES), has been chosen to supplant 3DES. Criteria for evaluating the Advanced Encryption Standard include security, software and hardware performance, suitability in constrained-space environments, and resistance to power analysis and other implementation attacks [6, 2, 24].

Joan Daemen and Vincent Rijmen submitted Rijndael for consideration as the Advanced Encryption Standard (AES). Its blend of security, performance, efficiency, implementability, and flexibility made it a fitting choice for AES. AES is designed to excel in software and hardware environments, exhibiting fast operation even on compact devices like smartphones and smart cards. It offers heightened security owing to its larger block size and longer keys—utilizing a 128-bit fixed block size and supporting keys of 128, 192, and 256 bits. Rijndael's adaptability allows for working with key and block sizes that are any multiple of 32 bits, ranging from a minimum of 128 bits to a maximum of 256 bits. According to NIST, AES is slated to replace 3DES, with both ciphers coexisting until the year 2030 to facilitate a gradual transition to AES. While AES theoretically boasts advantages over 3DES in terms of speed and efficiency, 3DES might perform faster in environments where support for it is well-established.

Many studies have endeavored to identify the most effective algorithm for encryption and decryption. Singh et al. [30] conducted a comprehensive comparison of various symmetric algorithms, including DES, 3DES, AES, and Blowfish. Despite the popularity of other methods in the field, their work revealed Blowfish to be the superior choice. Contrarily, AES was found to be less efficient, requiring longer processing times. Similarly, Cornwell [12] concluded that the Blowfish algorithm could maintain security over an extended period without any apparent breaches. According to Cornwell, Blowfish surpasses other algorithms in both security and efficiency. However, further research is warranted to validate and refine the findings presented. In another study by Tamimi [1], the ECB and CBC modes were employed to assess performance. Consistent with previous research, Blowfish emerged as the optimal choice due to its superior efficiency compared to AES and its ability to process data more effectively.

Numerous authors and researchers have recognized Blowfish as an exemplary method for encryption and decryption, with Nadeem [22] highlighting its numerous advantages as a means to outperform

competing algorithms. Nadeem's work also concluded that AES surpasses DES and 3DES in terms of development. Additionally, DES was found to be superior to 3DES, which requires three times the processing time.

In another study by Dhawan [14], AES outperformed other algorithms in terms of operations per second under varied user loads and response time in multiple user load scenarios. Singh et al. [31] conducted a comparison of popular encryption algorithms, including AES, DES, 3DES, and Blowfish, focusing on security and energy consumption. While AES was deemed superior to the basic form of Blowfish in their study, they suggested strengthening Blowfish against various attacks by adding extra keys to replace the old XOR operation.

Agrawal et al. [40] confirmed the superiority of Blowfish algorithms after extensive research on DES, 3DES, AES, and Blowfish. They highlighted the Blowfish algorithm's superior key size and security, attributing its F function to enhancing security levels for encoding 64-bit plaintext databases. Furthermore, the Blowfish algorithm demonstrated faster performance compared to other common algorithms with identical key encoding.

Mandal et al. [18] found that AES stands out from 3DES and DES in terms of throughput and decoding time. Apoorva et al. [7] concluded that Blowfish is the best algorithm in terms of security and processing time, as it consumes less time compared to others.

In this paper, Mandal et al. [17] conducted a comparative analysis of two widely adopted symmetric encryption methods: the Data Encryption Standard (DES) and the Advanced Encryption Standard (AES). Their comparison focused on several key metrics including the avalanche effect caused by a single-bit variation in plaintext while keeping the key constant, the avalanche effect resulting from a single-bit variation in the key while maintaining the plaintext constant, the memory requirements for implementation, and the simulation time necessary for encryption. The avalanche effect refers to a crucial property of any encryption algorithm where even a minor alteration in either the key or the plaintext should result in a substantial change in the ciphertext.

$$\text{Avalanche Effect} = \frac{\text{Number of flipped bits in ciphered text}}{\text{Number of bits in ciphered text}} \tag{1}$$

The avalanche effect is significantly more pronounced in AES compared to DES. Additionally, DES requires more memory and longer simulation times than AES, indicating that AES is more efficient. Due to these advantages, AES is particularly well-suited for encrypting messages exchanged through chat channels and is also beneficial for applications involving monetary transactions.

Abdul et al. [15] studied various algorithms including AES, DES, 3DES, RC2, Blowfish, and RC6. Their comparison showed no significant difference in hexadecimal base encryption or base 64 ciphering. However, Blowfish outperformed others in transforming pocket-size data, while 3DES's performance was mediocre compared to DES. A larger key size could significantly improve battery life and processing time.

Thakur et al. [38] conducted a moderate comparison between DES, AES, and Blowfish, concluding that Blowfish is the best-performing algorithm among the three. Alam et al. [4] demonstrated that 3DES consumes more energy and processes less input than DES due to its triple encryption feature. On the other hand, RC2 proved to be faster due to smaller throughput sizes compared to Blowfish. Blowfish has a larger input value compared to 3DES, DES, CAST-128, IDEA, and RC2, yet it consumes the least power. Overall, Blowfish excels in time, throughput, and power consumption. Saini [28] summarized that superior algorithms gain popularity for their efficiency, achieving a balance between feasibility and acceptance in cryptography.

Algorithm	Key Size	Block Size	Round	Structure	Flexible	Features
DES	64 bits	64 bits	16	Festiel	No	Not Strong Enough
AES	128, 192, 256 bits	128 bits	10, 12, 14	Substitution Permutation	Yes	Security is excellent. It is best in security and Encryption performance
Blowfish	32-448	64 bits	16	Festiel	Yes	Fast Cipher in SSL
RC2	8,128,64 by	64 bits	16	Festiel	-	Good and fast Security
RC4	Variable	40-2048	256	Festiel Stream	Yes	Fast Cipher
RC6	128 bits to256 bits	128 bits	20	Festiel	Yes	Good Security
E-DES	1024 bits	128 bits	16	Festiel	-	Good Security and fast Speed

Table 1: Comparison Between All Cryptography Algorithms Previously Discussion

10 Future research

In the field of symmetric cryptography, the continuous refinement of existing algorithms like AES and DES remains vital as new security challenges emerge. Future research should prioritize enhancing the resilience of these algorithms against sophisticated attack methods, such as side-channel attacks and advanced cryptanalysis techniques. As computational capabilities grow, particularly with the advent of quantum computing, the cryptographic community must also explore the development of new symmetric algorithms that are inherently resistant to quantum attacks. Additionally, the integration of cryptographic solutions in increasingly diverse environments, from cloud computing platforms to Internet of Things (IoT) devices, calls for adaptive and scalable cryptographic methods that maintain security without sacrificing performance. This would include researching more efficient modes of operation and investigating the potential of lightweight cryptography to provide secure yet resource-efficient solutions for constrained environments.

11 Conclusions

In conclusion, this survey has discussed the wide field of symmetric cryptography, highlighting the operational mechanisms, historical context, and key vulnerabilities of prominent encryption algorithms such as AES, DES, and others. We have seen that despite the robustness of modern algorithms like AES, challenges persist in the form of sophisticated attacks and evolving computational capabilities. Our comparative analysis highlights the nuanced trade-offs between different cryptographic systems, revealing that no single algorithm perfectly addresses all security needs. Moreover, the examination of various symmetric algorithms has demonstrated their pivotal role in securing digital communications across diverse platforms and applications. As we move forward, the continuous advancement in cryptanalysis and computational power will undoubtedly stimulate further innovations and improvements in encryption technologies.

The cryptography must remain sharp and bold in enhancing the security features of these algorithms to balance emerging threats. Embracing these challenges not only strengthens current cryptography practices but also ensures the resilience of digital security infrastructures in the future.

References

[1] Abdel-Karim, A.: Performance analysis of data encryption algorithms (2006) 14

[2] Abomhara, M., Zakaria, O., Khalifa, O.O., Zaidan, A.A., Zaidan, B.B.: Enhancing selective encryption for h.264/avc using advance encryption standard. International Journal of Computer and Electrical Engineering (IJCEE) **2**(2), XXX (April 2010) 14

[3] Abood, O.G., Guirguis, S.K.: A survey on cryptography algorithms. International Journal of Scientific and Research Publications (IJSRP) (2018) 3

[4] Alam, M.I., Khan, M.R.: Performance and efficiency analysis of different block cipher algorithms of symmetric key cryptography. International Journal of Advanced Research in Computer Science and Software Engineering **3**(10), 34– (2013) 16

[5] Alanazi, H., Bahaa, B., Zaidan, A., Jalab, H., Shabbir, M., Al-Nabhani, Y.: New comparative study between des, 3des and aes within nine factors (03 2010) 14

[6] Alanazi, H., Jalab, H.A., Zaidan, A.A., Zaidan, B.B.: New frame work of hidden data with in non multimedia file. International Journal of Computer and Network Security **2**(1), 46–54 (January 2010) 14

[7] Apoorva, Y.K.: Comparative study of different symmetric key cryptography algorithms. International Journal of Application or Innovation in Engineering and Management **2**(7), 204–206 (2013) 15

[8] Biham, E., Shamir, A.: Differential cryptanalysis of the full 16-round des. In: Advances in Cryptology-CRYPTO '92 Proceedings. pp. 487–496. Springer-Verlag (1993) 10

[9] Boura, C., Canteaut, A., Coggia, D.: A general proof framework for recent aes distinguishers. IACR Transactions on Symmetric Cryptology **2019**(1), 170–191 (2019) 6

[10] Chandra, S., Bhattacharyya, S., Paira, S., Alam, S.S.: A study and analysis on symmetric cryptography. In: 2014 International Conference on Science Engineering and Management Research (ICSEMR). pp. 1–8 (2014). https://doi.org/10.1109/ICSEMR.2014.7043664 5

[11] Charbathia, S., Sharma, S.: A comparative study of rivest cipher algorithms. International Journal of Information & Computation Technology **4**, 1831–1838 (2014) 12

[12] Cornwell, J.W., Columbus, G.A.: Blowfish survey. Tech. rep., Department of Computer Science, Columbus State University (2012) 14

[13] Daemen, J., Knudsen, L., Rijmen, V.: The block cipher square, fast software encryption (fse'97), lncs 1267 (1997) 9

[14] Dhawan, P.: Performance comparison: Security design choices. Tech. rep., Microsoft Developer Network (2002) 15

[15] Elminaam, D.S.A., Kader, H.M.A., Hadhoud, M.M.: Performance evaluation of symmetric encryption algorithms. IJCSNS International Journal of Computer Science and Network Security **8**(12), 280–286 (2008) 2, 15

[16] Kakkar, A., Singh, M.L., Bansal, P.K.: Comparison of various encryption algorithms and techniques for secured data communication in multinode network. International Journal of Engineering and Technology **2**(1), 87–92 (January 2012) 12

[17] Mandal, A.K., Parakash, C., Tiwari, M.A.: Performance evaluation of cryptographic algorithms: Des and aes. In: IEEE Students' Conference on Electrical, Electronics and Computer Science. pp. 1–5 (2012) 15

[18] Mandal, P.C.: Superiority of blowfish algorithm. International Journal of Advanced Research in Computer Science and Software Engineering **2**(9), 196–201 (2012) 15

[19] Matsui, M.: The first experimental cryptanalysis of the data encryption standard. In: Advances in Cryptology-CRYPTO '94 Proceedings. pp. 1–11. Springer-Verlag (1994) 10

[20] Matsui, M.: Linear cryptanalysis method for des cipher. In: Advances in Cryptology-EUROCRYPT '93 Proceedings. pp. 386–397. Springer-Verlag (1994) 10

[21] Menezes, A.J., Van Oorschot, P.C., Vanstone, S.A.: Handbook of applied cryptography. CRC press (2018) 2

[22] Nadeem, A., Javed, M.Y.: A performance comparison of data encryption algorithms. In: Information and Communication Technologies, 2005. ICICT 2005. First International Conference on. pp. 84–89 (August 2005) 14

[23] Nadeem, A., Javed, M.: A performance comparison of data encryption algorithms. pp. 84 – 89 (09 2005). https://doi.org/10.1109/ICICT.2005.1598556 5

[24] Naji, A.W., Hameed, S.A., Zaidan, B.B., Al-Khateeb, W.F., Khalifa, O.O., Zaidan, A.A., Gunawan, T.S.: Novel framework for hidden data in the image page within executable file using computation between advance encryption standard and distortion techniques. International Journal of Computer Science and Information Security (IJCSIS) **3**(1), 73–78 (August 2009) 14

[25] Omar, G.A., Elsadd, M.A., Guirguis, S.K.: Investigation of cryptography algorithms used for security and privacy protection in smart grid. In: Power Systems Conference (MEPCON), 2017 Nineteenth International Middle East. pp. 644–649. IEEE (December 2017) 2

[26] Ramanujam, S., Karuppiah, M.: Designing an algorithm with high avalanche effect. IJCSNS International Journal of Computer Science and Network Security **11**(1), 106–111 (January 2011) 9

[27] Riman, C., Abi-Char, P.E.: Comparative analysis of block cipher-based encryption algorithms: A survey. Information Security and Computer Fraud **3**(1), 1–7 (2015) 2, 10, 11

[28] Saini, B.: Survey on performance analysis of various cryptographic algorithms. International Journal of Advanced Research in Computer Science and Software Engineering **4**(4), 1–4 (2014) 16

[29] Schneier, B.: The blowfish encryption algorithm. Retrieved October 25, 2008, http://www.schneier.com/blowfish.html (2008) 12

[30] Singh, G., Kumar, A., Sandha, K.S.: A study of new trends in blowfish algorithm. International Journal of Engineering Research and Application (2011) 14

[31] Singh, S.P., Maini, R.: Comparison of data encryption algorithms. International Journal of Computer Science and Communication **2**(1), 125–127 (2011) 15

[32] Srilaya, S., Velampalli, S.: Performance evaluation for des and aes algorithms- an comprehensive overview. In: 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT). pp. 1264–1270 (2018). https://doi.org/10.1109/RTEICT42901.2018.9012536 9

[33] Stallings, W.: Cryptography and Network Security: Principles and Practice. Pearson Education/Prentice Hall, 5 edn. (2011) 9

[34] Stevens, M., Lenstra, A., De Weger, B.: Chosen-prefix collisions for md5 and colliding x. 509 certificates for different identities. In: Advances in Cryptology-EUROCRYPT 2007: 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Barcelona, Spain, May 20-24, 2007. Proceedings 26. pp. 1–22. Springer (2007) 8

[35] Taqa, A., Zaidan, A.A., Zaidan, B.B.: New framework for high secure data hidden in the mpeg using aes encryption algorithm. International Journal of Computer and Electrical Engineering (IJCEE) **1**(5), 566–571 (December 2009) 14

[36] Teng, L., Li, H., Yin, S., Sun, Y.: A modified advanced encryption standard for data security. Int. J. Netw. Secur. **22**(1), 112–117 (2020) 6

[37] Tezcan, C.: Optimization of advanced encryption standard on graphics processing units. IEEE Access **9**, 67315–67326 (2021) 6

[38] Thakur, J., Kumar, N.: Des, aes and blowfish: Symmetric key cryptography algorithms simulation-based performance analysis. International Journal of Emerging Technology and Advanced Engineering **1**(2), 6–12 (2011) 16

[39] Vaudenay, S.: On the weak keys in blowfish. In: Fast Software Encryption, Third International Workshop Proceedings. pp. 27–32. Springer-Verlag (1996) 13

[40] Verma, O.P., Agarwal, R., Dafouti, D., Tyagi, S.: Notice of violation of ieee publication principles peformance analysis of data encryption algorithms. In: Electronics Computer Technology (ICECT), 2011 3rd International Conference on. vol. 5, pp. 399–403 (April 2011) 15

[41] Wagner, D.: The boomerang attack. In: International Workshop on Fast Software Encryption. pp. 156–170. Springer (1999) 8

[42] Zaidan, A.A., Zaidan, B.B., Jalab, H.A.: A new system for hiding data within (unused area two + image page) of portable executable file using statistical technique and advance encryption standard. International Journal of Computer Theory and Engineering (IJCTE) **2**(2) (2010) 14