International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

An Enhanced Three Layer Cryptographic Algorithm for Cloud Information Security

Anjana[1], Dr. Ajit Singh[2]

Submitted: 21/12/2023 **Revised**: 27/01/2024 **Accepted**: 09/02/2024

Abstract: The rapid adoption of cloud computing has ushered in unparalleled opportunities for efficient data storage and processing, but it also brings forth significant challenges related to information security. This study focuses on the design and analysis of effective techniques for enhancing information security in cloud computing environments, with a particular emphasis on the hybrid encryption technique. This research delves into the theoretical underpinnings of AES+ChaCha20 and SHA-3 evaluates its suitability for cloud-based applications, considering factors such as encryption strength, computational efficiency, and resistance to cryptographic attacks. The SHA-3 algorithm is employed to generate fixed-size hash values from input data. In this study, the plain text file is encrypted using the AES+ChaCha20, the cipher text file is hashed using the SHA3 algorithm to produce a message digest. Through a comprehensive analysis, including performance benchmarks and security assessments, this study aims to provide a nuanced understanding of the effectiveness of AES+ChaCha20 in mitigating common threats in cloud computing environments. The outcomes of this research contribute valuable insights to the on-going discourse on information security in cloud computing, offering a foundation for the development and implementation of robust security measures. As organizations increasingly rely on cloud services, the findings of this study are poised to inform best practices for securing data and ensuring the confidentiality and integrity of information in cloud-based systems.

Keywords: *Advanced Encryption Standard, SHA 3, Cloud computing, information security, cloud storage, ChaCha20 encryption, storage security.*

1. Introduction

The cloud computing model and distribution architecture are founded on the Internet. Its primary objective is to store sensitive information swiftly and securely. A centralized collection of resources, including as servers, storage, networks, services, and applications, may be accessed via the internet from anywhere in the world thanks to cloud computing [1]. The scientific and industrial communities are now focusing on cloud computing. Cloud computing has the potential to improve scalability, availability, and dependability, among other attributes. Application security, user authentication, access control, and data security are all important [2] considerations there are hazards related to cloud computing. Large organizations are hesitant to migrate to the cloud due to security concerns, despite the fact that many organizations already store confidential information in the cloud [3]. The exponential expansion of sensitive data stored on cloud platforms has significantly increased its susceptibility to security breaches [4]. Security in cloud computing includes ideas, for example, organize security, hardware and control methodologies sent to ensure information, applications and foundation related with cloud computing [5]. This life-saving device must evolve into a network of interconnected items in a networked world, allowing doctors to perform remote procedures on individuals their residences and energy suppliers. Lead the infrastructure with adequate effectiveness and a plan for a dire national situation protection [6, 7]. Cloud computing refers to the process of using networked resources to do computations and send them via the Internet. Instead of locally storing or manually updating data and application preferences, they may be managed via the internet [8][9]. Remote locations enable people and organizations to use software and gear that is overseen by third-party entities. This network architecture is often referred to as a "cloud" network. Cloud resources provide limitless scalability, may be accessed at any time, and used as needed. It provides a comprehensive range of services on the internet, tailored to the user's specific needs, including operating systems, networks, hardware, software, resources, and storage. The level of acceptability for every computer paradigm is determined by its advantages and disadvantages [10]. The architecture of a cloud system consists of its features, delivery methodology, and deployment model. The key features of cloud computing are on-demand self-service, wide network access, resource pooling, quick scalability, and usage-based billing [11]. Moreover, due to the foundation of two essential components, namely

Department of CSE & IT, Bhagat Phool Singh Mahila Vishwavidyalaya, Khanpur Kalan,
Sonipat, Haryana, India
ORCID: 0000-0003- 1511-6471
saroha.anjana@gmail.com, bpsmv.ajit@gmail.com

cloud computing and networking, the cloud heavily relies on internet connection and infrastructure. The network may be used for cloud computing (CC) and other applications in many cloud applications [12]. Consequently, an increasing number of application service providers (ASPs) [13] are gaining a clear understanding of the distinction between real use and maintenance. The ASP regularly evaluates the rental services to determine demand predictions and takes appropriate choices about goal and resource allocation [14].

Cloud computing is primarily responsible for recent technological advances in the field of information technology [15]. As the majority of businesses, municipalities, institutions, etc. increase their use and processing of data, The cloud storage service has become one of the most popular and indispensable. Cloud computing allows users to access and store data and programs via the internet, as opposed to a hard drive. From any Internet-connected device, users can access documents and utilize applications[16]. The internet is depicted as a cloud in the cloud computing diagram shown in Figure 1.

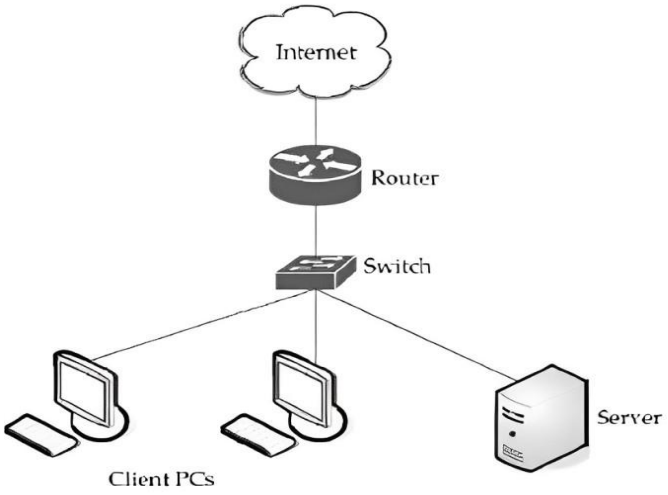

Fig 1: Representation of cloud in a network [15]

Customers of cloud computing service providers such as Google, Microsoft, and Amazon receive cloud computing resources and services according to a business model that dynamically utilizes the resources and services to meet customer demand. Due to the massive quantity of data, cloud storage security is a major concern, and service providers must ensure the privacy and confidentiality of customer and user data during transfer, retrieval, and storage. This level of security can be achieved by using multiple encryption algorithms and defence systems extensively. Hybrid cryptosystems were developed to combine the simplicity of asymmetric key cryptography, which does not need sharing the recipient's secret, and the effectiveness of symmetric key encryption. On the basis of the idea of hybrid cryptosystems in cloud storage, the implementations and tactics outlined below [17] will be discussed. Smaller chips, fewer energy usage, and improved performance are all achieved through the application of the ECC to produce cryptographic keys more quickly, simply, and effectively. Blowfish is used in many products, including secure encrypted email, password management systems, and backup software, and is immune to viruses. Due to the small number of iterations, Blowfish has a relatively basic structure for a block cipher. In the cloud, encrypted data is stored using the blowfish technique. Additionally, the EC public 3 key is used to encrypt the blowfish key. The decrypted key from the EC private key is used by Blowfish to decrypt data. blowfish encryption is used to communicate the method, and receiving the uploaded method requires decrypting the blowfish key [18]. The use of both symmetric and asymmetric encryption methods is referred to as "hybrid cryptography." You can combine the speed and strength of the two algorithms if you can utilize several algorithms of various types to strengthen the encryption. Cloud storage systems are protected using this technique [19]. To highlight the difference between less secure and secure systems, two techniques are used. AES is used to encrypt data or text, while RSA is used to encode credentials. The second, safer technique uses both Blowfish and AES. As cloud computing becomes more prevalent in our daily lives, researchers' interest in data security in the cloud, as well as techniques to encrypt that data and how to speed up its encryption, is growing. These industries include those in the military, real estate, banking, and health. The idea of "cloud computing" makes use of the internet to offer many services, including software, systems, data storage, and many more [20]. The many services that cloud computing offers exhibit three paradigms. SaaS

(software as a service) When utilizing this kind of cloud service, the user is unable to control the cloud network's components, services, memory, or operating system. Several partial parameters are at the user's control [21]. (ii) Platform as a Service (PaaS): With the aid of this technology, users can create a variety of apps utilizing a variety of programming languages. These applications can be made using the services, resources, and tools that service providers offer Python is one of the programming languages utilized, for example, for developing applications for Google App Engine. Infrastructure as a Service (IaaS) The infrastructure that supports the cloud is virtualized in this configuration [22]. This system is made up of virtual servers with limited storage and processing capabilities. Cryptography is the science that prevents data from being stolen and interpreted by uninvited outsiders by converting it from its readable and interpretable form to a form that is unintelligible to

- Data Encryption Standard, Blowfish, and others.

undesired outsiders. "Encryption" is the process of converting understandable data into unintelligible data with the use of a secret key. "Decryption" refers to the process of converting encrypted data into plain text using a secret key. Depending on the kind of key or non-key employed, cryptography can be divided into one of four groups: hybrid encryption, hash function, symmetric encryption, and asymmetric encryption.

- Secret key encryption and symmetric key encryption employs a single the same key is used for both decryption and encryption, making it a single key. In this sort of encryption, as seen in Figure 2, the recipient uses the same key for both encryption and decryption. The sender is in charge of key management. Among the most popular symmetric encryption techniques are Data Encryption Standard, Advanced Encryption Standard, Triple

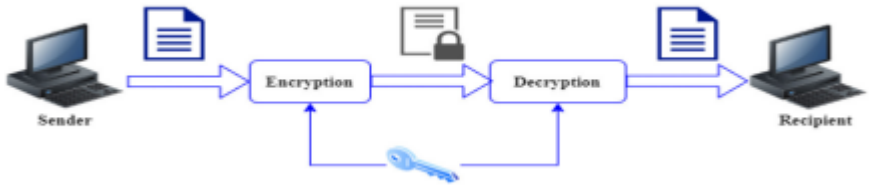

Fig 2. Symmetric key encryption

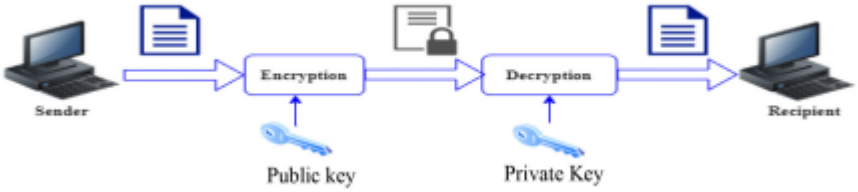

Fig 3. Symmetric key encryption [22]

- Asymmetric key encryption, commonly referred to as public key encryption, employs various keys, such as the public key, for encryption and decryption. The private key is only disclosed to a select few persons, whereas the public key is accessible to everyone as its name suggests. The diagram in Figure 2 illustrates asymmetric encryption. Adleman, Rivest, and Shamir the most popular asymmetric encryption methods are RSA, Diffi-Hillman DH, and Elliptic Curve Cryptography ECC.
- Hashing encryption is one method of encryption that differs from the others in that it encrypts data without using a key. Instead, a string of random attributes with a predetermined length is created from plain text using the hash function. Data of any size are transformed using a mathematical technique called the message digest one-way function into a fixed-size hash value. When a message digest is being used, it is hard to find or get the original string back. Message Digest and Secure Hash Algorithm are the two most used hashing-based encryption methods.
- Hybrid encryption is one type of encryption that permits the employment of a variety of algorithms, either of the same type of encryption or of distinct sorts. In this instance, it is possible to combine the speed and strength of various algorithms to strengthen the encryption.

2. Literature Review

Anjana (2022) [27] studied the delves into the critical realm of information security in cloud computing, offering a hybrid cryptographic solution utilizing RSA, Blowfish, and MD5 encryption techniques. The paper intelligently addresses the growing concerns surrounding data privacy and security in cloud storage services, particularly focusing on the lack of control by data owners and potential threats during data transmission. The comprehensive approach proposed the incorporates RSA Partial encryption before data transmission to the cloud server, followed by MD5 hashing to ensure data integrity. The systematic processes of encryption/decryption, uploading data to the cloud, and hashing are well-articulated, providing clarity to readers about the methodology. One of the strengths of the article lies in its emphasis on the practical application of encryption techniques within the cloud computing context. The integration of well-established cryptographic algorithms like RSA, Blowfish, and MD5 adds credibility to the proposed solution. However, it could benefit from a more in-depth discussion on the performance metrics and computational overhead introduced by the hybrid cryptographic solution. Furthermore, insights into potential limitations or challenges in implementing the proposed solution would contribute to a more balanced analysis.

Ramachandra et al. (2022) [28] proposed that big data analysis has emerged as a key academic subject in recent decades. Therefore, big data security provides Cloud application security and monitoring for hosting very sensitive data in order to enable Cloud platforms. Nevertheless, the preservation and protection of large-scale data has emerged as a concern that limits the organization's ability to make use of Cloud services. The current privacy-preserving methods have shown several limitations, including inadequate protection of data privacy and accurate data analysis, inefficiency in performance, and total dependence on third parties. To address this problem, the Triple Data Encryption Standard (TDES) technique is suggested as a means of ensuring the security of large amounts of data in the Cloud environment. The suggested TDES methodology offers a more straightforward approach by augmenting the key sizes in the Data Encryption Standard (DES) to enhance protection against assaults and safeguard data privacy. The experimental findings demonstrated the efficacy of the suggested TDES approach in ensuring security and privacy for large-scale healthcare data inside the Cloud environment. The TDES approach demonstrated reduced encryption and decryption time in comparison to the IFHDS method for Healthcare Data Security.

Sudharson et al. (2022) [18] described the development of the IT industry requires distributed computing, but cloud information security is also essential. Data stored in the cloud has serious reliability issues, unfortunately. Even though cloud-based frameworks are naturally more sophisticated and secure than conventional PC hardware, they nevertheless have a number of information security issues to deal with, both internally and externally. Security insurance is encouraged by a robust decentralized storage framework that is investigated. When a user uploads material to cloud storage, they are also given access to the corresponding private key. RSA-based methods use a cryptographic computation developed lately called Client End-generated Encrypted Keycode CEEK to secure distributed computing settings. It automates the monitoring of suspect devices, lessening the stress on the outsider and enhancing security checks by creating key-code using Device motherboard numbers, Device HDD numbers, and client passwords.

Chinnasamy et al. (2021) [17] examined the cloud environment, services are shared between all servers, users, and people. Since the original data form can be viewed, misused, and destroyed, cloud providers face challenges ensuring file protection when processing and transmitting data. Security in the cloud is an important issue. In order to secure the cloud, numerous studies have been proposed. Cryptography is used to address the security issue and achieve the CIA property confidentiality, integrity, and availability. Cryptography is the most reliable way for protecting data during storage and transmission. There are limitations to both symmetrical and asymmetrical designs. A novel hybrid strategy that offers extraordinarily high levels of data security and confidentiality is being developed to tackle. The hybrid method based on the combination of ECC and Blowfish was implemented. When comparing the suggested method to the standard hybrid method, the benefits to patient privacy and safety become clear. The weaknesses of symmetric and asymmetric encryption are balanced out by using hybrid cryptography.

Bermani et al. (2021) [9] described Because of the value of the information used and the expanding use of technology, which is currently used to supply a wide range of services in different industries a lot of emphasis on cloud computing security in recent years. Therefore, cloud storage raises serious concerns about data security. Cryptographic algorithms are an essential tool for protecting data in the cloud. Information is encrypted using a combination of the Message-Digest algorithm version 5 (MD5), Blowfish, and the data protection architecture presented the Advanced Encryption Standard (AES).

Orobosade et al. (2020) [4] examined the exponential expansion of sensitive data stored on cloud platforms has significantly increased its susceptibility to security breaches. Thus, undoubtedly, the vulnerability stems from the rising number of users whose motives are harmful. The cloud is overseen by a third party, it is crucial to prioritize the provision of cloud security services. This is particularly important considering that cloud data and services are widely present in data centres. The user's increased reliance on cloud computing for various reasons necessitates the need for highly secure and protected data. This study suggests a hybrid encryption method that combines symmetric and asymmetric cryptography techniques to ensure users' privacy and security in the cloud. The system is designed to provide a secure environment. The proposal involves implementing a privacy model that utilizes the Advanced Encryption Standard (AES) as the initial level of data encryption before storing the data in a cloud application. Additionally, we employ Elliptic Curve Cryptography (ECC) as a subsequent encryption scheme, using the AES key, to ensure both data confidentiality and security in the cloud. The proposed model used AES algorithm with its key encryption using ECC, leveraging its feature as a fast-symmetric scheme and less computationally complex robust cryptosystem algorithms respectively.

Sharma et al. (2020) [22] examined the cloud is a revolutionary technology that offers highly convenient, dynamic virtualized asset pools. Security issues with data security, protection, secrecy, and verification exist since distributed computing depends on the internet. To eliminate these, experts employ a wide range of methods and encryption algorithms. Similarly, we strengthened the protection of cloud-based data by combining hybrid encryption with cross-breed cryptographic computations. In this study plan to thoroughly investigate the potential incorporation of several encryption algorithms in a hybrid computation that can better secure data in the cloud.

Poduval et al. (2020) [6] conducted that cloud computing enables the provision of a cost-effective utility to both people and companies. It enhances the capabilities of organizations by providing these services over the internet. Documents may be distributed via cloud storage. These files may contain confidential data that must be safeguarded from unidentified users. This is accomplished via the use of cryptographic techniques. Hybrid cryptography may be used to encrypt the data, ensuring a robust degree of security. The symmetric key encryption techniques used for securing data include Advanced Encryption Standard (AES) and Triple Data Encryption Standard (TDES). RSA, an asymmetric key encryption method, facilitates a hybrid cryptography approach. The security of the produced key may be further strengthened by using the picture steganography technique known as Least Significant Bit (LSB). This paper will discuss the concerns pertaining to security and its associated obstacles, as well as analyze the strategies to effectively manage them. This technique helps in achieving higher efficiency and better security due to the use of multiple algorithms for the encryption/decryption process.

Ageed et al. (2020) [13] investigated Cloud computing is one of the most significant topics, on which many academics rely by using various algorithms and techniques. Some of these methods were used to manage massive amounts of data and scheduling concerns, while others were implemented to boost performance and speed and make the most of task-level parallelism. The quantity of computation performed during implementation is affected by a number of additional parameters, memory capacity chief among them. The ability to analyze data in parallel is one of the most popular uses of infrastructure in the cloud. The scheduling principles provide a simple method of utilizing resources and processing data in parallel, hence reducing the implementation time of processing algorithms due to the inconsistent outcomes and approaches. In this study opens up novel possibilities for implementing the right technique in the area of parallel data processing. Our findings reveal the best strategies according to a number of criteria.

Anjana & Singh (2019) [26] provided a thorough qualitative analysis of security concerns in cloud computing, focusing on the three service models – SaaS, PaaS, and IaaS. The article rightly recognizes security as a significant obstacle to the broader adoption of cloud computing, especially with the outsourcing of services from third parties. The strength of this paper lies in its structured examination of vulnerabilities and threats associated with each service model. The qualitative approach provides a holistic understanding of the security landscape, enabling readers to grasp the nuances of risks inherent in cloud computing. The proposal of countermeasures in the concluding section adds practical value to the research, offering potential solutions to mitigate identified security risks. By addressing security concerns specific to SaaS, PaaS, and IaaS, the paper caters to a broad audience interested in diverse cloud computing service models. However, the benefit from a more detailed exploration of recent advancements in security technologies and their applicability to the identified vulnerabilities. Additionally, a discussion on the evolving nature of cyber threats and how the proposed countermeasures align with emerging challenges would enhance the relevance of the research.

Rashid et al. (2019) [12] described panel will focus on the two hottest topics in this sector, distributed parallel processing and distributed cloud computing. Several topics have been explored in this research, with special attention paid to the question of whether or not similar topics have been explored in other studies at the same time. Distributed parallel computing and distributed cloud computing simulations were used to test the methods. Shifting computing between servers is a necessary part of optimizing processing activities across resources. These helps improve system efficiency at just the right speeds.

Al-gohany et al. (2019)[29] studied that security is often regarded as one of the most crucial factors in everyday computing. Security is crucial in cloud computing, particularly for safeguarding sensitive and valuable data that is accessible to many users. Regrettably, the rise in cloud users has coincided with an escalation in malicious activities inside the cloud, resulting in compromised data integrity. Consequently, cloud computing security has emerged as a significant concern in the realm of cloud data. Researchers have shown a significant interest in the risks posed by malicious activities in the cloud and the potential for cloud service failures. In this work, we compare state-of-the-art methods aimed at addressing these challenges. This paper examines and compares the performance of the Data Encryption Standard (DES) and the Advanced Encryption Standard (AES) with respect to varying input sizes. The results indicate that AES is faster than DES in terms of encryption time for input sizes ranging from 20KB to 100KB. However, for input sizes beyond 100KB, DES demonstrates a significant increase in decryption time compared to AES, which only experiences a slight increase. Consequently, AES is faster than DES in terms of decryption time for input sizes ranging from 120KB to 300KB.

Wankhade et al. (2014) [30] examined this concept is currently prevalent in consumer applications such as email and picture sharing, as well as in certain commercial applications. This paper outlines a method for safeguarding data by using various compression and encryption techniques, while also concealing its storage and retrieval location from users. Similar to the Internet, on-demand apps have become so widespread that almost every business user engages with at least one, such as an email service, a web conferencing program, or a file hosting system. The data is distributed across several locations within the information space (i.e., the Internet). It resembles file hosting platforms that keep submitted material from many users and may be accessed via appropriate authentication. The only distinction lies in the fact that the system in which the document is given is an application-based system, specifically designed to operate on the client's own system. This program enables users to securely upload files of various types, using encryption and compression technologies. The submitted files are universally accessible using the given application. And posit that this approach functions as a fundamental basis for forthcoming endeavors in the integration and fortification of information sources across the World Wide Web.

Table 1. Comparison Table of Literature Review

Author year	Main Focus	Techniques	Key Findings
Sudharson [18] (2022)	Cloud information security	Client End-generated Encrypted Keycode (CEEK), RSA-based methods	Explored a strong framework for decentralized storage that enhances security in distributed computing settings
Chinnasamy [17] 2021	Cloud security and data protection	Hybrid algorithm combining ECC with Blowfish	improved security and confidentiality of patient data compared to existing hybrid methods through the use of hybrid cryptography
Bermani [9] 2021	Cloud computing security and data encryption	Hybrid cryptographic method using MD5, Blowfish, and AES	Quick and reliable data encryption through the proposed model using hybrid cryptographic algorithms
Orobosade [4] 2020	Privacy and security in cloud	Hybrid encryption method.	The suggested model uses AES with ECC key encryption, a fast-symmetric method and less computationally demanding resilient cryptosystem techniques.
Sharma [22] 2020	Data security and encryption in cloud	Hybrid ciphering with hybrid cryptographic	The security of data files kept in the cloud is increased by the use of hybrid

	computing	computations	encryption techniques.
Poduval [6] 2020	Enhancing security in cloud computing	AES and TDES	This technique helps in achieving higher efficiency and better security due to the use of multiple algorithms for the encryption/decryption process
Ageed [13] 2020	Cloud computing and parallel data processing	Techniques for dealing with huge data and improving performance	Analysis of various techniques and factors impacting parallel data processing in cloud computing
Rashid [12] 2019	Distributed parallel processing and distributed cloud computing	Simulated methods for optimizing computation in cloud computing	Investigation of distributed parallel processing and its impact on system efficiency and optimization

3. Research Methodology

The proposed methodology for securing a plain text file involves the generation of a symmetric key using the AES+ChaCha20 hybrid encryption method as shown in figure 3. This involves using both the Advanced Encryption Standard and the ChaCha20 algorithm to produce a highly secure key for encrypting the file. Once the symmetric key has been created, the plain text file is encrypted to create a cipher text file. The encrypted text file is then hashed using the SHA3 algorithm to produce a message digest, which may be used to verify the integrity of the file. The resulting encrypted file, which includes the message digest and cipher text, is safely sent to the cloud storage site in order to upload the encrypted data file there. Once the file has been stored on the cloud, it can be accessed for decryption. The file decryption process begins by first verifying the message digest using the SHA3 algorithm. If the message digest matches the one that was originally generated during the encryption process, then the file can be assumed to be intact and unaltered. The original plain text file is then recovered by decrypting the cipher text with the same symmetric key that was used to encrypt the original file.

The proposed methodology for securing a plain text file involves a multi-step process that includes generating a highly secure symmetric key using the AES+ChaCha20 hybrid encryption method, encrypting the plain text file, hashing the resulting cipher text file using the SHA3 algorithm to produce a message digest, uploading the resulting encrypted data file to the cloud, and decrypting the file by verifying the message digest and using the symmetric key to recover the original plain text file. This method provides a high level of security for sensitive data and guarantees the file's integrity throughout the encryption, storage, and decryption processes. The suggested Secure Cloud architecture will provide entrepreneurs with a secure and flexible solution for cloud computing environments. The new framework will address the limitations of existing security techniques used in cloud computing. The performance comparison of Secure Cloud with existing techniques will provide insights into the efficiency of the proposed structure.

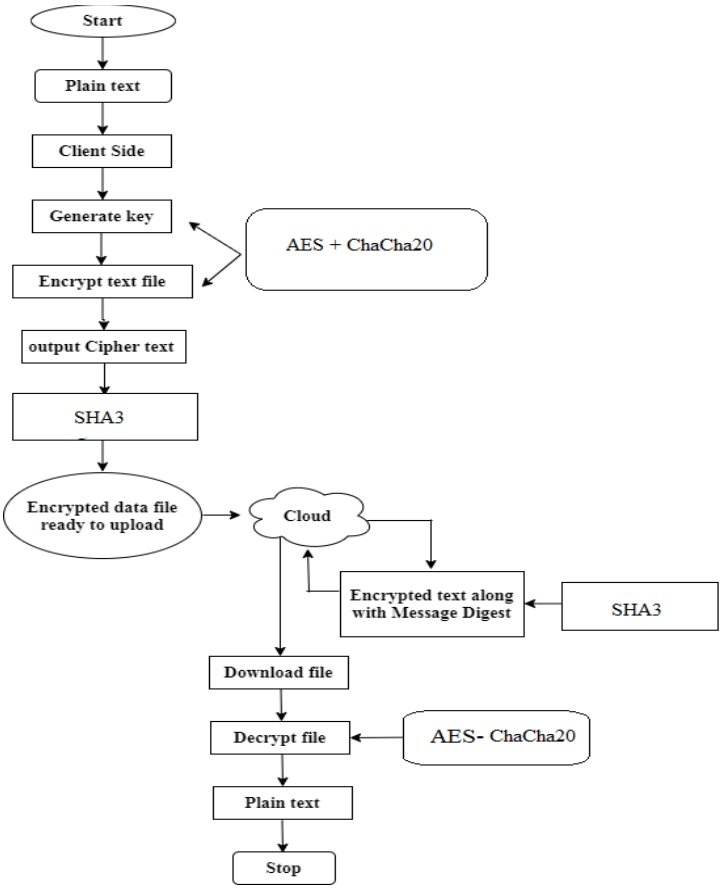

Fig 3. Flowchart of proposed methodology

3.1 AES:

One of these techniques for symmetric block encryption is the Advanced Encryption Standard [31]. In December 2001, the National Institute of Standards and Technology presented it. Each 128-bit block of plaintext is encrypted using a new key value for cycles 10, 12, and 14, which can be 128 bits, 16 bytes, 192 bytes, or 256 bits. Four segments are created from 128 bits of plain text using the Advanced Encryption Standard. These bits are grouped into the state, a 44 by 44 matrix, and are known as an array of bytes [32]. AES increases security by using four different transformations for each round of encryption for each block of 128 bits of plain text.

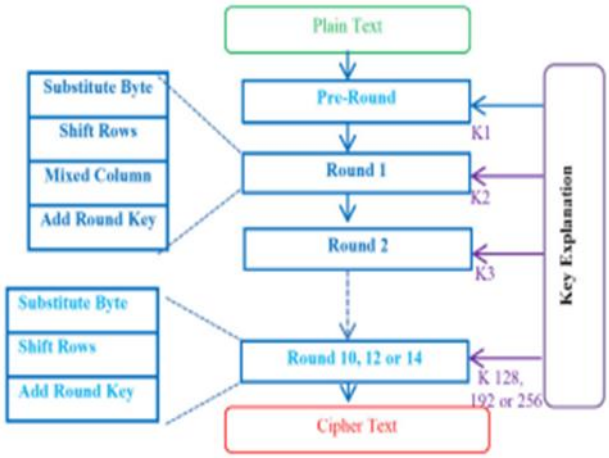

Fig 4. Encryption in Advanced Encryption Standard.

1. Substitution bytes (Sub Bytes): Since the Advanced Encryption Standard technique employs 128-bit data blocks, each block of data is 16 bytes long. Using a Rijndael S-box, an 8-bit (Byte) substitution box, each 8-bit (Byte) in a data block undergoes sub byte translation into another block.

2. Permutation (Rearrange Rows): Each of the four rows of the matrix is rotated to the left. This procedure yields a matrix consisting of 16 bytes.

3. MxCloumns: This is a straightforward substitution operation. The (finite GaloisField-GF (28)) matrix multiplication is used to change each matrix column. Following this procedure, a new matrix with sixteen additional bytes will result.

4. AddRoundKey: Here, the round key matrix and the state are combined using the XOR algorithm.

Each iteration includes a new set of these four steps, with the total number of iterations ranging from 10 to 14 for keys of 128, 192, and 256 bits in length. A diagram of the Advanced Encryption Standard encryption technique is shown in Figure 4. When it comes to protecting sensitive information, the AES algorithm is consistently recommended as a top choice. There are defining features of the AES algorithm: It's the quickest implementation algorithm and uses less memory while still being flexible and scalable.

3.2 ChaCha20

The proposed model builds upon the ChaCha20 symmetric encryption algorithm, a widely recognized method for safeguarding data confidentiality and integrity through its stream cipher architecture, which operates on individual bits or bytes of data [33][34].

1. Key and Nonce Setup: To initiate the encryption process, a 256-bit secret key (32 bytes) and a 32-bit nonce (8 bytes) are selected. Simultaneously, the block counter is set to 0, establishing the foundational parameters for the subsequent cryptographic operations.

2. Initialization: The initialization phase involves defining a 16-byte constant known as the "ChaCha constant." The 32-byte key and 8-byte nonce are then expanded into a 64-byte block termed the "ChaCha state." This expansion follows a specific structure, allocating the initial 16 bytes for the ChaCha constant, the next 32 bytes for the key, and the subsequent 8 bytes for both the block counter and the nonce.

3. ChaCha20 Core Function: The core function of ChaCha20 unfolds through a series of iterations, typically 20 rounds. Within each round, the algorithm undergoes a Quarter Round, involving operations on four 32-bit words in the state. Additionally, row and column mixing operations permute the words within the state, contributing to the overall security of the encryption [35].

4. Generating the Keystream: Upon completing the specified rounds, the ChaCha state reaches its final configuration. The first 64 bytes of this final state serve as the keystream, a crucial element in subsequent encryption and decryption processes.

5. Encryption: For the encryption process, the plaintext is divided into 64-byte blocks. Each block undergoes an XOR operation with the corresponding 64-byte segment of the previously generated keystream. In cases where the plaintext size is not a multiple of 64 bytes, the remaining bytes are XORed with the corresponding remaining keystream bytes.

6. Decryption: Decryption is achieved by XORing the ciphertext with the keystream, effectively reversing the encryption process and regenerating the original plaintext.

However, it is imperative to note that to maintain security in each encryption instance, the block counter and nonce must be unique. This methodology offers a comprehensive overview of the fundamental operations of the ChaCha20 algorithm, elucidating its key components and processes.

3.3 SHA 3

The SHA-3 algorithm, also known as Secure Hash Algorithm 3, is a cryptographic hash function designed to generate fixed-size hash values from input data. Additionally, SHA-3 offers flexibility in output sizes, allowing users to generate hash values of different lengths based on their specific requirements. Widely adopted in various cryptographic protocols and applications, SHA-3 is renowned for its efficiency, performance, and robustness, making it a cornerstone in modern cryptographic systems for ensuring data integrity and authenticity. SHA-3 is a significant cryptographic algorithm used to enhance information security and ensure the integrity of data in digital transactions [36]. Recent cryptographic hash functions such as MD5, RIPEMD, SHA-0, SHA-1, and SHA-2 have been shown to be vulnerable to attacks [37]. SHA-3 is a specific part of the Keccak family that has been officially specified by the NIST [38]. SHA-3 is defined by the standard, which includes four particular implementations of SHA-3 and two extendable-output functions, namely SHAKE128 and SHAKE256. The SHA-3 functions have a predetermined output length, but the two SHAKE variations allow for the extraction of output data of varying lengths. This feature makes SHA-3 a desirable choice for generating pseudo-random bits [39]. The SHA-3 functions all function inside a same basic structure referred to as the sponge architecture, as seen in Figure 4a. This framework is very versatile and enables the production of hash values with varying lengths, making it ideal for a wide range of applications.

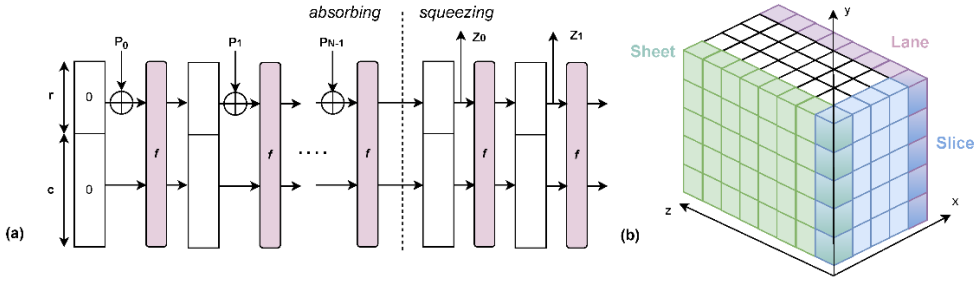

Fig 4. (a) Sponge function (b) Keccak state

4. Result and Discussion

The table 1 shows a comparative analysis of encryption and decryption performance for different file sizes. The table's data reflects the varying encryption and decryption times for different file sizes, which can be crucial for selecting an encryption algorithm based on specific performance requirements. The security rates for the proposed model and Triple DES also indicate the effectiveness of the encryption algorithms in ensuring data security. In summary, the table presents a detailed performance analysis of encryption and decryption for different file sizes, while the search results offer additional insights into the performance comparison of various data encryption algorithms, which can be valuable for making informed decisions regarding data security.

Table 1: Comparative Analysis of Encryption and Decryption Performance

File Size (Bytes)	Encryption Time (Seconds)	Plaintext Size	Decryption Time (Seconds)	Time (s)	Security Rate (Proposed Model)	Security Rate (Triple DES)
10^4	0.004	10^4	0.004	0	1.00E+00	1.00E+00
10^5	0.04	10^5	0.04	10	1.00E+00	9.00E-01
10^6	0.4	10^6	0.4	20	1.00E+00	8.00E-01
10^7	4	10^7	4	30	1.00E+00	7.00E-01
10^8	40	10^8	40	T (40)	1.00E+00	6.00E-01
-	-	-	-	50	1.00E+00	5.00E-01
-	-	-	-	60	1.00E+00	4.00E-01

In Figure 7, the graph illustrates a decline in the security rate of both algorithms over time, indicative of the increasing susceptibility of encryption algorithms as computational power advances. Notably, the Proposed algorithm consistently maintains a higher security rate than Triple DES for the majority of the depicted time frame, suggesting enhanced resilience against decryption attempts. However, a sudden drop in the security rate of the Proposed algorithm around 40 seconds raises questions about potential weaknesses. In contrast, Triple DES security rate stabilizes after approximately 20 seconds, signifying a slower deterioration in its resistance. Overall, the graph underscores the superiority of the Proposed algorithm in terms of security, emphasizing its robustness compared to Triple DES. Nevertheless, it remains essential to acknowledge that given sufficient time and resources, any encryption algorithm's security can be compromised. In Figure 5, the encryption time of Triple DES, and the Proposed algorithm is portrayed as a function of increasing plaintext size. All three algorithms exhibit an augmentation in encryption time as the plaintext size expands, aligning with the increased data processing demands. Notably, Triple DES demonstrates a linear increase and the Proposed algorithm exhibits a sub-logarithmic trend, showcasing its superior efficiency. The Proposed algorithm outpaces Triple DES for all plaintext sizes due to its stream cipher nature, encrypting data byte by byte, offering a substantial speed advantage over block ciphers. Additional insights into the proposed algorithm, factors influencing encryption time, and a comparative analysis of stream and block ciphers are available upon request. Figure 6 focuses on the decryption time of the Proposed algorithm in comparison

to Triple DES across varying plaintext sizes from 10^4 bytes to 10^8 bytes. Key takeaways include AES+ChaCha20's consistent superiority in speed over Triple DES across all file sizes. While both algorithms experience an increase in decryption time with larger files, the rate of increase diverges significantly. Triple DES exhibits a near-linear growth, while AES+ChaCha20's curve suggests a sub-linear increase, indicating a slower pace relative to file size expansion. Notably, AES+ChaCha20's decryption time appears to approach a constant value as the plaintext size becomes larger, implying efficiency in handling very large files. Additional nuances, such as the logarithmic y-axis, linear x-axis, and potential behavior deviations for larger or smaller file sizes, are noteworthy considerations.

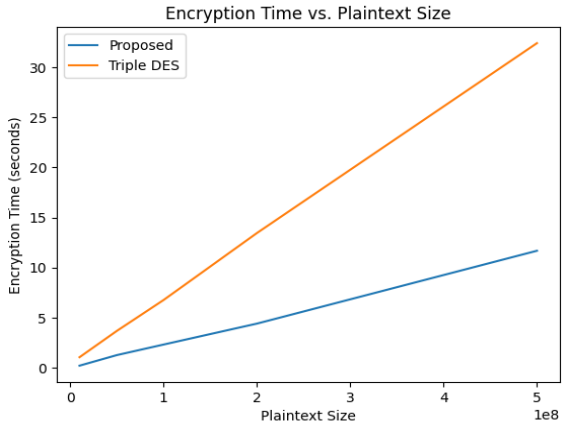

Fig 5: Encryption execution proposed technique.

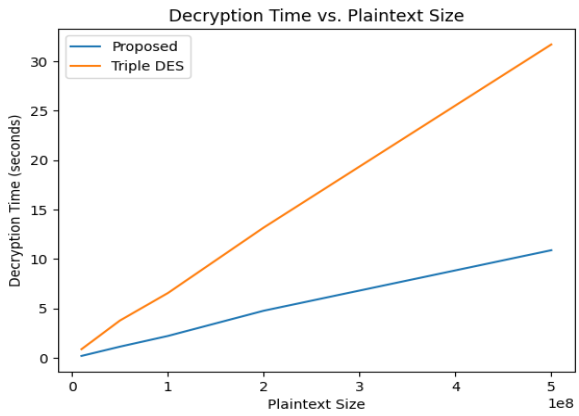

Fig 6: Decryption execution proposed technique.

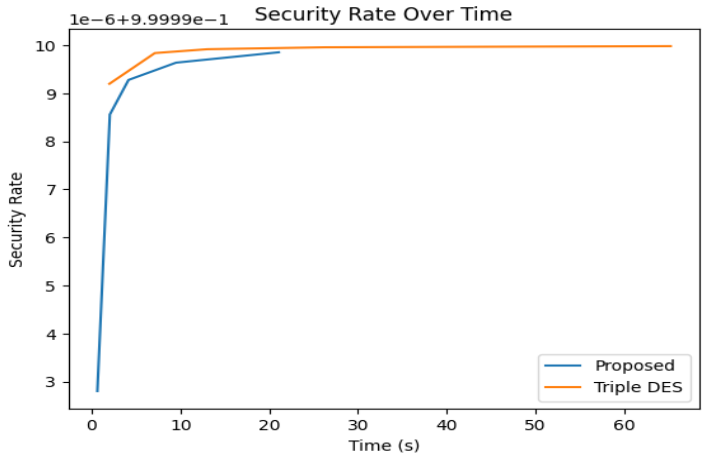

Fig 7: Security of rate proposed technique.

5. Conclusion

In conclusion, this study delves into the design and analysis of effective techniques for information security in cloud computing, a pervasive system used for on-demand data storage. Despite the prevalent utilization of cloud computing, concerns persist regarding data protection, privacy, access control, and confidentiality. This research specifically focuses on encryption strategies employed to safeguard sensitive data stored in the cloud. Recognizing the evolving landscape of cloud security challenges, the study introduces a novel hybrid technique aimed at enhancing the security and confidentiality of sensitive information. Cryptography emerges as a pivotal means to fortify cloud data security. The proposed model demonstrates efficiency and security in data encryption, offering a robust approach to address the expanding concerns associated with storing critical and sensitive data in the cloud. Furthermore, the graphical analyses presented in Figures 7, 8, and 9 provide valuable insights into the performance and security aspects of encryption algorithms. Notably, the Proposed algorithm exhibits superior security rates compared to existing technique, highlighting its resilience against decryption attempts. While a momentary drop in the security rate prompts further investigation into potential weaknesses, the overall findings emphasize the robustness of the Proposed

algorithm. The study also examines encryption and decryption times in relation to plaintext size, revealing distinctive characteristics of Triple DES, and the Proposed algorithm.

In summary, this research contributes to the ongoing discourse on information security in cloud computing by proposing a hybrid cryptographic algorithm and conducting a comprehensive analysis of encryption and decryption performance. The findings not only underscore the significance of robust encryption techniques but also provide valuable insights for optimizing the security and efficiency of cloud storage systems in the face of evolving threats and computational advancements.

Refrences

[1] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing.

[2] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., ... & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, *53*(4), 50-58.

[3] Sadeeq, M. A., Zeebaree, S. R., Qashi, R., Ahmed, S. H., & Jacksi, K. (2018, October). Internet of Things security: a survey. In *2018 International Conference on Advanced Science and Engineering (ICOASE)* (pp. 162-166). IEEE.

[4] Orobosade, A., Favour-Bethy, T. A., Kayode, A. B., & Gabriel, A. J. (2020). Cloud application security using hybrid encryption. *Communications on Applied Electronics*, *7*(33), 25-31.

[5] Kaur, R., & Singh, R. P. (2014, September). Enhanced cloud computing security and integrity verification via novel encryption techniques. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 1227-1233). IEEE.

[6] Poduval, V., Koul, A., Rebello, D., Bhat, K., & Wahul, R. M. (2020). Cloud based secure storage of files using hybrid cryptography and image steganography. *International Journal of Recent Technology and Engineering (IJRTE)*, *8*(6), 665-667.

[7] Salih, A. A., Zeebaree, S. R., Abdulraheem, A. S., Zebari, R. R., Sadeeq, M. A., & Ahmed, O. M. (2020). Evolution of mobile wireless communication to 5G revolution. Technology Reports of Kansai University, 62(5), 2139-2151.

[8] Liu, F., Tong, J., Mao, J., Bohn, R., Messina, J., Badger, L., & Leaf, D. (2011). NIST cloud computing reference architecture. *NIST special publication*, *500*(2011), 1-28.

[9] Cloud, H. (2011). The nist definition of cloud computing. *National Institute of Science and Technology, Special Publication*, *800*(2011), 145.

[10] Rimal, B. P., Choi, E., & Lumb, I. (2009, August). A taxonomy and survey of cloud computing systems. In *2009 fifth international joint conference on INC, IMS and IDC* (pp. 44-51). Ieee.

[11] Tsai, W. T., Sun, X., & Balasooriya, J. (2010, April). Service-oriented cloud computing architecture. In *2010 seventh international conference on information technology: new generations* (pp. 684-689). IEEE.

[12] Wang, C., Wang, Q., Ren, K., Cao, N., & Lou, W. (2011). Toward secure and dependable storage services in cloud computing. *IEEE transactions on Services Computing*, *5*(2), 220-232.

[13] Son, S., Jung, G., & Jun, S. C. (2013). An SLA-based cloud computing that facilitates resource allocation in the distributed data centers of a cloud provider. *The Journal of Supercomputing*, *64*, 606-637.

[14] Wei, G., Vasilakos, A. V., Zheng, Y., & Xiong, N. (2010). A game-theoretic method of fair resource allocation for cloud computing services. *The journal of supercomputing*, *54*, 252-269.

[15] Abbas, M. S., Mahdi, S. S., & Hussien, S. A. (2020, April). Security improvement of cloud data using hybrid cryptography and steganography. In 2020 international conference on computer science and software engineering (CSASE) (pp. 123-127). IEEE.

[16] Rashid, A., & Chaturvedi, A. (2019). Cloud computing characteristics and services: a brief review. *International Journal of Computer Sciences and Engineering*, *7*(2), 421-426.

[17] Chinnasamy, P., Padmavathi, S., Swathy, R., & Rakesh, S. (2021). Efficient data security using hybrid cryptography on cloud computing. In Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020 (pp. 537-547). Springer Singapore.

[18] Sudharson, K., Akshaya, M., Lokeswari, M., & Gopika, K. (2022, March). Secure Authentication scheme using CEEK technique for Trusted Environment. In *2022 International Mobile and Embedded Technology Conference (MECON)* (pp. 66-71). IEEE.

[19] Bermani, A. K., Murshedi, T. A., & Abod, Z. A. (2021). A hybrid cryptography technique for data storage on cloud computing. *Journal of Discrete Mathematical Sciences and Cryptography*, *24*(6), 1613-1624.

[20] AbdElnapi, N. M., Omara, F. A., & Omran, N. F. (2016). A hybrid hashing security algorithm for

data storage on cloud computing. *International Journal of Computer Science and Information Security (IJCSIS)*, *14*(4).

[21] Jassem, Y. H., & Abdullah, A. A. (2020). Enhancement of quantum key distribution protocol for data security in cloud environment. *Icic International*, *11*(3), 279-288.

[22] Sharma, S., Singla, K., Rathee, G., & Saini, H. (2020). A hybrid cryptographic technique for file storage mechanism over cloud. In *First International Conference on Sustainable Technologies for Computational Intelligence: Proceedings of ICTSCI 2019* (pp. 241-256). Springer Singapore.

[23] kadhim Bermani, A., Manaa, M. E., & Al-Salih, A. (2020). Efficient cryptography techniques for image encryption in cloud storage. Periodicals of Engineering and Natural Sciences, 8(3), 1359-1373.

[24] Sadeeq, M. M., Abdulkareem, N. M., Zeebaree, S. R., Ahmed, D. M., Sami, A. S., & Zebari, R. R. (2021). IoT and Cloud computing issues, challenges and opportunities: A review. *Qubahan Academic Journal*, *1*(2), 1-7.

[25] AlSufaian, R. A., AlMoussa, R. A., AlQahtani, K. H., AlGhamdi, R. A., AlAjmi, R. M., & Nagy, N. Secure File Storage on Cloud Using Hybrid Cryptography.

[26] Anjana, & Singh, A. (2019). Security concerns and countermeasures in cloud computing: a qualitative analysis. International Journal of Information Technology, 11, 683-690.

[27] Anjana, D. A. S. (2022). Hybrid Cryptographic solution using RSA, Blowfish and MD5 for Information Security in Cloud Computing. Mathematical Statistician and Engineering Applications, 71(3s), 1250-1268.

[28] Ramachandra, M. N., Srinivasa Rao, M., Lai, W. C., Parameshachari, B. D., Ananda Babu, J., & Hemalatha, K. L. (2022). An efficient and secure big data storage in cloud environment by using triple data encryption standard. *Big Data and Cognitive Computing*, *6*(4), 101

[29] Al-gohany, N. A., & Almotairi, S. (2019). Comparative study of database security in cloud computing using AES and DES encryption algorithms. *Journal of Information Security and Cybercrimes Research*, *2*(1), 102-109

[30] Wankhade, N. M., Sahare, K. A., & Bhujade, V. G. (2014). Secure cloud simulation using triple DES. *International Journal of Research in Advent Technology*, *2*(1)

[31] Abdullah, A. M. (2017). Advanced encryption standard (AES) algorithm to encrypt and decrypt data. *Cryptography and Network Security*, *16*(1), 11

[32] D'souza, F. J., & Panchal, D. (2017, May). Advanced encryption standard (AES) security enhancement using hybrid approach. In *2017 International Conference on Computing, Communication and Automation (ICCCA)* (pp. 647-652). IEEE

[33] Mahdi, M. S., Hassan, N. F., & Abdul-Majeed, G. H. (2021). An improved chacha algorithm for securing data on IoT devices. *SN Applied Sciences*, *3*(4), 429

[34] Taha, M. H., & Al-Tuwaijari, J. M. (2021). Improvement of Chacha20 Algorithm based on Tent and Chebyshev Chaotic Maps. *Iraqi Journal of Science*, 2029-2039

[35] Pfau, J., Reuter, M., Harbaum, T., Hofmann, K., & Becker, J. (2019, September). A hardware perspective on the ChaCha ciphers: Scalable Chacha8/12/20 implementations ranging from 476 slices to bitrates of 175 Gbit/s. In *2019 32nd IEEE International System-on-Chip Conference (SOCC)* (pp. 294-299). IEEE

[36] Chang, S. J., Perlner, R., Burr, W. E., Turan, M. S., Kelsey, J. M., Paul, S., & Bassham, L. E. (2012). Third-round report of the SHA-3 cryptographic hash algorithm competition. *NIST Interagency Report*, *7896*, 121

[37] Bayat-Sarmadi, S., Mozaffari-Kermani, M., & Reyhani-Masoleh, A. (2014). Efficient and concurrent reliable realization of the secure cryptographic SHA-3 algorithm. *IEEE transactions on computer-aided design of integrated circuits and systems*, *33*(7), 1105-1109

[38] Preneel, B. (2010, March). The first 30 years of cryptographic hash functions and the NIST SHA-3 competition. In *Cryptographers' track at the RSA conference* (pp. 1-14). Berlin, Heidelberg: Springer Berlin Heidelberg

[39] Ye, G., Jiao, K., & Huang, X. (2021). Quantum logistic image encryption algorithm based on SHA-3 and RSA. *Nonlinear Dynamics*, *104*, 2807-2827.