Advancing Cloud Image Security via AES Algorithm Enhancement Techniques

Zahraa A. Mohammed
Department of Computer Science, College of Science for Women, University of Babylon, Iraq
zahraa.abed@uobabylon.edu.iq (corresponding author)

Hadeel Qasem Gheni
Department of Computer Science, College of Science for Women, University of Babylon, Iraq
wsci.hadeel.qasem@uobabylon.edu.iq

Zahraa Jabbar Hussein
Department of Computer Science, College of Science for Women, University of Babylon, Iraq
zahraa.jabbar@uobabylon.edu.iq

Ali Kadhum M. Al-Qurabat
Department of Cyber Security, College of Science, Al-Mustaqbal University, Iraq
ali.kadhum.mohammed@uomus.edu.iq

Received: 7 November 2023 | Revised: 23 November 2023 | Accepted: 20 December 2023

 DOI: https://doi.org/10.48084/etasr.6601

ABSTRACT

Communication system and internet dominance in our society, has made image security a matter of paramount concern. Cryptography involves encrypting data to protect information exchange between senders and receivers, establishing a foundation for secure communication. The Advanced Encryption Standard (AES) is an exceptional algorithm that plays a pivotal role in this area because of its ability to consistently transform plain data into cipher data using the same encryption key. This algorithm engages intricate encryption techniques, harnessing a variety of algorithms and transformations to ensure robust data security. This study introduces an image encryption technique to comprehensively address security requirements. The proposed approach uses encryption to provide high reliability and security, effectively protecting sensitive media from unauthorized access. The sender's file is divided into multiple pieces to maximize confidentiality, using an advanced algorithm. Upon proper decryption, these pieces seamlessly reconstruct the original file. The suggested technique enables customers to securely keep information on cloud storage, addressing concerns about possible leakage, damage or corruption. By integrating cloud storage and digital signatures, this method ensures protection and reliability for sensitive information.

Keywords-cloud computing; cryptography; encryption; decryption; AES algorithm; digital signature; openSSL

I. INTRODUCTION

In today's interconnected world, ensuring the confidentiality and integrity of sensitive data is crucial. Whether related to business, personal affairs or other important information, there is an ongoing need to protect data from unauthorized access and misuse. Cryptography plays a key role in achieving secure communication, providing a framework to safeguard data even when transmitted over public networks where not all viewers may be trusted. As a multifaceted field, cryptography is dedicated to techniques for concealing and authenticating information stored and shared digitally, such as in cloud environments. Its core function is to transform data in a way that only approved individuals with the proper credentials can comprehend its true meaning. In an environment where data breaches and cyber incidents have become common, cryptography provides fundamental security by validating users and shielding information from improper access or alteration. Through encryption and related methods, it aims to uphold privacy, integrity and safe transfer of sensitive data online and in the cloud. As technologies continue to connect the world in new ways, responsible use of cryptography remains important for responsible data stewardship [1]. Fundamentally, cryptography consists of two steps: data concealment through encryption and data restoration for authorized users through decryption. By converting data into an incomprehensible format, encryption protects it against unauthorized access. This procedure is reversed during decryption, enabling descriptors to effectively access and utilize protected data [2-4]. Segmentation is a technique used to improve the security of

digital assets. Files are split up into smaller, encrypted components and then securely rejoined. This fortifies the barriers against unauthorized access to or alteration of private information. To verify file ownership and integrity, more methods can incorporate authentication levels. By using private/public key pairs, one may verify there was no transmission manipulation and avoid impersonation. Mismatches caused by incorrect keys indicate potential problems that should be looked into.

Cryptography is also useful in fields such as online commerce, communications, healthcare data, and government/military applications [5]. It underlies technologies that provide privacy, security, and identification, such as passwords, digital signatures, and distributed systems. As threats change with new capabilities, so does cryptography via novel research, underlining its continued importance for digital interactions and information safety in an interconnected society. Responsible implementation helps maintain security standards and data stewardship practices in many spheres of contemporary life.

A. Problem Statement

Image security, especially in cloud-based storage and communication systems, is a major problem. Malicious attackers are always present, posing a persistent danger to critical visual data security and integrity. To overcome this challenge, the scientific community has turned to cryptography for an answer. Among the different encryption methods, the Advanced Encryption Standard (AES) is notable for its ability to reliably convert plain pictures into encrypted ones while utilizing the same encryption key. To guarantee comprehensive data protection, AES utilizes advanced encryption techniques, varied algorithms, and transformations. However, there is still a need to improve and optimize AES for even greater degrees of picture privacy.

B. Motivation

The motivation behind this study stems from the pursuit of unparalleled image security (with the AES algorithm significantly contribuiting to this goal accomplishment). By intelligently fragmenting sender images and embracing cutting-edge cloud-based techniques, this study aims to maximize image confidentiality, minimize vulnerability to attacks, and empower users to safely store their data in the cloud without fear of the latter being compromised.

C. Contribution

This study makes several significant contributions to the field of cloud image security and cryptography:

1. It presents an optimized AES algorithm specifically addressing to image safety, ensuring high reliability and protection against malicious attacks.
2. It employs a novel fragmentation technique to enhance image confidentiality, making the former more sealed against breaches.
3. It introduces a cloud-based image security method, allowing ssfe data storage in the cloud, and mitigating risks of intended data damage or corruption.
4. It integrates digital signatures via OpenSSL to certify tight security and reliable verification of the recipient's identities.

II. BACKGROUND

Cryptographic algorithms play an essential role in protecting data in cloud computing environments due to the significant benefits they bring to cloud computing [6-7]. Cryptographic algorithms come in various forms, each designed to address specific security needs and challenges. Broadly, they can be categorized into two main types:

- Symmetric key encryption: One secret key is utilized for both encryption and decryption in symmetric key encryption. AES and DES are two well-known examples of symmetric key algorithms. The algorithms are highly efficient and suitable for safeguarding data at rest within the cloud.
- Asymmetric key encryption: Two keys are used in asymmetric key encryption, sometimes referred to as public-key encryption: a public key and a private key. It is only possible to decrypt data encrypted with the public key with the matching private key, and vice versa. Elliptic Curve Cryptography (ECC) and Rivest-Shamir-Adleman (RSA) are two popular asymmetric key algorithms that are useful for safe key exchange and data protection while in transit.

III. LITERATURE REVIEW

Various studies and research efforts have been carried out to explore and enhance the performance and safety of cryptographic algorithms in cloud computing environments. Several studies have investigated security in cloud computing and Internet of Things (IoT) [1, 8-12]. In [13], a proposal was presented to improve the AES algorithm to better address emerging privacy threats in the cloud. The results showed that the suggested method had increased encryption speed, reduced power consumption, better load balancing, and upgraded trust and resource management. Additional investigation suggested a lightweight encryption approach to accelerate the encryption and decryption procedures [14]. In [15], an improved Blowfish algorithm was combined with an elliptic curve-based algorithm to achieve effective data confidentiality and integrity in cloud computing. This hybrid approach enhanced security and performance, while ensuring data integrity through digital signatures. Most of the research on secure data transmission in a cloud storage environment has been focused on symmetric and asymmetric encryption techniques. Data privacy is usually assured through the use of tools like RSA and AES while other methods such as packet parsing and rule generation protect against passive data attacks [16]. Another study [17] compared hybrid AES and RSA techniques to the Blowfish/Paillier-based protocol's efficiency. The new hybrid protocol aims at boosting cloud storage by ensuring data integrity, reducing ciphertext size, improving computation time. In [18], a cloud RSA encryption system was presented, using distinct evaluation and private keys. This approach allowed third parties to perform operations on encrypted data, using the evaluation key, while the private key was only known to the data owner.

In [19], an in-depth performance assessment of various symmetric cryptographic algorithms in a cloud computing environment was carried out, including DES, 3DES, Blowfish, Twofish, RC2, RC5, RC6, and AES. The evaluation considered algorithm structure, encryption/decryption times, throughput, and memory consumption. Some of research in fully homomorphic encryption (FHE) has proposed a unique and efficient FHE scheme that can be homomorphically verified, to address the challenge of efficiently managing noise and making FHE more practical for real-world applications, especially in cloud environments [20-21]. These works attempted to overcome the difficulty of efficiently handling noise, making FHE more applicable to real-world applications, especially in cloud environments. Hybrid encryption techniques were presented in [22] in order to improve data privacy on cloud servers. costumers have the option to select their preferable encryption methods like Eclipse IDA, DSA, RSA, AES, and Blowfish, instead of relying on encryption provided by third parties. For the purpose of maximizing the security of cloud data, the current study also experimented a number of cryptographic techniques and developed a hybrid encryption architecture. In [23], the authors introduced homomorphic encryption and offered a structure for Mobile Multi-Cloud computing (MMC) analysis and evaluation. Fully Homomorphic Encryption (FHE) was utilized in their study [24] to reinforce cloud computing security and enable servers to match different customers' expectations. The principle goal of this work was to analyze the response times in relation to the size of the public key, and to facilitate the FHE method. These outcomes demonstrate the necessity to use more measures to decrease emerging risks, strengthen cloud data security, and increase the efficiency, reliability, and utilization of cryptographic algorithms.

TABLE I. SUMMARY OF LITERATURE REVIEW

Reference	Key Contributions	Algorithms Used	Notable Outcomes
Current Research	Modified AES for better speed, reduced power, and better load balancing	Enhanced AES	Improved security, reduced resource usage, lower latency
[15]	Combined enhanced Blowfish with an elliptic curve-based algorithm	Blowfish, Elliptic Curve	Improved security and performance, ensured data integrity
[16]	Comparison of symmetric and asymmetric methods; using RSA and AES	RSA, AES	Mitigated passive attacks, ensured data confidentiality
[17]	Hybrid protocol based on Blowfish and Paillier; compared with AES and RSA	Blowfish, Paillier, AES, RSA	Reduced calculation time and ciphertext size, enhanced security and integrity
[18]	Cloud (RSA) with distinct evaluation and private keys	RSA	Allowed third-party operations on encrypted data, maintained data ownership
[19]	Evaluated DES, 3DES, Blowfish, Twofish, RC2, RC5, RC6, AES	DES, 3DES, Blowfish, Twofish, RC2, RC5, RC6, AES	Considered algorithm structure, encryption/decryption times, throughput, memory consumption
[20-21]	Novel symmetrically verifiable FHE scheme	Fully Homomorphic Encryption (FHE)	Made FHE more practical for cloud applications
[22]	Hybrid encryption approaches, consumer choice in encryption methods	Blowfish, RSA, AES, Eclipse IDA, DSA	Optimized cloud data protection, various cryptographic techniques explored
[23]	Introduced homomorphic encryption for data security	Homomorphic Encryption	Evaluated homomorphic encryption in mobile cloud computing
[24]	Applied FHE to enable server operations for client requests	Fully Homomorphic Encryption (FHE)	Reduced FHE algorithm complexity, assessed response times

IV. EXISTING METHODS AND PROPOSED MODEL

A. Digital Signature

It is highly important to make a difference between digital signatures from other types of signatures, such scanned physical signatures, email headers, and letterheads, for them having various aims. Digital signatures simplifies the procedure of authenticating communications sent by customers and confirming their identity

B. One-Way Hash Functions

One-way hash algorithms are utilized to generate digital authorizations for photographs. These programs transform each message—graphics comprised—into a fixed-length message digest, typically 128 bits in length. It is usual practice in the business to use hash algorithms to securely encrypt communications. Hash algorithms are very reliable, hard to reverse engineer, and provide distinct values for every image. MD2, MD4, and MD5 are known examples of hash algorithms.

C. Message Digest Security

In order to protect against manipulation or unauthorized access, the message digest is encrypted using a public key approach such as RSA. This demonstrates that the message integrity is maintained and that any modifications are simple to detect.

D. AES Algorithm

AES is a symmetric key cryptographic technique, which means that it uses the same cryptographic key for both encryption and decryption. This type of encryption confirms that data can be safely encrypted and decrypted, using the same key. AES relies on block cipher encryption where data are encrypted by substituting, transforming, and permuting. The data are divided into states, each consisting of 128-bit elements organized in a matrix format with rows and columns serving as its keys. Each element within the matrix is called a cell. AES offers various security levels through its different variants, each employing a different number of rounds. The key sizes for AES encryption are in increments of 128, 192, and 256 bits. However, regardless of the variant, the block size and the round

key size remain fixed at 128 bits. AES excels in the following four key tasks:

1. Encryption: AES encrypts data using a symmetric key, ensuring its confidentiality during transmission or storage.
2. Decryption: The same AES key is used for decryption, allowing authorized recipients to recover the original data.
3. Substitution: AES uses substitution operations to transform the data during encryption.
4. Permutation: AES uses permutation operations to further enhance encrypted data security and complexity.

Digital signatures and AES are crucial components in signifying data authenticity, integrity, and confidentiality. Digital signatures verify the sender's identity and message integrity, while AES safeguards the data itself, providing secure communication and storage capabilities. AES performs the following four fundamental operations:

- Sub Bytes Operation: Sub Bytes Operation: Replaces An 8-bit S-box determines the unique byte for each byte in the state.
- Shift Rows Operation: Shifts the i^{th} row in the state matrix to the left by a specified number of bits.
- Mix Columns Operation: The state matrix columns are subjected to this process, employing a transformation matrix that independently alters each column.
- Add Round Key Operation: This operation is a critical step during the encrypting procedure. It involves adding a round key element to each state matrix column generated through XOR operations.

The AES encryption process begins with the plaintext encoded, followed by a series of rounds that employ the Add Round Key operation. These rounds continue until the final round, which consists of only three operations, and omits the Mix Columns step. Round keys, distinct from the cipher key used for data encryption, are required for the Add Round Key operation, and are generated through the key expansion process. Importantly, the encryption process must be reversed, except for the Add Round Key method to decrypt the data. This reversibility makes decryption the inverse of encryption.

Encryption changes a straightforward sentence into a corresponding ciphertext image by generating a particular key. Moreover, the decryption process is just the opposite of it. Thus, it can be explained as changing the same key that was used to encrypt it, returning the ciphertext image to its original legible format. Moreover, the decryption process is just the opposite of it. Thus, it can be explained as changing the same key that was used to encrypt it, returning the ciphertext image to its original legible format. Because it makes sure that only those with permission may access the information, this process is essential to data security. Image splitting and merging is another method for protecting IT data by cutting the picture into many little parts and then gathering them again. This method is very good at hindering the disclosure or modification of private information.

E. Project Flow

The first step is dissecting an original image into its component elements. Next, each component is encrypted utilizing an AES encryption algorithm by using its private key. Adhering this particular plan renders exchanging photos with other people secure, simple, and generative. To affirm their integrity within transmission and cloud server storage, these photos are further double encrypted. When the pictures come to be viewed, authorized receivers may regenerate its original state using their private keys. This method depends completely on digital signatures, which are composed of both private and public keys that are generated from images. This digital signature is produced by the sender of the image. After then, the images' receiver confirms its authenticity by checking this signature. This two-pronged method impressively promotes the genuineness and authenticity of image sharing.

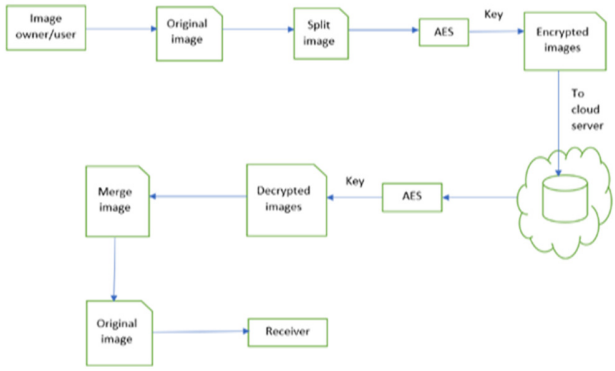

Fig. 1. Encryption flow.

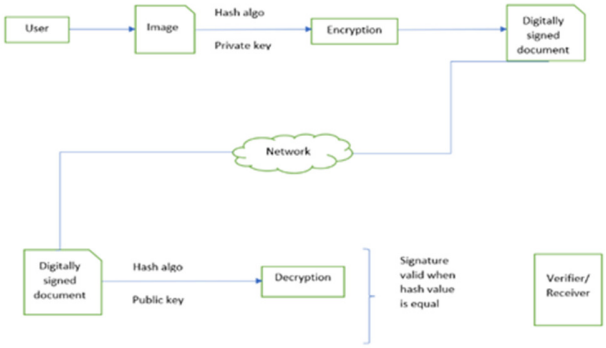

Fig. 2. Decryption flow.

F. Problem Description

Enhancing image security has been a preferred ability for cloud storage solutions. The outcome of one such endeavor was the creation of the program "Secure Image Preservation in Cloud Environments: Leveraging AES and Digital Signatures." This project's primary objective is to safeguard image data against alteration, illegal access, and even forgeries. reinforcement of the authenticity, creditability, and integrity of image data stored in the cloud is its main:

- Each of the image files (I_1, I_2) that compensate image data (I) is a digital copy of a different picture.
- Cloud Storage: Image files are preserved remotely and may be accessed by using programming application interfaces and protocols.
- Digital images are encrypted utilizing AES. Using a symmetric key K, the encryption and decryption processes are implemented simultaneously.
- Digital signature: The authenticity and integrity of image files are guaranteed by digital signatures. When making a digital signature, a personal key (*PrKey*) must be owned along with a public key (*PubKey*) to affirm that the signature is authentic.
- Encryption Process: The image data I is encrypted, using the AES algorithm and the encryption key K. The encrypted image data is represented as $Enc(I, K)$.
- Image data I employs the AES for encryption and key K. $Enc(I, K)$ is the encoding for the encrypted image data. The notation $Dec(Enc(I, K), K)$ is used to denote the decrypted image data.
- Digital Signature Generation: A digital signature is created for each image file I_i using the *PrKey*. An example of a constructed signature would be $Sign\ (I_i, PrKey)$.
- Digital Signature Verification: The validity and integrity of an image file I_i are verified by comparing its digital signature $Sign(I_i, PrKey)$ with the corresponding public key *PubKey*.

In summary, this project aims to secure image data stored in cloud storage systems by employing AES encryption for confidentiality and digital signatures for integrity and validity verification. This comprehensive approach ensures the protection and trustworthiness of image files in cloud-based storage environments.

V. ANALYSIS OF THE PROPOSED SYSTEM

The proposed system "Image Security Enhancement on Cloud Storage Using AES Algorithm and Digital Signature" offers a robust solution for addressing the security challenges associated with image data stored in cloud environments. This analysis delves into the key aspects of the system and evaluates its effectiveness in achieving the stated objectives.

- Enhanced confidentiality: One of the primary goals of the system is to improve data confidentiality. Utilizing the AES algorithm, image data are encrypted with a symmetric key before being stored in the cloud. This encryption ensures that even with unauthorized access, the data remain unreadable without the corresponding decryption key. AES, known for its strength and reliability, provides a solid foundation to maintain sensitive data privacy.
- Integrity assurance: Digital codes are utilized by the technology to verify that image files are correct. A digital signature made with a private key is tied to each picture file. This signature is like an encryption mark; it proves that the picture was not changed while being sent or stored. The public key that goes with the signing is used to make sure that the picture has not been changed. This thorough integrity check is very important to make sure that the saved pictures can be trusted.
- Verification and grant of permission. Digital signatures make sure that only people who are allowed to and have access to the secret key can make accurate signatures. This plan makes things safer by proving who the writer is and letting people who are allowed to use digital signatures do so.
- Data recovery and reconstruction: If access is granted, the technology can turn encrypted pictures back to the way they were before they were encrypted. The AES decoding method lets people who have the right encryption key get to and get back picture data. This feature lets you get back lost info while keeping it safe.
- Cloud storage integration: The system works with cloud storage services without any problems thanks to APIs and standards. This link makes sure that protected pictures are saved safely in remote cloud storage, profiting from the cloud's ability to grow and be accessed from anywhere.
- The system's interface is easy for approved users to use and understand, so picture data can be changed while still being safe. This makes it easy to get to while still meeting protection needs. The AES encryption and recovery processes are designed to be as quick and easy as possible for the system.

Fig. 3. Original image.

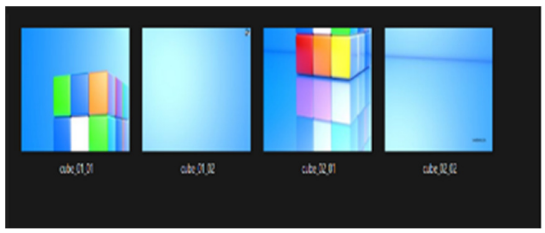

Fig. 4. Split images.

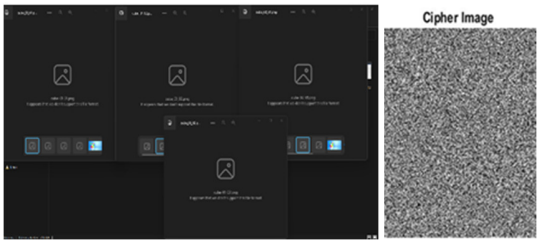

Fig. 5. Encrypted images.

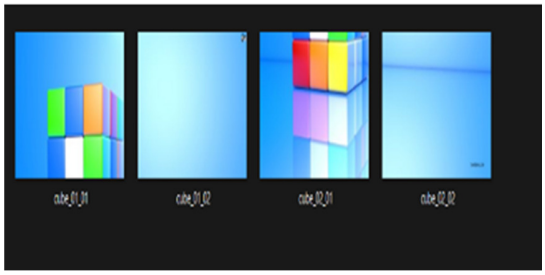

Fig. 6. Decrypted images.

Fig. 7. Merge Image (Original).

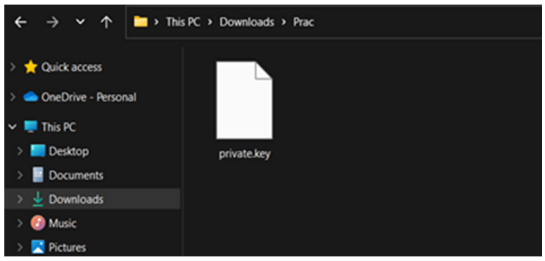

Fig. 8. Generate private key with OpenSSL.

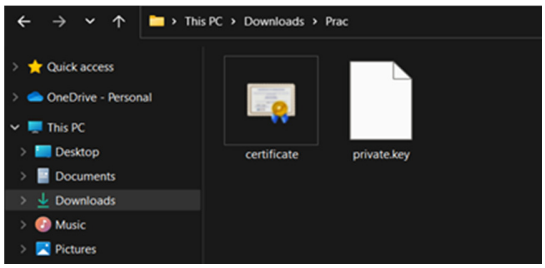

Fig. 9. Generate the digital certificate with OpenSSL.

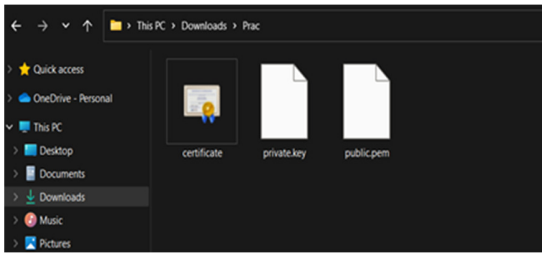

Fig. 10. Extract the public key from the certificate.

An essential aspect of the system's operation, shown in Figure 12, is the generation and validation of file digital signatures. When the outcome reads "success" after verifying a file's signature, it means the file is authentic and uncorrupted. Put simply, the file was not subject to any unauthorized changes or manipulations while it was being stored or sent. To validate the dependability of the files stored on the system, this verification step is necessary. This part of the system checks all files for modifications and uses only the ones that are still valid

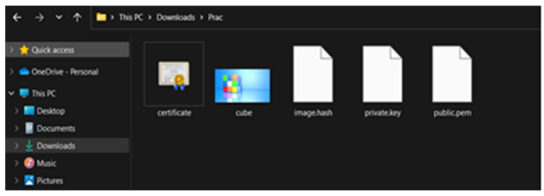

Fig. 11. Create a hash of the image.

Fig. 12. Verify with public key.

A. *Performance Evaluation*

The proposed method was evaluated on a Core i7 laptop with Windows 10 using parameters like Avalanche Test, Visual Assessment, Correlation and Statistical Analysis, Time Complexity, Execution Time, Image Histogram, and Image Entropy to compare its efficiency against traditional algorithms. Table I shows the basic measurements, and Tables II and III illustrate the results of the image entropy test and the differential cryptanalysis.

TABLE II. BASIC MEASUREMENTS

Original Image	**Size**	
	After Encryption	**After Decryption**
125 Kb	125 Kb	125 Kb
Encryption time		**Decryption time**
10 s		5 s

Surprisingly, after encryption, the picture file size stayed constant at 125 KB. This shows that the encryption mechanism used in the system does not considerably increase file size. It's advantageous because it makes it easier to store and send protected images. The file size stayed at 125 KB after decryption, the same as in the original picture. This result shows that the decryption process worked to get the picture back to its original size without missing any data or quality. It took 10 seconds to finish the encryption process, which is the same amount of time it took to change the original imagine into an encrypted version using the AES method. On the other hand, decryption was much faster; it only took 5 seconds to get the encrypted imagine back to how it was before it was encrypted. It follows that decryption is a faster process that may use fewer resources than encryption.

B. *Overall Implications*

The consistent file size before and after encryption and decryption is a positive outcome, as it certifies that the security measures do not introduce unnecessary file size overhead. The processing times for encryption and decryption are reasonable, with decryption being notably faster. This suggests that users

can efficiently access their original images with minimal delay after decryption. The system appears to strike a balance between safety and efficiency, offering users the benefit of secure image storage and transmission without significantly affecting file size or processing time. The measurement table indicates that the proposed system effectively maintains file size consistency and provides reasonable processing times for encryption and decryption operations, making it a practical solution for securing image data in cloud storage.

TABLE III. IMAGE ENTROPY TEST

No	Image	Dimensions	Entropy (ORG)	Entropy (ENC)
1	Baboon	128×128	7.2608	7.9891
		220×220	7.1662	7.9958
		256×256	7.2091	7.9973
2	Lena	128×128	7.4810	7.9885
		220×220	7.4618	7.9962
		256×256	7.4436	7.9970
3	Banda	256×256	7.5966	7.9969
		512×512	7.5217	7.9982
5	Peppers	256×256	7.5519	7.9970
		512×512	7.5555	7.9992

TABLE IV. DIFFERENTIAL CRYPTANALYSIS

Image	Size	NPCR	UACI
Baboon	256×256	99.5826	26.3210
Lena	256×256	99.5758	25.0544
Banda	256×256	99.6052	23.0526
Peppers	256×256	99.6231	31.1101

VI. CONCLUSION AND FUTURE SCOPE

Image security is significant in many different circumstances in our everyday life. Data guarding is critical, precisely when transferred over open networks like cloud services. Encryption techniques are strictly tested to improve the encryption procedure and display solid security. Moreover, the use of digital signatures enhances the privacy of encrypted images by giving permission to receivers to give credibility to the authenticity of the received data, so reinforcing the system's legitimacy. The present study proposes an image fragmentation and encryption system that combines the AES algorithm and digital signatures to safeguard cloud-stored images against malicious attacks. There is opportunity to improve the image security system by integrating sophisticated authentication procedures. Future research can focus on incorporating multifactor authentication mechanisms, like biometrics, smart cards, or single-use passwords, to improve the security of accessing and retrieving picture data from cloud storage. Furthermore, blockchain technology offers unique characteristics such as distribution, transparency, and tamper resistance, giving a new approach to image security in cloud contexts.

REFERENCES

[1] F. Thabit, O. Can, A. O. Aljahdali, G. H. Al-Gaphari, and H. A. Alkhzaimi, "Cryptography Algorithms for Enhancing IoT Security," *Internet of Things*, vol. 22, Jul. 2023, Art. no. 100759, https://doi.org/10.1016/j.iot.2023.100759.

[2] A. A. Pujari and S. S. Shinde, "Data Security using Cryptography and Steganography," *IOSR Journal of Computer Engineering*, vol. 18, no. 04, pp. 130–139, Apr. 2016, https://doi.org/10.9790/0661-180405130139.

[3] K. D. Abel, S. Misra, A. Agrawal, R. Maskeliunas, and R. Damasevicius, "Data Security Using Cryptography and Steganography Technique on the Cloud," in *Computational Intelligence in Machine Learning*, Singapore, 2022, pp. 475–481, https://doi.org/10.1007/978-981-16-8484-5_46.

[4] M. E. Saleh, A. A. Aly, and F. A. Omara, "Data Security Using Cryptography and Steganography Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 6, 2016, https://doi.org/10.14569/IJACSA.2016.070651.

[5] A. O. Aljahdali, F. Thabit, H. Aldissi, and W. Nagro, "Dynamic Keystroke Technique for a Secure Authentication System based on Deep Belief Nets," *Engineering, Technology & Applied Science Research*, vol. 13, no. 3, pp. 10906–10915, Jun. 2023, https://doi.org/10.48084/etasr.5841.

[6] Z. Yan, R. H. Deng, and V. Varadharajan, "Cryptography and data security in cloud computing," *Information Sciences*, vol. 387, pp. 53–55, May 2017, https://doi.org/10.1016/j.ins.2016.12.034.

[7] V. Agrahari, "Data security in cloud computing using cryptography algorithms," *International Journal of Scientific Development and Research (IJSDR)*, vol. 5, no. 9, pp. 258–260, 2020.

[8] F. Thabit, A. S. Alhomdy, A. H. A. Al-Ahdal, and D. S. Jagtap, "Exploration of Security Challenges in Cloud Computing: Issues," *Journal of Information and Computational Science*, vol. 10, no. 12, pp. 35–57, 2020.

[9] F. Thabit, S. Alhomdy, and S. Jagtap, "Security analysis and performance evaluation of a new lightweight cryptographic algorithm for cloud computing," *Global Transitions Proceedings*, vol. 2, no. 1, pp. 100–110, Jun. 2021, https://doi.org/10.1016/j.gltp.2021.01.014.

[10] F. Thabit, S. Alhomdy, and S. Jagtap, "A new data security algorithm for the cloud computing based on genetics techniques and logical-mathematical functions," *International Journal of Intelligent Networks*, vol. 2, pp. 18–33, Jan. 2021, https://doi.org/10.1016/j.ijin.2021.03.001.

[11] F. Thabit, O. Can, R. U. Z. Wani, M. A. Qasem, S. B. Thorat, and H. A. Alkhzaimi, "Data security techniques in cloud computing based on machine learning algorithms and cryptographic algorithms: Lightweight algorithms and genetics algorithms," *Concurrency and Computation: Practice and Experience*, vol. 35, no. 21, 2023, Art. no. e7691, https://doi.org/10.1002/cpe.7691.

[12] O. Can, F. Thabit, A. O. Aljahdali, S. Al-Homdy, and H. A. Alkhzaimi, "A Comprehensive Literature of Genetics Cryptographic Algorithms for Data Security in Cloud Computing," *Cybernetics and Systems*, 2023, https://doi.org/10.1080/01969722.2023.2175117.

[13] E. S. I. Harba, "Secure Data Encryption Through a Combination of AES, RSA and HMAC," *Engineering, Technology & Applied Science Research*, vol. 7, no. 4, pp. 1781–1785, Aug. 2017, https://doi.org/10.48084/etasr.1272.

[14] A. H. Al-Omari, "Lightweight Dynamic Crypto Algorithm for Next Internet Generation," *Engineering, Technology & Applied Science Research*, vol. 9, no. 3, pp. 4203–4208, Jun. 2019, https://doi.org/10.48084/etasr.2743.

[15] G. S. Mahmood, D. J. Huang, and B. A. Jaleel, "Achieving an Effective, Confidentiality and Integrity of Data in Cloud Computing," *nternational Journal of Network Security,* vol. 21, no. 2, pp. 326–332, https://doi.org/10.6633/IJNS.201903 21(2).17.

[16] S. M. Chavanv and S. C. Tamane, "Comparison of symmetric and asymmetric algorithms for cloud storage security," *International Journal of Advanced Science and Technology*, vol. 29, no. 3, pp. 785–91, 2020.

[17] B. Seth *et al.*, "Secure Cloud Data Storage System using Hybrid Paillier-Blowfish Algorithm," *Computers, Materials & Continua*, vol. 67, no. 1, pp. 779–798, 2021, https://doi.org/10.32604/cmc.2021.014466.

[18] K. El Makkaoui, A. Beni-Hssane, A. Ezzati, and A. El-Ansari, "Fast Cloud-RSA Scheme for Promoting Data Confidentiality in the Cloud Computing," *Procedia Computer Science*, vol. 113, pp. 33–40, Jan. 2017, https://doi.org/10.1016/j.procs.2017.08.282.

[19] E. Elgeldawi, M. Mahrous, and A. Sayed, "A Comparative Analysis of Symmetric Algorithms in Cloud Computing: A Survey," *International Journal of Computer Applications*, vol. 182, no. 48, pp. 7–16, Apr. 2019, https://doi.org/10.5120/ijca2019918726.

[20] A. El-Yahyaoui and M. D. Ech-Chrif El Kettani, "A verifiable fully homomorphic encryption scheme to secure big data in cloud computing," in *2017 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, Rabat, Morocco, Aug. 2017, pp. 1–5, https://doi.org/10.1109/WINCOM.2017.8238186.

[21] A. El-Yahyaoui and M. D. Ech-Cherif El Kettani, "A Verifiable Fully Homomorphic Encryption Scheme for Cloud Computing Security," *Technologies*, vol. 7, no. 1, Mar. 2019, Art. no. 21, https://doi.org/10.3390/technologies7010021.

[22] V. Keer, S. I. Ali, and P. N. Sharma, "Hybrid Approach of Cryptographic Algorithms in Cloud Computing," *International Journal of Emerging Technology and Advanced Engineering*, vol. 6, no. 7, pp. 87–90, 2016.

[23] M. Louk and H. Lim, "Homomorphic encryption in mobile multi cloud computing," in *2015 International Conference on Information Networking (ICOIN)*, Cambodia, Jan. 2015, pp. 493–497, https://doi.org/10.1109/ICOIN.2015.7057954.

[24] S. Bajpai and P. Srivastava, "A fully homomorphic encryption Implementation on cloud computing," *International Journal of Information & Computation Technology*, vol. 4, no. 8, pp. 811–816, 2014.