Journal of Library and Information Studies 21:2 (December 2023) pp.1-26
https://doi.org/10.6182/jlis.202312_21(2).001

Analysis of a Double-stage Encryption Scheme Using Hybrid Cryptography to Enhance Data Security in Cloud Computing Systems

Moses Kazeem Abiodun[1], Agbotiname Lucky Imoize[2,3], Joseph Bamidele Awotunde[4], Cheng-Chi Lee[5,6,7], Abidemi Emmanuel Adeniyi[8], Ugbaja Chioma[9], Chun-Ta Li[10]

Abstract

Recently, network users have been scared of storing sensitive information, such as bank details, health records, and other vital information, on the Internet because it is vulnerable to attack by a third party. Several threat models are impacting the security of the cloud. Having a secure cloud system will help to be at ease in using cloud computing facilities. This study aims at providing a cryptography approach to eliminating the vulnerabilities in the cloud-based system, and making access and data storage in the cloud very safe. The system uses Rivest-Shamir-Adleman (RSA) to encrypt files and the Advanced Encryption Standard (AES) key to encrypt the encrypted files. The hash function is used for extra key security, and Python programming language was used to implement the system, and for cloud storage, MongoDB was used. Generally, results indicate that the Double Stage Encryption (DSE) takes an average time for encryption of 83% and decryption of 75% compared to RSA and AES singly. The RSA is 68% faster than AES during the encryption process, but there is no significant difference between the two during decryption. The Avalanche effect testing showed the DSE to be 17% higher than singly testing AES and RSA, which implies it is more secure than RSA and AES as single encryption schemes. Therefore, the study recommends using DSE to secure valuable data on the cloud.

Keywords: Cloud Computing; Cryptography; Security and Privacy; Rivest-Shamir-Adleman; Advanced Encryption Standard

[1,9] Department of Computer Science, Landmark University, Omu-Aran, Nigeria
[2] Department of Electrical and Electronics Engineering, University of Lagos, Lagos, Nigeria
[3] Department of Electrical Engineering and Information Technology, Ruhr University, Bochum, Germany
[4] Department of Computer Science, University of Ilorin, Ilorin, Nigeria
[5] Research and Development Center for Physical Education, Health, and Information Technology, College of Education, Fu Jen Catholic University, New Taipei, Taiwan
[6] Department of Library and Information Science, Fu Jen Catholic University, New Taipei, Taiwan
[7] Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan
[8] Department of Computer Science, Precious Cornerstone University, Ibadan, Nigeria
[10] Program of Artificial Intelligence and Information Security, Fu Jen Catholic University, New Taipei, Taiwan
* Corresponding Author: Cheng-Chi Lee, E-mail: cclee@mail.fju.edu.tw

1. Introduction

In recent years, there is a growing interest in information analysis as a research theme. Various security measures ensure the safety and security of user data. Cloud computing requires strong security mechanisms to ensure that users can use the technology safely. Thanks to various security measures, enabling a unified perspective within this procedure. Thus, end-user security and privacy must be taken seriously in order to transmit critical user data and information for various decision-making needs such as image processing, medical examination, target tracking, and risk analysis (Jimoh et al., 2022). Security and privacy measurements must be taken seriously to enhance and significantly improve performance within cloud computing systems (Imoize et al., 2020).

Computing services, including software, storage, databases, apps, networking, and IT resource analytics, are key examples of cloud computing. Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) are typical examples of services accessible online (Abiodun, Adeniyi et al., 2022; Mohammend & Zebaree, 2021; Selvanayagam et al., 2018). One can easily use low-cost resources that are highly adaptable and effective thanks to cloud computing (Meshram, Ibrahim et al., 2022). Meanwhile, researchers have long been troubled by how to secure cloud computing servers from attackers and intruders. This has reduced the usability and confidentiality of the users in storing critical data and information on a cloud computing database. To this end, a hybrid cryptography mechanism should be considered in order to secure the data in a cloud computing environment.

Many researchers have inspired the need to ensure cloud computing is secure, and there are various communications protocols for developing different cloud computing models. One of the most important ways to secure cloud computing is the cryptographic algorithms that protect data and information in cloud computing databases. This can reduce the number of attackers in a cloud computing environment. However, it has not been adequately studied that using hybrid security measures to store data in databases significantly reduces intruders in cloud computing.

Numerous models have been presented to analyze and handle attacks and intruders in cloud computing, like intrusion detection, cryptography (AbdulRaheem et al., 2021; Meshram, Ibrahim et al., 2022; Selvanayagam et al., 2018), and steganography. These methods, which concentrate on various traits based on the demands of the industry, have been used worldwide.

Cryptography is a technique for protecting information from unauthorized parties (Abikoye et al., 2023). Cryptography aims to safeguard and maintain the security of information and data from unauthorized users (Meshram et al., 2023). Symmetric and asymmetric algorithms, hashing, blowfish, and elliptical curve cryptography have been reported. Examples of cryptographic methods include Data Encryption Standards (DES) and Advanced Encryption Standards (AES). These techniques are used to protect cloud file storage. In order to protect data and information on cloud computing platforms, this study suggests a hybrid cryptography model.

High scalability, confidentiality, and simple information accessibility over the Internet are all features of cloud computing. Despite the robust

security of the standard encryption scheme, the most concerning issue is the routine side-channel assault for collecting one's private and sensitive images, audio, and video. In addition to a targeted Virtual Machine (VM), a malicious VM can extract all data, and addressing this issue is the motivation for this paper.

The key contributions can be summed up as follows:

(1) The study presents a hybrid cryptographic algorithm for image encryption that uses RSA and AES for various files-updatable cryptography.

(2) The study provides a robust crypto-architecture for the highlighted model, including key management, encryption, data integrity, digital signatures, and decryption modules.

(3) The study conducted an assessment of the highlighted model using a variety of performance indicators, including temporal complexity and computational overhead.

1.1 Motivation

Given the importance of the data stored in the cloud, safekeeping is vital in the cloud computing environment. The information may be sensitive and highly attackable. As a result, data management needs to be entirely reliable (AbdulRaheem et al., 2022). Cloud data must be protected from harmful intrusions (Abiodun, Adeniyi et al., 2022; Ayo et al., 2023; Rana et al., 2022). Concerns about data availability, confidentiality, and integrity are raised, and there is a need to address these prevalent concerns. Data confidentiality is lost as a result of unauthorized access to information. Failure of cloud services compromises data availability and integrity, and reliability and security are complementary (Abiodun, Awotunde et al., 2022).

The most modern and sophisticated elliptic curve cryptography (ECC) method is called the elliptic curve (EC). EC is frequently employed to enhance the safety of public communication systems and to permit specified individuals with verified characteristics to participate in the current digital society, leveraging Multi-Cloud Data Encryption (MDE) (Ullah et al., 2023). Social networking, the cloud, and the Internet of Things sector are just a few of the technologies that MDE users apply. The entire environment must preserve the users' safety and privacy regardless of the tool they are using (Ghiasi et al., 2023).

Learning cryptography is essential since insecure networks expose information to data transmission, mugging, and hacking via an open network. As a result, understanding cryptography is crucial to data security (Kumbhare et al., 2022; Kumar et al., 2022). A digital signature, the integrity of cryptographic data, the authentication procedure, mathematical computations to determine the signature, and the sender depend on the recipient's address (Ahmed & Barukab, 2022; Imoize et al., 2020). To illustrate the differences between the two processes, the stated solution is compared to the approach currently used by Elliptic Curve Digital Signature Algorithm (ECDSA) during the signature and verification processes (Shukla et al., 2022; Ullah et al., 2023).

High-tech equipment is frequently utilized in cloud computing to store various data. Several kinds of medical images include X-rays, MRIs, and CT scans. Such images are necessary for diagnosing a variety of ailments by doctors. These images must be transmitted through various

channels, including the Internet. Their privacy will be threatened by harmful attacks when sharing photographs online.

Consequently, research into and implementation of medical data encryption has become crucial. The necessity of protecting patient privacy for medical photographs and the widespread reproduction of such photos in most medical facilities have served as strong motivators for researchers. In order to connect patients and doctors with different specializations and obtain a quick diagnosis, medical applications and solutions have been widely employed in addition to actual medical facilities. Patients cannot use such applications unless they are confident that their information privacy has been protected (Abd Elminaam et al., 2022).

The remaining sections of the paper are structured as follows: The review of related literature is presented in Section 2. The research technique is covered in Section 3. Section 4 gives details of implementations, while Section 5 presents the discussion of the results. Finally, Section 6 provides a conclusion of the study with key findings.

2. Review of Related Work

The authors discuss the rise of cloud computing and its safety anxieties, including data theft, data breaks, insecure Application Programming Interfaces (APIs), account theft, and denial of service (Meshram, Imoize et al., 2022; Meshram et al., 2021). Like Garg et al. (2017), several researchers have investigated numerous aspects of cloud security, such as basic protection, cross-site programming, SQL injection, and man-in-the-middle assaults. In Subasree and Sakthivel (2010), the authors described a method that interacted with the destination over a protected channel after encrypting the plaintext using ECC. The same plaintext was utilized simultaneously with Message-Digest algorithm 5 (MD5) to generate the hash result. This value was transmitted to the destination while encrypted with DUAL RSA. ECC and DUAL RSA are two asymmetric encryption algorithms that take advantage of extra time for encryption to achieve integrity and make it difficult for hackers to retrieve text from an encrypted file.

A hybrid method utilizing symmetric and asymmetric techniques was employed (Zhu, 2011). The key and digital signature were encrypted with ECC, a component of the AES approach, while the content was encrypted with AES. The key-dependent Advanced Encryption Standard algorithm (KAES) key, which belonged to the AES family, was only used once by the sender. In order to obtain the original data, signature verification was done at the receiver's end. This system's level of security was quite weak. A hybrid two-phase cryptography technique called two-phase hybrid cryptography algorithm (THCA) was used for wireless networks (Rizk & Alkady, 2015). By using this method, the plaintext is separated into two halves. The AES method was used to encrypt the first section; then the ECC algorithm to encrypt the key. Both of these cryptographic techniques are asymmetric. The RSA technique was used to encrypt the second half. The MD5 technique was employed to ensure that the data was secure. For decryption and retrieval of the plaintext, the previous processes were reversed.

Pavani and Trinatha (2019) proposed a routing protocol centered on clusters to secure the wireless sensor networks. A particle swarm that adapts optimization (APSO) improves the firefly algorithm during data transmission. The encrypted data was sent to the sink node within the network. The required transmission of the message was allocated into two portions. The AES encryption algorithm was utilized for the first, while the Rivest cipher 6 (RC6) algorithm was used for the second. Data integrity was also achieved using the MD5 method. The Secured Cluster-Based Routing Protocol (SCBRP) aims to minimize energy usage per node to extend the network's lifetime. The SCBRP was designed with energy-efficient clustering, safe routing, and security verification. Before data was uploaded to cloud storage, it was encrypted using various cryptography algorithms.

Bansal and Agrawal (2017) described a method for securely storing information in cloud storage. The method uses a unique key to secure cloud storage access. In order to increase security during authentication, image matching was used. The user can upload their file using the cloud database storage. The file comprising sensitive user information was divided into various blocks choosing particular bits using ECC techniques to encrypt the blocks. This help secures the data in a cloud-based storage system. Thus, the file is securely stored on the cloud. This method involved selecting the metadata from each block, encrypting it with ECC, and then appending it to the end of the file. The data in the full file is not sufficiently protected by this method. Only checking to determine if data has changed or not may be necessary.

Timothy and Santra (2017) proposed symmetric and asymmetric encryption techniques to protect cloud computing security. The Blowfish algorithm was used to encrypt the file that was uploaded to the cloud, and the RSA technique was used for the secret key generated by the Blowfish procedure. A secure hash algorithm-2 (SHA-2) generated the message digit on the encrypted file. The previously prepared message digit was then put through the digital signature process (DSA).

The authors devised and implemented a security mechanism for cloud storage (Chueh & Sun, 2017). In order to keep data securely in the cloud, the method combines the AES encryption technology with a third-party auditor (TPA). A user or a Cryptographic Service Provider (CSP) could not decrypt the encrypted file because the encryption key was kept in the TPA. This solution improved key management and increased the security of verification. The user can access the encrypted file and the encryption key through the CSP and TPA. The master key might then be obtained by decrypting the encryption key with his specified password. Ultimately, the user could unlock the encrypted file and get the original. Because the key had to be encrypted before being placed in the TPA, this method incurred increased overhead and cost complexity.

For file security, a self-encryption system's design and implementation were suggested (Han et al., 2016). Before being implemented in the cloud, the scheme was used in the text. It employed the eXclusive OR (XOR) method to partition the plaintext and encrypted text into 1024-bit chunks. This technique was considered problematic due to the risk of storing the key in the database. Therefore, Li et al. (2017) proposed a

Security-Aware Efficient Distributed Storage (SA-EDS) using three algorithms called Alternative Data Distribution (AD2). The study split up the plaintext needed to be stored on the cloud servers. The Secure Efficient Data Distributions (SED2) provided a plaintext with two distinct ciphertexts as output in the second technique. The third algorithm, Efficient Data Conflation (EDCon), was combined with two other encryption messages to enable users to retrieve text from disseminated cloud networks. After accepting the two encoded text fragments and the key, it provided the plaintext. The complexity and overhead of this method were increased by using three methods.

Rahardjo and Shidik (2017) proposed a self-encryption system for file security before uploading the text to the cloud. The XOR method separated the plaintext and encrypted text into 1024-bit pieces. It also used a database to keep track of the file's ID, which enclosed both plaintext and a key. In order to obtain the plaintext, the key was acquired from the databank during the decryption process. Loading the key in the databank was seen as a flaw in this technique since it was vulnerable to attack. The authors presented a lightweight Speck encryption technique (AbdulRaheem et al., 2022) for Smart Healthcare Systems (SHS) medical data. The suggested approach reduces the time required for encoding on the SHS framework while maintaining the swap between safety and effectiveness. The suggested structure is compared to current attempts on lightweight start-ups in regard to accuracy, memory utilization, processing time, and specificity. Outcomes demonstrate that the procedure is extremely protected, efficient, and better suited for data security in an IoT-driven edge computing environment (Tripathy et al., 2022). To ensure healthcare data's safety, privacy, confidentiality, integrity, and processing mode, it is essential to keep it safe from hackers. Ogundokun et al. (2021) proposed a New Lightweight Speck Cryptographic Algorithm to enhance the security of high-performance computing for patient information. The findings indicated a high degree of security and an apparent improvement in the time it requires to encode data and protect it is attainable compared to the cryptographic methods frequently used in cloud computing.

Similar proposals for lightweight encryption centered on the Tiny Encryption Algorithm (TEA) for an IoT-driven setting were made by AbdulRaheem et al. (2021). Instead of using hardware implementation, increase speed from a software standpoint. The proposed technique was utilized to shorten the encryption process on the IoT platform while maintaining the efficiency-security trade-off. The suggested study compared favourably to existing studies on lightweight start-ups regarding memory use, time complexity, and correctness. Results indicate that the technique is more effective and safer in an IoT-driven setup, making it better suited for data security. Chakrabarti and Suresh Babu (2021) proposed a second-stage encryption technique for interactive content protection using a pseudorandom generation method. The first step encodes audiovisual data using a symmetrical public key ciphertext-1. Next, an asymmetric private key created randomly is used to encrypt the ciphertext-1 in the cloud. Anyone who obtains the cipher text cannot decrypt the contents of the multimedia files. The proposed technique is a cloud computing security measure broadly

applicable due to its low complexity and simple implementation.

AbdulRaheem et al. (2021) suggested using a Crypto-Stegno method to protect medical data in the Internet of Medical Things (IoMT) platforms. The study tested the scheme on medical datasets and found astounding outcomes in perceptibility quality, extreme resistance to data loss, embedding capabilities, and safekeeping. The projected model was a real plan for savvy and effective medical data on the IoMT platform.

Abod et al. (2020) introduced a new technique for hiding hybrid steganography and quantum cryptography messages in graphics. The output is encrypted using quantum one-time pad encryption after the least significant bit (LSB) substitution is used to conceal hidden messages within cover images of three bands (Red, Green, and Blue). This hybrid approach is stimulated and put into practice. The models are explicitly presented and put to the test. Additionally, to find LSB steganography in images, the test analysis uses the steganalysis application StegExpose. The experimental results demonstrated that image concealment is consistently secure and untraceable, and as a result, the suggested new hybrid model offers a sufficient level of security. By using powerful, cutting-edge steganalysis tools, it was discovered that the suggested system's low payload threshold results in a high margin of communication security and safety. Although each file had the whole content of the material as embedded text, no payload files were found (0% detections).

Maitri and Verma (2016) developed a new security approach using steganography and the symmetric key cryptography algorithm. AES, blowfish, RC6, and Byte Rotation Algorithm (BRA) are employed in the study to secure data in blocks. For all techniques, the key size is 128 bits. LSB steganography is presented for key data protection. The key data defines which technique and key are to be used to encode a specific portion of the file, and the file is split into eight portions. Every component of the file is encoded using a unique algorithm. The multithreading approach is used to encrypt every component of the file simultaneously. Data encryption keys are placed using the LSB approach into the cover image. Steganography (Stego) image is sent through email to a verified recipient. The reverse encryption procedure is used to decrypt files.

Based on user demand, cloud computing provides users with infrastructure, platforms, and software as services over the internet economically. However, potential data breaches could result from storing sensitive or confidential data on cloud servers and infrastructure that customers do not control or maintain. Security is thus the primary issue that restricts the benefits of cloud computing. In order to accomplish the level of security required in the cloud, cryptography is essential. In addition, hybrid cryptography leverages the benefit of combining multiple cryptographic methods to improve overall efficiency and availability (Murad & Rahouma, 2021).

The quick rise of cloud services became outstanding due to the rapid development of cloud computing technologies. However, today's society faces difficult issues with data security. The security of the cloud and effective cloud implementation across the network are the key concerns with cloud computing. Confidentiality, authentication, accessibility, data recovery, and

data integrity are the security models used in the cloud (Sajay et al., 2019). These concerns encompassed cloud computing hurdles, cloud computing security issues, and cloud computing services. Enhancing cloud data security has become a top priority in the modern era, and using appropriate encryption methods while storing data on the cloud is the answer to this problem. The three most important factors for cloud computing are (a) data secrecy, (b) data integrity, (c) availability, and (d) confidentiality (Abiodun, Adeniyi et al., 2022; Erondu et al., 2022; Sajay et al., 2019). Hence, a suitable approach to data security and privacy on cloud database storage has been presented.

3. Materials and Methods

The cloud is a computing and data paradigm in which digital data is kept in conceptual streams. A hosting company manages the physical layout. These online backup companies are responsible for keeping the information protected, encrypted, and operating, as well as the physical ecosystem.

The encryption process:

The data is first encrypted with RSA. The RSA is an asymmetric encryption algorithm that uses private and public keys obtained using two very large prime numbers for encoding and decoding. When the file is ready to be encrypted, RSA requires the private key to be present before encryption can be done. Likewise, when a file is ready to be decrypted, RSA requires the public key that relates to the previously used private key to be present before decryption can be performed.

There are digital signatures in each pair of keys that are used to know the public key that belongs to a private key. The work of this digital signature is to certify the integrity of the user. After the encryption with RSA, the data is then encrypted with AES.

The AES is a symmetric cryptography algorithm. This means AES uses a single key for both encoding and decoding, and the key can only be generated and provided by the user. Before the encryption, a series of actions, such as substitution, shifting bytes, and rotation, are carried out. As a result, there are different bits of AES keys. These 128 bits, randomly chosen, are used in this case as a password (key) of 16 characters or less that are needed for the AES. Figure 1 shows the double encryption phases.

The encryption process is divided into 7 steps:

Step 1: The user inputs the files to be encrypted.

Step 2: The user inputs the password. The password is a prerequisite for encryption and acts as the AES 128-bit key.

Step 3: The user generates RSA keys (public and private keys)

Step 4: The file is then encrypted using RSA.

Step 5: The file already encrypted with RSA is again encrypted using the AES algorithm, hence the double encryption.

Step 6: The system checks and prompts you if any more files need to be encrypted. If there are more files, the process is started again for the new file to be encrypted. Else it stops.

Step 7: The encrypted files are uploaded to the cloud.

Figure 1. Double Encryption Phases

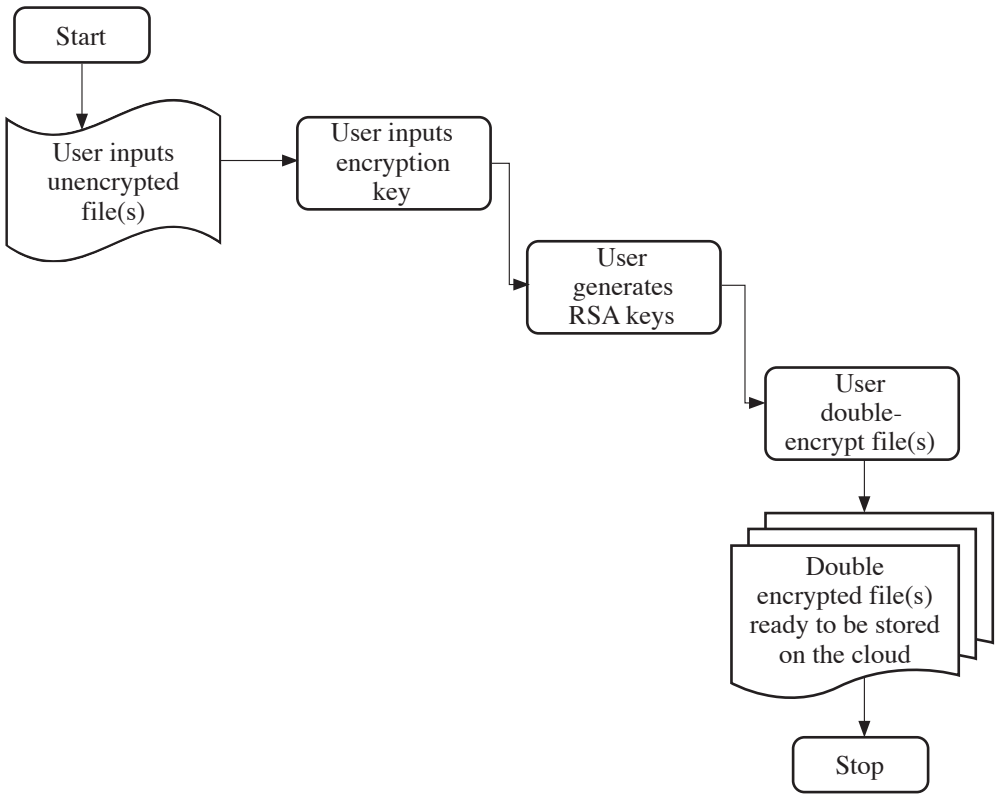


Figure 2 displays the double decryption phases for the proposed system. The steps of the decryption process are explained below:

The decryption process is divided into 6 steps:

Step 1: The user inputs the files to be decrypted.

Step 2: The user inputs the password. The password is a prerequisite for encryption and acts as the AES 128-bit key.

Step 3: System checks for the correct RSA private key. If it is correct, the decryption process continues, else the process is truncated.

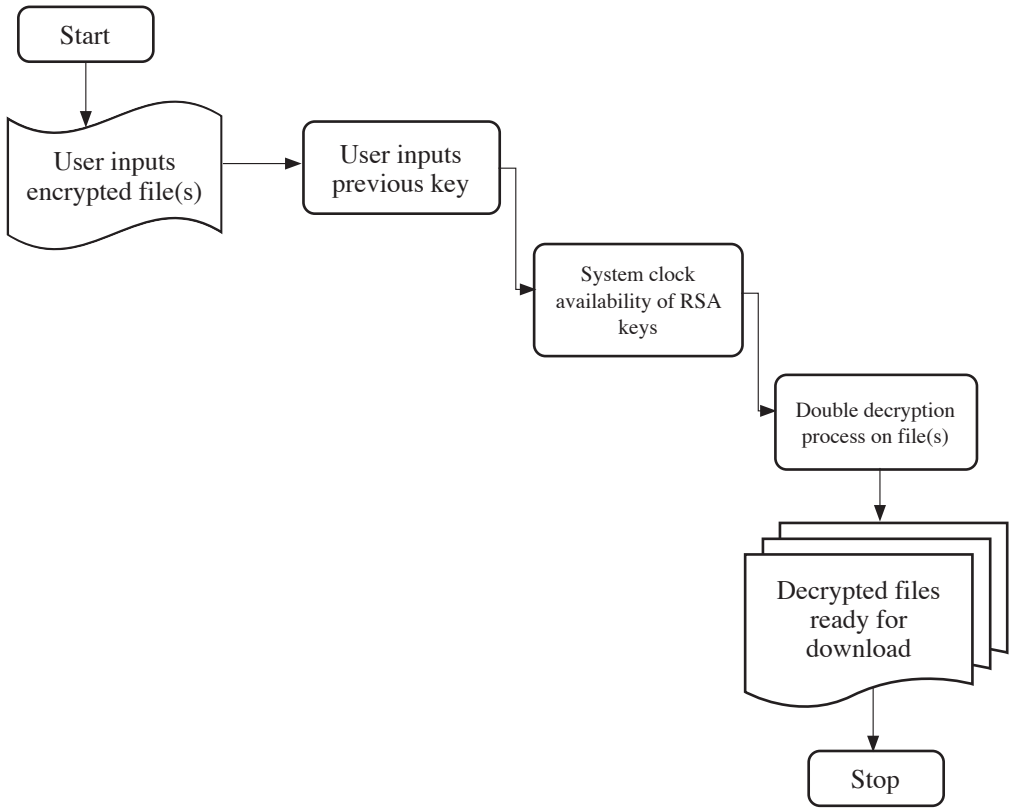
Step 4: AES decryption is first performed on the encrypted file then RSA decryption is performed.

Step 5: The system checks and prompts you if there are any more files to be decrypted. If there are more files, the process is started again for the new file to be decrypted else, it stops.

Step 6: The decrypted files are allowed by the authorized user then the files are downloaded from the cloud.

The system user interface is divided into a register, login, and file manager. First, the register section is created for new users. This would enable

Figure 2. Double Decryption Phases

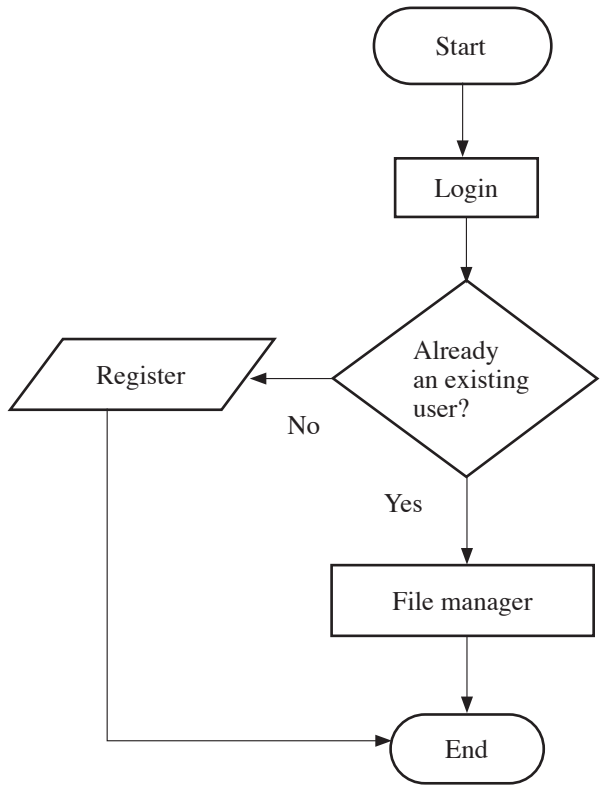


them to create an account and use the system. The login section is for already existing users. In this form, username and password are required. Then the file manager is the section where the main events of the system occur. The file manager's actions are file management, viewing existing files, uploading new files, encrypting, decrypting, downloading, and deleting files. The proposed model framework is displayed in Figure 3.

4. Results

The adapted model uses symmetric key encryption with both RSA and AES as part of hybrid cryptography. Attacks through side channels can target any component. But the security level should be raised using the trivium cipher before using AES. This gain is achieved by avoiding side-channel attacks like differential power analysis (DPA). The study uses documents, images and videos with different files size to simulate the performance of RSA, AES, and DSE in terms of computational time and security. The encryption time and decryption time from the input files were analyzed to determine which algorithms performed efficiently in terms of computational cost. The hybrid architecture lessen the amount of encrypted data with an encryption

Figure 3. Flowchart of the System User Interface



key. As a result, the volume of data exposed by one significant compromise has decreased. This implies that most widely used attacks that require a lot of data, such as well-known "plain text" and "algebraic" attacks, would fail with the suggested approach. The asymmetric private key created at random is used to encrypt the ciphertext-1 once more in the cloud. Anyone who obtains the cipher text cannot decrypt the contents of the multimedia files. The proposed technique is a cloud computing security measure that is broadly applicable due to its low complexity and simple implementation.

Double-stage encryption using RSA and AES algorithms was implemented, and input files of varying contents and sizes were encrypted to test their performance. The algorithms were implemented in Python and were tested to compare their performance. Next, the user chooses the files to encrypt and upload to the database. Different types of files can be selected, ranging from audio to video, text files, etc. When the file has been uploaded, the process of encrypting begins, and then the encrypted file is sent to the cloud. Figure 4 shows the interface to upload a file.

Here, one can choose to delete or decrypt your already encrypted files. Figure 5 shows the interface and decryption for the upload file.

Figure 4. Upload File Interface

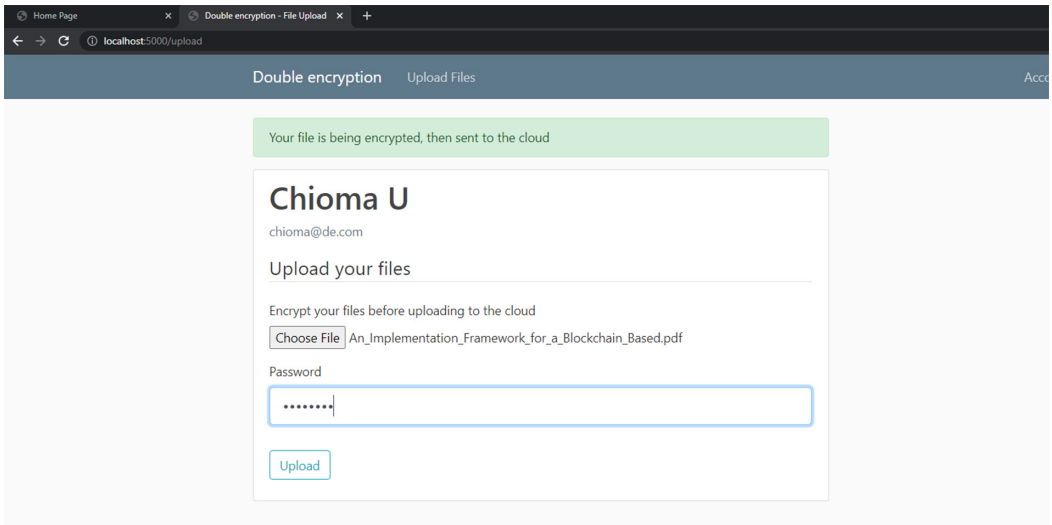
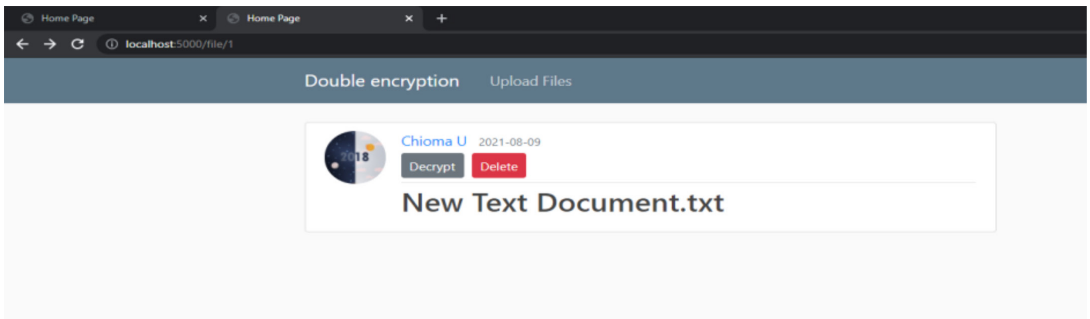


Figure 5. The Interface of Deletion and Decryption



4.1 Performance evaluation

Table 1 presents document files encoding and decoding time for the selected DSE, RSA, and AES Algorithms. The results in Table 1 show that the presented double-stage encryption has the highest encryption and decryption time compared with RSA and AES, respectively. The model has 1513.300 and 1289.600 time required for encoding and decoding. At the same time, the RSA gives 10.291 and 77.154 encryption and decryption time, respectively, and AES has improved encryption and decryption time performance with 43.070 and 78.584, respectively.

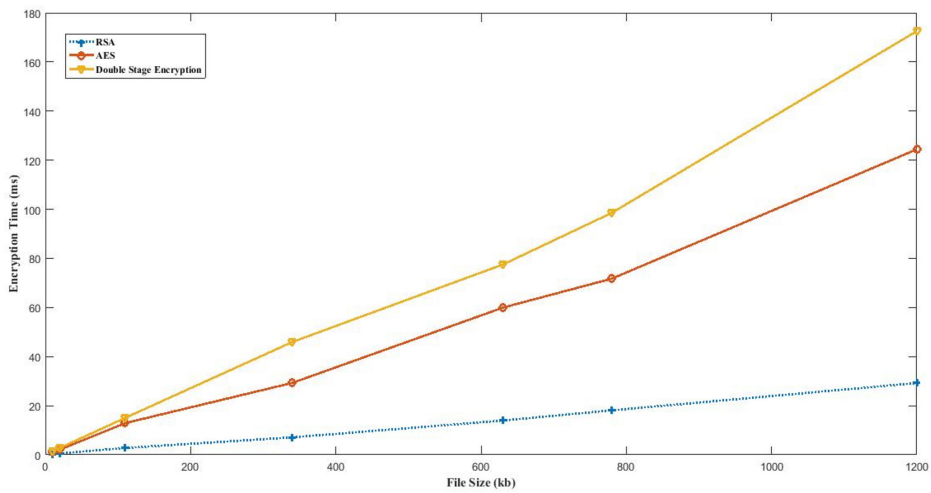
Table 1 displays the data analysis of the encryption time and decryption time obtained during the encryption and decryption of document type of various file size using RSA, AES and DES.

Figure 6 and Figure 7 displayed the encryption and decryption of the text file for the double cryptography model, RSA, and AES, respectively.

Table 1. Text Files Encryption and Decryption Time for Double Stage Encryption (DSE), Rivest-Shamir-Adleman (RSA), and Advanced Encryption Standard (AES)

File size	Double encryption (RSA & AES)		RSA		AES	
	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)
10KB	1.414	5.470	0.319	2.255	1.076	2.277
20KB	2.820	10.940	0.555	4.052	2.119	3.100
110KB	15.057	60.176	2.780	20.860	12.917	27.780
340KB	45.931	185.997	7.099	53.143	29.278	61.614
630KB	77.491	344.642	13.953	105.230	59.934	92.569
780KB	98.538	416.190	18.112	132.997	71.736	115.042
1.2MB	172.548	677.372	29.222	221.542	124.403	247.703
Mean	59.114	242.970	10.291	77.154	43.070	78.584

Figure 6. The Encryption Time for Document Files



In particular, Figure 6 shows that double-stage encryption consumes more time and complexity while encrypting document files of various file sizes. As revealed in Figure 7, double-stage encryption has a higher time complexity, while there is no significant difference between AES and RSA decryption time complexity.

Table 2 shows the video files for encryption and decryption time for DSE, RSA, and AES Algorithms. The results in Table 2 demonstrate

Figure 7. The Decryption Time for Document Files

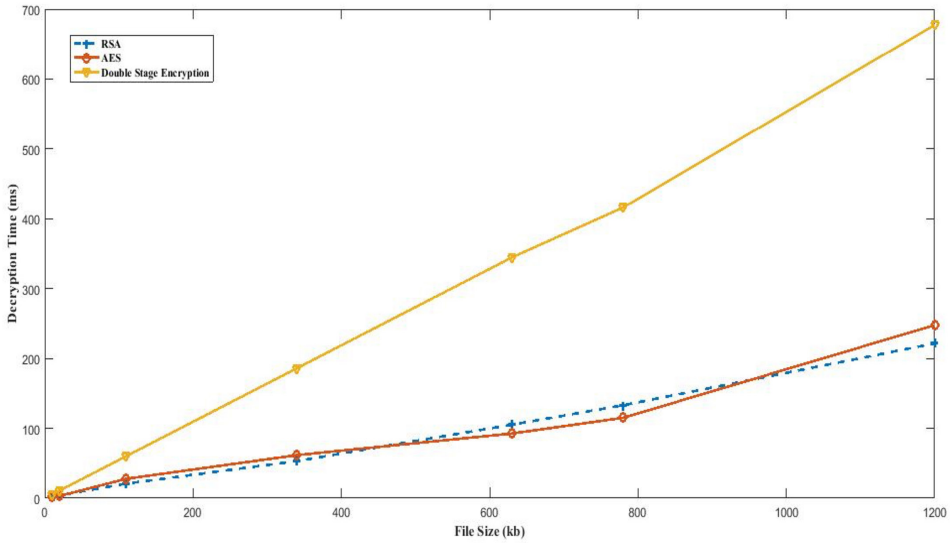


Table 2. Video Files Encryption and Decryption time for Double Stage Encryption (DSE), Rivest-Shamir-Adleman (RSA), and Advanced Encryption Standard (AES)

File size	Double encryption (RSA & AES)		RSA		AES	
	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)
780KB	109.407	1.732	11.640	2.817	11.640	2.817
2.38MB	349.754	12.272	42.612	16.622	42.612	16.622
3.0MB	441.522	48.935	15.401	48.236	15.401	48.236
5.46MB	799.773	28.567	32.340	23.777	32.340	23.777
8.23MB	1311.727	61.485	32.095	22.039	32.095	22.039
Mean	602.437	30.598	26.818	22.698	26.818	22.698

that the double-stage encryption has the highest encryption and decryption time compared to RSA and AES. For example, the model has 602.400 and 30.600 encryption and decryption time, respectively. At the same time, the RSA gives 26.800 and 22.700 encryption and decryption time, respectively, and AES has the same encryption and decryption time with 26.800 and 22.700, respectively.

Figure 8 and Figure 9 displayed the encryption and decryption of the video file for the double cryptography model, RSA, and AES, respectively. Figure 8 shows that RSA and AES consume

Figure 8. The Encryption Time for Video Files

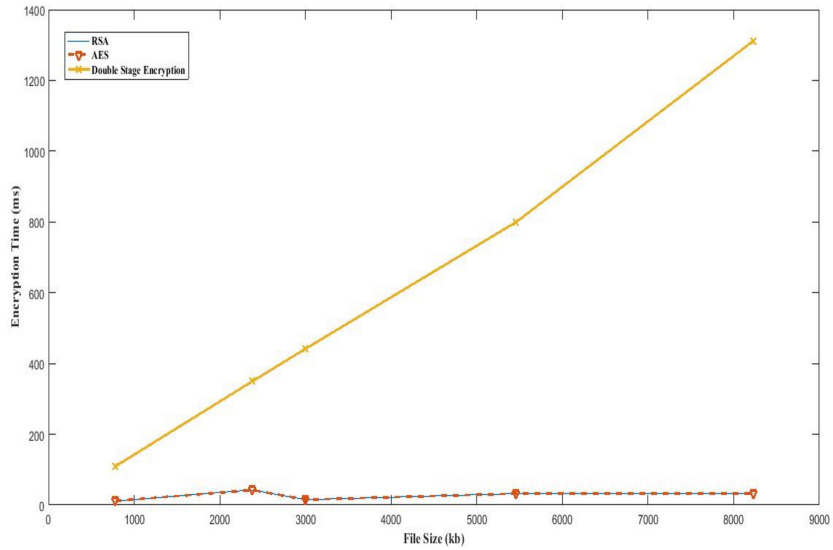
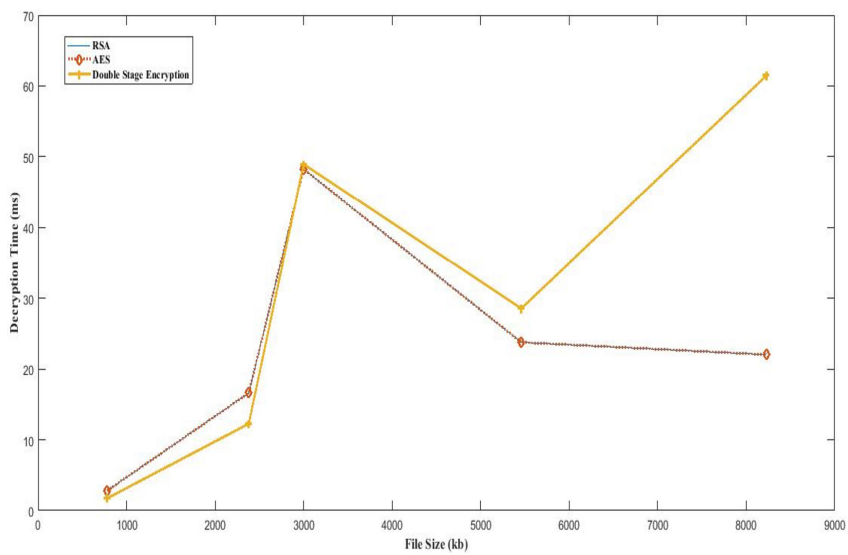


Figure 9. The Decryption Time for Video Files



lesser time complexity during the encryption, while double-stage encryption uses more time complexity. Figure 9 shows that double-stage encryption consumes more run-time during the decryption of video files compared to the two other algorithms.

Table 3 shows the image files for encryption and decryption time for DSE, RSA, and AES Algorithms. The results in Table 3 show that the double-stage encryption has the highest encryption and decryption time when compared with RSA and AES, respectively. For example, the model has 184.400 and 103.000 encryption and decryption time, respectively. At the same time, the RSA gives 26.600 and 208.700 encryption and decryption time, respectively, and AES has encryption and decryption time of 26.800 and 22.700, respectively.

Figure 10 and Figure 11 show the decoding and encryption of image files for the double

Table 3. Image Files Encryption and Decryption time for Double Stage Encryption (DSE), Rivest-Shamir-Adleman (RSA) and Advanced Encryption Standard (AES)

File size	Double encryption (RSA & AES)		RSA		AES	
	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)	Encryption time (ms)	Decryption time (ms)
164KB	23.252	31.92	4.281	31.92	11.64	2.817
224KB	31.771	43.43	5.928	43.43	42.612	16.622
411KB	53.879	73.232	9.7588	73.232	15.401	48.236
1.4MB	212.411	304.829	37.459	304.829	32.34	23.777
2.92MB	600.795	61.485	75.694	590.279	32.095	22.039
Mean	184.422	102.979	26.624	208.738	26.818	22.698

Figure 10. The Encryption Time for Image Files

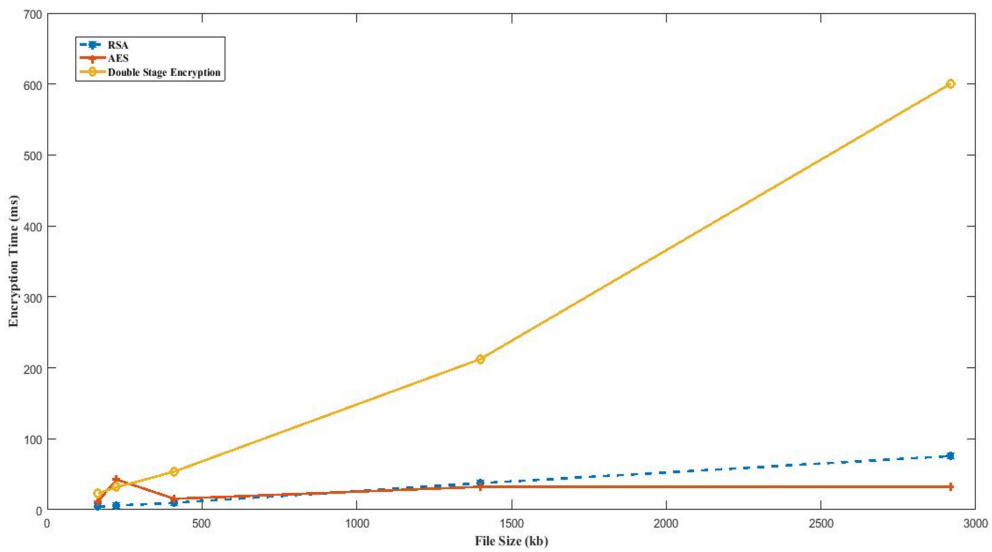
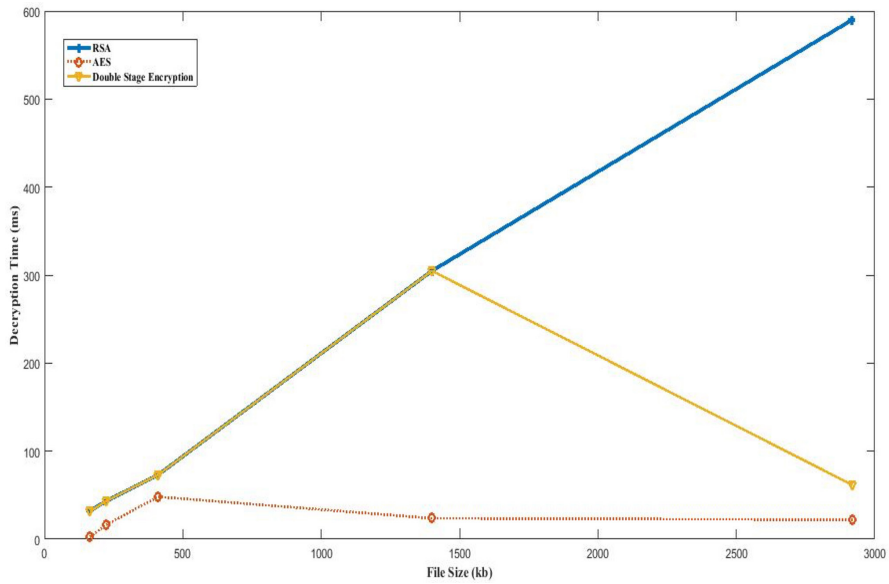


Figure 11. The Decryption Time for Image Files



cryptography model, RSA, and AES, respectively. Figure 10 and Figure 11 show that DSE trade-off time complexity for security performed better. Figure 10 shows that RSA and AES consume lesser time complexity during the encryption, while double-stage encryption uses more time complexity. Figure 11 shows that double-stage encryption uses lesser run-time during the decryption of image files compared to the two other algorithms.

4.2 Comparison of the highlighted algorithms and file types

Table 4 shows that DSE consumes more average time complexity during the encryption process for all the file types used compared to other algorithms.

Table 5 shows that DSE uses more time during each file type's decryption than RSA and AES.

Table 6 shows that the DSE has the highest average avalanche effect. This denotes that double-stage encryption is more secure than the other two algorithms. This can be deduced from the various tables and graphs that double-stage encryption trade-off time complexity to its security which was the goal of this study to improve the security of cloud files. Double-stage encryption combines symmetric and asymmetric encryptions to enhance the security of files in the cloud environment.

This study was carried out to combine AES and RSA to form a DSE to mitigate the issue of safekeeping and privacy in cloud environment. The various tables and graphical analyses in this study show that the DSE takes a long time on average for encryption 83% and decryption 75% compared to RSA and AES singly. The RSA is 68% faster than AES during the encryption process, but there is no significant difference between the two during decryption. The Avalanche

Table 4. Comparison of the Average Encryption Time of the Algorithms

Files	RSA	AES	DSE
Document files	10.291	43.070	59.114
Image files	26.624	26.818	184.422
Video files	26.818	26.818	602.437

Table 5. Comparison of the Average Decryption Time of the Algorithms

Files	RSA	AES	DSE
Document files	77.154	78.584	242.970
Image files	208.738	22.618	102.979
Video files	22.698	22.698	30.598

Table 6. Comparison of the Overall Average Avalanche Effect

	AES	RSA	DSE
One bit change (%)	40.18	38.97	57.92
Key change (%)	39.22	38.82	51.58

effect testing showed DSE to be 17% higher than singly testing AES and RSA, which means it is more secure than RSA and AES as single encryption schemes.

5. Discussion

The study presented a suitable hybrid model using double-stage encryption and decryption to enhance cloud security using RSA and AES cryptography algorithms. Since RSA cryptography primarily operates on bits, this study employs input integers *N*, first converting it to binary and then using the RSA technique to encrypt each of its bits. Then the second layer of the AES algorithm receives a concatenated version of all these strings. This model was implemented to improve cloud security using the Python programming language and cryptographic method. Similar to encryption, decryption uses the same methods, except as an initial step, it requires the encrypted text as an input, then uses it in the AES decryption layer, after which the message has been decrypted before being sent to the RSA layer for decryption, and once that bit has been received, compare the bits and convert them back to their original integer form. This is how the multi-layer approach works. This is revocable and significantly more effective than other encryption methods. Similarly, the AES encryption algorithm's key creation procedure generates a safe key for both encoding and decoding.

AES is a symmetric block cipher capable of encrypting and decrypting data securely. It uses the same key for both operations. The AES technique is also known as a variable-length key block cipher. Since the key does not change and is speedier than most encryption techniques, this is appropriate for various uses. The decryption and encryption of cloud data have been chosen for text files. Owing to the hybrid method suggested in this study, hacking would be substantially very difficult for intruders. This hybrid method uses encryption algorithms to provide improved storage and security over a cloud environment. More information must be protected using encryption techniques to prevent data attacks from hackers. The RSA encryption algorithm's performance is adaptable and secure by design.

The security and privacy concerns in the cloud are developed using the AES algorithm. A symmetric key block is used for both encryption and decryption, and it generates the security key. The RSA encryption adds a new layer to cloud storage and ensures data secrecy because no process stage exposes information in plain text. As the demand for security grows, a trustworthy authentication system is required to limit illegal access and assist secure data. Even though cloud storage offers several advantages, numerous security concerns still need to be addressed. The efficacy of performance is anticipated to increase with the employment of various other algorithms in the same procedure. If the security concerns are addressed, cloud storage solutions for small and large businesses will be the future. In the future, using multiple authentication factors to give sensitive data even more security will be considered. In order to increase the overall system confidence, we will also work to maintain multiple layers of protection for various types of data in the cloud. Additionally, the use of AI-driven solutions to automate the protection process and give the key management and distribution stages a more autonomous aspect will be explored.

6. Conclusions

This study highlights the strengths of double stage cryptography to secure files in a cloud computing context, leveraging the RSA and AES. The aim was to boost file security in the cloud computing environment. RSA and AES cryptographic algorithms were used separately to encrypt and decrypt data sent to the cloud. Also, these two algorithms were merged to achieve a double-stage encryption scheme used to protect data sent to the cloud to improve data security. Based on the various experimental results performed on text, image, and video files, it was shown that double stage encryption takes a longer amount of time on average for encryption 83% and decryption 75% compared to RSA and AES singly. The RSA is 68% faster than AES during the encryption process, but there is no significant difference between the two during decryption. The Avalanche effect testing showed the DSE to be 17% higher than singly testing AES and RSA. Therefore, the DSE scheme is more secure than RSA and AES as single encryption schemes. This study recommends this approach, combining the RSA and AES algorithms for file security in cloud computing. The purpose of this technology was to improve data security while increasing information confidentiality. The drawback of this approach is the computational cost. However, enhancing the security level of files in the cloud

environment is highly pertinent when sending data over an unsecured platform. Implementing a DSE scheme can be complex, and any errors in the implementation can compromise the system's security. This complexity can increase the development time and cost of the system. The DSE scheme may not be scalable for large-scale distributed systems where multiple users can access data simultaneously. The scheme may require additional mechanisms to manage the access control and encryption keys for each user.

Data Availability Statement

The data that support the findings of this paper is available upon reasonable request from the corresponding author.

Conflicts of Interest

The authors declare no conflict of interest related to this work.

Acknowledgments

The work of Agbotiname Lucky Imoize is supported in part by the Nigerian Petroleum Technology Development Fund (PTDF) and in part by the German Academic Exchange Service (DAAD) through the Nigerian-German Postgraduate Program under Grant 57473408.

References

Abd Elminaam, D. S., Mousa, M. A. W., & Abd El Fattah, M. (2022). Secure data storage in the cloud by using DNA and chaos cryptography. In A. Bahaa-Eldin, A. AbdelRaouf, N. Shorim, S. Refaat, & S. E. Elbohy (Eds.), *2022 2nd international mobile, intelligent, and ubiquitous computing conference (MIUCC)* (pp. 175-182). IEEE. https://doi.org/10.1109/MIUCC55081.2022.9781704

AbdulRaheem, M., Awotunde, J. B., Jimoh, R. G., & Oladipo, I. D. (2021). An efficient lightweight cryptographic algorithm for IoT security. In S. Misra & B. Muhammad-Bello (Eds.), *Communications in computer and information science: Vol. 1350. International conference on information and communication technology and applications* (pp. 444-456). Springer. https://doi.org/10.1007/978-3-030-69143-1_34

AbdulRaheem, M., Balogun, G. B., Abiodun, M. K., Taofeek-Ibrahim, F. A., Tomori, A. R., Oladipo, I. D., & Awotunde, J. B. (2021). An enhanced lightweight speck system for cloud-based smart healthcare. In H. Florez & M. F. Pollo-Cattaneo (Eds.), *Communications in computer and information science: Vol. 1455. Applied informatics. ICAI 2021* (pp. 363-376). Springer. https://doi.org/10.1007/978-3-030-89654-6_26

AbdulRaheem, M., Oladipo, I. D., González-Briones, A., Awotunde, J. B., Tomori, A. R., & Jimoh, R. G. (2022). An efficient lightweight speck technique for edge-IoT-based smart healthcare systems. In A. K. Bhoi, V. H. C. de Albuquerque, S. Nath Sur, & P. Barsocchi (Eds.), *5G IoT and edge computing for smart healthcare* (pp. 139-162). Academic Press. https://doi.org/10.1016/B978-0-323-90548-0.00005-X

Abikoye, O. C., Oladipupo, E. T., Imoize, A. L., Awotunde, J. B., Lee, C.-C., & Li, C.-T. (2023). Securing critical user information over the internet of medical things platforms using a hybrid cryptography scheme. *Future Internet, 15*(3), Article 99. https://doi.org/10.3390/fi15030099

Abiodun, M. K., Adeniyi, E. A., Awotunde, J. B., Bhoi, A. K., AbdulRaheem, M., & Oladipo, I. D. (2022). A framework for the actualization of green cloud-based design for smart cities. In S. Nath Sur, B. E. Balas, A. K. Bhoi, & A. Nayyar (Eds.), *EAI/Springer innovations in communication and computing. IoT and IoE driven smart cities* (pp. 163-182). Springer. https://doi.org/10.1007/978-3-030-82715-1_8

Abiodun, M. K., Awotunde, J. B., Adeniyi, A. E., Ademuagun, D., & Aremu, D. R. (2022). Securing digital transaction using a three-level authentication system. In O. Gervasi, B. Murgante, S. Misra, A. M. A. C. Rocha, & C. Garau (Eds.), *Lecture notes in computer science: Vol. 13380. International conference on computational science and its applications 2022* (pp. 135-148). Springer. https://doi.org/10.1007/978-3-031-10542-5_10

Abod, Z. A., Abbas, M. S., & Bermani, A. K. (2020). Image security system using hybrid cryptosystem. *Periodicals of Engineering & Natural Sciences (PEN), 8*(4), 2007-2018.

Ahmed, A. A., & Barukab, O. M. (2022). Unforgeable digital signature integrated into lightweight encryption based on effective ECDH for cybersecurity mechanism in internet of things. *Processes, 10*(12), Article 2631. https://doi.org/10.3390/pr10122631

Ayo, F. E., Awotunde, J. B., Olalekan, O. A., Imoize, A. L., Li, C.-T., & Lee, C.-C. (2023). CBFISKD: A combinatorial-based fuzzy inference system for keylogger detection. *Mathematics, 11*(8), Article 1899. https://doi.org/10.3390/math11081899

Bansal, A., & Agrawal, A. (2017). Providing security, integrity and authentication using ECC algorithm in cloud storage. In *2017 international conference on computer communication and informatics (ICCCI)* (pp. 1-5). IEEE. https://doi.org/10.1109/ICCCI.2017.8117749

Chakrabarti, S., & Suresh Babu, G. N. K. (2021). The security enhancement of symmetric key crypto mechanism based on double stage secret model. *Information Security Journal: A Global Perspective, 30*(6), 325-341. https://doi.org/10.1080/19393555.2020.1842945

Chueh, J. S., & Sun, M. T. (2017). Design and implementation of security system for cloud storage. In *2017 19th Asia-Pacific network operations and management symposium (APNOMS)* (pp. 129-134). IEEE. https://doi.org/10.1109/APNOMS.2017.8094191

Erondu, U. I., Adebayo, N., Arowolo, M. O., & Abiodun, M. K. (2022). A review on different encryption and decryption approaches for securing data. In A. M. Tyagi (Ed.), *Handbook of research on technical, privacy, and security challenges in a modern world*

(pp. 357-370). https://doi.org/10.4018/978-1-6684-5250-9.ch019

Garg, P., Goel, S., & Sharma, A. (2017). Security techniques for cloud computing environment. In P. N. Astya, A. Swaroop, V. Sharma, M. Singh, & K. Gupta (Eds.), *2017 international conference on computing, communication and automation (ICCCA)* (pp. 771-776). IEEE. https://doi.org/10.1109/CCAA.2017.8229900

Ghiasi, M., Niknam, T., Wang, Z., Mehrandezh, M., Dehghani, M., & Ghadimi, N. (2023). A comprehensive review of cyber-attacks and defense mechanisms for improving security in smart grid energy systems: Past, present and future. *Electric Power Systems Research, 215*, Part A, Article 108975. https://doi.org/10.1016/j.epsr.2022.108975

Han, K., Li, Q., & Deng, Z. (2016). Security and efficiency data sharing scheme for cloud storage. *Chaos, Solitons & Fractals, 86*, 107-116. https://doi.org/10.1016/j.chaos.2016.02.010

Imoize, A. L., Ben-Adeola, B. S., & Adebisi, J. A. (2020). Development of a multifactor-security-protocol system using ambient noise synthesis. *EAI Endorsed Transactions on Security & Safety, 6*(22), Article e4. http://doi.org/10.4108/eai.13-7-2018.163979

Jimoh, R. G., Olusanya, O. O., Awotunde, J. B., Imoize, A. L., & Lee, C.-C. (2022). Identification of risk factors using ANFIS-based security risk assessment model for SDLC phases. *Future Internet, 14*(11), Article 305. https://doi.org/10.3390/fi14110305

Kumar, V., Malik, N., Singla, J., Jhanjhi, N. Z., Amsaad, F., & Razaque, A. (2022). Light weight authentication scheme for smart home IoT devices. *Cryptography, 6*(3), Article 37. https://doi.org/10.3390/cryptography6030037

Kumbhare, A., & Thakur, P. K. (2022). Security and privacy of biomedical data in IoMT. In A. Prasanth, D. Lakshmi, R. K. Dhanaraj, B. Balusamy, & P. C. Sherimon (Eds.), *Cognitive computing for internet of medical things* (pp. 77-104). Chapman and Hall/CRC.

Li, Y., Gai, K., Qiu, L., Qiu, M., & Zhao, H. (2017). Intelligent cryptography approach for secure distributed big data storage in cloud computing. *Information Sciences, 387*, 103-115. https://doi.org/10.1016/j.ins.2016.09.005

Maitri, P. V., & Verma, A. (2016). Secure file storage in cloud computing using hybrid cryptography algorithm. In *2016 international conference on wireless communications, signal processing and networking (WiSPNET)* (pp. 1635-1638). IEEE. https://doi.org/10.1109/WiSPNET.2016.7566416

Meshram, C., Lee, C.-C., Bahkali, I., & Imoize, A. L. (2023). An efficient fractional chebyshev chaotic map-based three-factor session initiation protocol for the human-centered IoT architecture. *Mathematics, 11*(9), Article 2085. https://doi.org/10.3390/math11092085

Meshram, C., Ibrahim, R. W., Meshram, S. G., Imoize, A. L., Jamal, S. S., & Barve, S. K. (2022). An efficient remote user

authentication with key agreement procedure based on convolution-Chebyshev chaotic maps using biometric. *The Journal of Supercomputing, 78*(10), 12792-12814. https://doi.org/10.1007/s11227-021-04280-8

Meshram, C., Imoize, A. L., Jamal, S. S., Tambare, P., Alharbi, A. R., & Hussain, I. (2022). An efficient three-factor authenticated key agreement technique using FCM under HC-IoT architectures. *Computers, Materials & Continua, 72*(1), 1373-1389. https://doi.org/10.32604/cmc.2022.024996

Meshram, C., Imoize, A. L., Aljaedi, A., Alharbi, A. R., Jamal, S. S., & Barve, S. K. (2021). A provably secure IBE transformation model for PKC using conformable Chebyshev chaotic maps under human-centered IoT environments. *Sensors, 21*(21), Article 7227. https://doi.org/10.3390/s21217227

Mohammed, C. M., & Zebaree, S. R. M. (2021). Sufficient comparison among cloud computing services: IaaS, PaaS, and SaaS: A review. *International Journal of Science & Business, 5*(2), 17-30.

Murad, S. H., & Rahouma, K. H. (2021). Implementation and performance analysis of hybrid cryptographic schemes applied in cloud computing environment. *Procedia Computer Science, 194*, 165-172. https://doi.org/10.1016/j.procs.2021.10.070

Ogundokun, R. O., Awotunde, J. B., Adeniyi, E. A., & Ayo, F. E. (2021). Crypto-Stegno based model for securing medical information on IOMT platform. *Multimedia Tools & Applications, 80*(21/23), 31705-31727. https://doi.org/10.1007/s11042-021-11125-2

Pavani, M., & Trinatha Rao, P. (2019). Adaptive PSO with optimised firefly algorithms for secure cluster-based routing in wireless sensor networks. *IET Wireless Sensor Systems, 9*(5), 274-283. https://doi.org/10.1049/iet-wss.2018.5227

Rahardjo, M. R. D., & Shidik, G. F. (2017, October). Design and implementation of self encryption method on file security. In *2017 international seminar on application for technology of information and communication (iSemantic)* (pp. 181-186). IEEE. https://doi.org/10.1109/ISEMANTIC.2017.8251866

Rana, P., Batra, I., Malik, A., Imoize, A. L., Kim, Y., Pani, S. K., Goyal, N., Kumar, A., & Rho, S. (2022). Intrusion detection systems in cloud computing paradigm: Analysis and overview. *Complexity, 2022*, Article 3999039. https://doi.org/10.1155/2022/3999039

Rizk, R., & Alkady, Y. (2015). Two-phase hybrid cryptography algorithm for wireless sensor networks. *Journal of Electrical Systems & Information Technology, 2*(3), 296-313. https://doi.org/10.1016/j.jesit.2015.11.005

Sajay, K. R., Babu, S. S., & Vijayalakshmi, Y. (2019). Enhancing the security of cloud data using hybrid encryption algorithm. *Journal of Ambient Intelligence & Humanized Computing*. https://doi.org/10.1007/s12652-019-01403-1

Selvanayagam, J., Singh, A., Michael, J., & Jeswani, J. (2018). Secure file storage on

the cloud using cryptography. *International Research Journal of Engineering & Technology (IRJET), 5*(3), 2044-2047.

Shukla, P. K., Aljaedi, A., Pareek, P. K., Alharbi, A. R., & Jamal, S. S. (2022). AES based white box cryptography in digital signature verification. *Sensors, 22*(23), Article 9444. https://doi.org/10.3390/s22239444

Subasree, S., & Sakthivel, N. K. (2010). Design of a new security protocol using hybrid cryptography algorithms. *IJRRAS, 2*(2), 95-103.

Timothy, D. P., & Santra, A. K. (2017). A hybrid cryptography algorithm for cloud computing security. In *2017 international conference on microelectronic devices, circuits and systems (ICMDCS)* (pp. 1-5). IEEE. https://doi.org/10.1109/ICMDCS.2017.8211728

Tripathy, S. S., Imoize, A. L., Rath, M., Tripathy, N., Bebortta, S., Lee, C.-C., Chen, T.-Y., Ojo, S., Isabona, J., & Pani, S. K. (2022). A novel edge-computing-based framework for an intelligent smart healthcare system in smart cities. *Sustainability, 15*(1), Article 735. https://doi.org/10.3390/su15010735

Ullah, S., Zheng, J., Din, N., Hussain, M. T., Ullah, F., & Yousaf, M. (2023). Elliptic curve cryptography; applications, challenges, recent advances, and future trends: A comprehensive survey. *Computer Science Review, 47*, Article 100530. https://doi.org/10.1016/j.cosrev.2022.100530

Zhu, S.-H. (2011). Research of hybrid cipher algorithm application to hydraulic information transmission. In *2011 international conference on electronics, communications and control (ICECC)* (pp. 3873-3876). IEEE. https://doi.org/10.1109/ICECC.2011.6066481

(Received: 2023/1/16; Accepted: 2023/6/16)

圖書資訊學刊　第21卷第2期　（2023.12）　頁1-26
https://doi.org/10.6182/jlis.202312_21(2).001

於雲端運算系統中使用混合密碼學與兩階段加密方法之分析

Analysis of a Double-stage Encryption Scheme Using Hybrid Cryptography to Enhance Data Security in Cloud Computing Systems

Moses Kazeem Abiodun[1], Agbotiname Lucky Imoize[2,3], Joseph Bamidele Awotunde[4], Cheng-Chi Lee[5,6,7], Abidemi Emmanuel Adeniyi[8], Ugbaja Chioma[9], Chun-Ta Li[10]

摘　要

最近，網路使用者對於將其個人敏感資訊（例如銀行帳戶資料、健康紀錄和其他重要資訊）儲存在雲端空間會感到擔憂，因這些資訊在雲端很容易受到第三方的攻擊且現今多種威脅模式亦影響到雲端的安全性。本研究將提出一使用混合密碼學之兩階段加密方法，以消除基於雲端系統中的弱點並使其在雲端環境之資料存取變得更安全。本系統先使用RSA對檔案進行加密後，再使用AES金鑰對已加密之檔案進行加密，並透過雜湊函式以確保加密金鑰之安全性。本雲端系統透過Python及MongoDB進行實作並藉由雪崩效應測試進行分析，其分析結果顯示本系統所提出之兩階段加密方法比單獨使用AES和RSA單一加密方法之安全性高出17%，這也意味著本研究所提出之方法會讓雲端系統變得更加安全。

關鍵字：雲端運算、密碼學、安全與隱私、RSA加密演算法、進階加密標準

[1,9] 奈及利亞地標大學電腦科學系
Department of Computer Science, Landmark University, Omu-Aran, Nigeria
[2] 奈及利亞拉各斯大學電機與電子工程系
Department of Electrical and Electronics Engineering, University of Lagos, Lagos, Nigeria
[3] 德國波鴻魯爾大學電機與資訊科技學系
Department of Electrical Engineering and Information Technology, Ruhr University, Bochum, Germany
[4] 奈及利亞伊洛林大學電腦科學系
Department of Computer Science, University of Ilorin, Ilorin, Nigeria
[5] 輔仁大學教育學院體育健康資訊科技研究發展中心
Research and Development Center for Physical Education, Health, and Information Technology, College of Education, Fu Jen Catholic University, New Taipei, Taiwan

[6] 輔仁大學圖書資訊學系
Department of Library and Information Science, Fu Jen Catholic University, New Taipei, Taiwan
[7] 亞洲大學資訊工程學系
Department of Computer Science and Information Engineering, Asia University, Taichung, Taiwan
[8] 奈及利亞珍貴基石大學電腦科學系
Department of Computer Science, Precious Cornerstone University, Ibadan, Nigeria
[10] 輔仁大學人工智慧與資訊安全學士學位學程
Program of Artificial Intelligence and Information Security, Fu Jen Catholic University, New Taipei, Taiwan
* 通訊作者Corresponding Author: 李正吉Cheng-Chi Lee, E-mail: cclee@mail.fju.edu.tw

註：本中文摘要由作者提供。
以APA格式引用本文：Abiodun, M. K., Imoize, A. L., Awotunde, J. B., Lee, C.-C., Adeniyi, A. E., Chioma, U., & Li, C.-T. (2023). Analysis of a double-stage encryption scheme using hybrid cryptography to enhance data security in cloud computing systems. *Journal of Library and Information Studies, 21*(2), 1-26. https://doi.org/10.6182/jlis.202312_21(2).001
以Chicago格式引用本文：Moses Kazeem Abiodun, Agbotiname Lucky Imoize, Joseph Bamidele Awotunde, Cheng-Chi Lee, Abidemi Emmanuel Adeniyi, Ugbaja Chioma, and Chun-Ta Li, "Analysis of a double-stage encryption scheme using hybrid cryptography to enhance data security in cloud computing systems," *Journal of Library and Information Studies* 21, no. 2 (2023): 1-26. https://doi.org/10.6182/jlis.202312_21(2).001