Providing a Hybrid and Symmetric Encryption Solution to Provide Security in Cloud Data Centers

Desong Shen*
School of Business and Trade, Anhui Wenda University of Information Engineering, Hefei, 230121, China

***Abstract*—One of the most crucial components of information technology infrastructure in the modern world is cloud data centers. Customers have access to these data centers' infrastructure and software, which enable them to store and process massive amounts of data. However, the security and protection of private data in cloud data centers is a serious problem that needs effective and c solutions. Security and privacy issues exist because cloud computing outsources the processing of sensitive data. Consumer worries about cloud infrastructure security remain, particularly those related to data privacy. A thorough analysis of research efforts in the area of cloud security is the main objective of this study. In order to do this, a variety of models were evaluated, their advantages and disadvantages were identified, and a viable security solution based on symmetric algorithms was put forth. The original text in the proposed solution (Hybrid encryption algorithm) is first encrypted using the faster symmetric key method AES, and then its key is encrypted using the faster asymmetric key scheme RSA. This increases efficiency and speed. This method will shorten the time required for data encryption while enhancing its security. The final step was implementing the desired solution in the Eclipse software environment and comparing it against the Blowfish and RSA algorithms. The evaluation's findings indicate that the solution is more advantageous, which has resulted in a nearly two-fold decrease in execution time and a marked increase in throughput when compared to the RSA algorithm. Additionally, the execution time has shrunk, and throughput has been vastly improved compared to the Blowfish method.**

Keywords*—Hybrid encryption algorithm; security; cloud computing; symmetric algorithms

I. Introduction

In the ever-evolving IT landscape, cloud data centers stand as giants of innovation, transforming the way we store, process and access data on a scale previously unimaginable. Their ubiquitous presence in our digital lives has ushered in an era of unparalleled convenience, accessibility, and efficiency. However, in the midst of this digital utopia, a looming specter looms – the great challenge of protecting sensitive data in cloud environments. The relentless advancement of technology has brought with it increased security concerns, and data protection has become a critical battleground across the vast expanses of cloud data centers. As we unlock the enormous potential of cloud computing, we are acutely aware of the vulnerabilities it presents. The imperative to fortify data storage fortresses against malicious intrusions and data breaches has never been more prominent. Since the start of its operation, the Internet has seen several changes, some of which have altered how people live today because cloud computing offers consumers a wide range of facilities as a service, this new technology has swiftly gained popularity [1]. Naturally, any modification and fresh idea in the technological world has its advantages, drawbacks, and issues [2]. This rule applies to using cloud computing as well [3]. We can include the absence of time and location constraints, easy resource sharing, and decreased capital and operating expenses as benefits of cloud computing [4]. Because cloud computing offers scalable resources as a service over the Internet, it has created a number of difficulties for the field's professionals, including data protection or preservation [5]. That is privacy. Early adopters still hesitate to move their businesses to the cloud despite the entire buzz [6]. The difficulties associated with data privacy and information protection continue to disrupt the cloud computing market, and security is one of the primary problems that are slowing down the expansion of cloud computing [7]. Other crucial and major elements of the current model shouldn't be threatened or jeopardized by a new model whose goal is to enhance its features [8]. Cloud architecture poses risks to the security of these technologies when they are employed in a cloud environment [9]. Users of cloud services should be cautious and knowledgeable about the risks associated with data breaches in this novel environment [10]. User data is typically safeguarded from hackers using a variety of encryption techniques in order to assure security and secrecy. This technique has also been applied in cloud computing settings. However, reliability is not ensured when numerous services are used concurrently to carry out operations like combining functions [11]. For instance, if a malicious application interferes with the service used by other customers then the numerous cloud platforms are shared by many clients. Other people's environments are also impacted by it [12]. Common security threats for cloud computing include integrity, availability, and confidentiality [13]. Attacks in this system can be split into two categories, assaults from both internal and external parties. Internal assaults are those in which the perpetrator seeks to get access to the network and its operations by obtaining virtual machine control, knowing the password or other authentication credentials, or both [14].

In contrast, attackers that use external attacks want to distribute false routing information or stop nodes from offering services. An internal intruder poses a greater threat than an outside one. A secure link between data centers and users is created concurrently using symmetric encryption as well as hybrid encryption [15]. With the aid of this technique, key structures for data encryption and decryption in cloud environments are made to be secure and dependable [16]. The benefits of utilizing hybrid and symmetric encryption in cloud data centers include boosting user privacy protection, lowering the risk of cyberattacks, strengthening protection against

intrusion, and increasing security and protection of sensitive information retrieval. Security mechanisms against hackers (attackers) including: Firewalls and intrusion detection systems (IDS), access control and authentication, encryption, security patch management, intrusion response plan, network segmentation, security information and event management (SIEM), and monitoring and logging.

Additionally, the usage of symmetric and hybrid encryption boosts the effectiveness of cloud data centers [17]. Additionally, symmetric and hybrid encryption can withstand physical assaults. Information is kept in decryption mode, and the likelihood of unwanted access to them is decreased by utilizing detection and prevention techniques. These techniques enable cloud data centers to deliver services more dependably while limiting illegal data access [18]. When using symmetric encryption algorithms such as AES, a separate MAC algorithm or an authenticated encryption mode (such as AES-GCM or AES-CCM) is used to simultaneously ensure data confidentiality and integrity/authentication. These modes incorporate MAC functions in the encryption process to achieve data integrity protection. Also, when a secure cryptographic system is implemented, a well-established MAC algorithm (such as HMAC, CMAC) or an authenticated encryption mode should be considered to protect data integrity and validity in addition to encryption. In order to guarantee security and boost efficiency in cloud data centers, a novel approach utilizing hybrid and symmetric encryption is given in this article. Combining many distinct encryption techniques at once is known as hybrid encryption. By using this technique, it is possible to increase security and defend against decoding attempts. A secure link between data centers and users is created concurrently using symmetric encryption as well as hybrid encryption. With the aid of this technique, key structures for data encryption and decryption in cloud environments are made to be secure and dependable. The author's contribution to this work can be summed up as follows:

- Offering an integrated solution based on the RSA and AES algorithms to improve security.
- Shortening the duration of encryption operations.

The remainder of the essay is structured as follows: A summary of earlier efforts is provided in the Section II. Section III goes into further detail about the suggested approach. The evaluation and effectiveness of the suggested method are described in the Section IV. Section V contains the conclusion and recommendations for further research.

II. Related Works

One of the most talked-about subjects in the IT world is cloud computing. Due to the cloud's vast resources and low entry barrier, it is being enthusiastically embraced by many new businesses. The cloud, however, has both pros and cons, just like any other topic. Information about cloud users is saved remotely. Therefore, one of the primary concerns of any firm contemplating a move to the cloud is cloud data security. Firewalls and VPNs (virtual private networks) are two of the most popular ways for data owners to protect their information at home or the office. The company is the data owner, but it uses uncontrolled cloud servers to store sensitive information, and its users can access this information when needed. For this reason, there is a security risk associated with storing client data elsewhere.

As a result, safeguarding data in the cloud has emerged as a key field of study [19]. Various cryptographic methods, including AES (a symmetric cryptographic method), SHA-1 (a hashing method), and ECC, are used in [20], with data first being organized into categories according to its sensitivity and significance. In the asymmetric cryptography method of elliptic curve cryptography, most existing works rely on a single key for encryption and decryption, making them vulnerable to a wide range of well-known harmful attacks. As a result, the hybrid algorithm we developed uses two independent keys applicable to any encoding/decoding process. If you wish to access data stored in the cloud, you'll need to sign up with both your cloud service provider and the cloud's owner which is required for decrypting cloud-based information. The purpose of this research [21] was to create a novel approach to cloud data security based on hybrid encryption architecture. Encrypting user data before it is sent to the cloud is an active study area because of the prevalence of malicious actors in this environment.

Since the scope of cloud computing security concerns extends to data access control, identity management, auditing, integrity control, and risk management, a hybrid cryptosystem was developed to address these concerns. Data privacy is addressed with the symmetric Blowfish algorithm rather than the asymmetric RSA scheme. Authentication is the focus of the algorithm. The Secure Hash-2 algorithm is also used in this method; therefore, the data may be trusted. Based on the current research results, it has been determined that the suggested method offers both high security for data transmission over the Internet and easy, on-demand network access to the manufacturer's shared pool of computer resources, including the latter two in particular. Due to concerns over data security, many large companies are hesitant to adopt cloud computing services. There have been numerous incidents of cloud security breaches documented over the past few years, despite the cloud service provider's claim of having a solid third-party security system. Therefore, for cloud service providers to succeed, they must have stringent safety measures. Strong security for user data is proposed in [22] via a two-layer agent-based hybrid framework that combines symmetric and asymmetric key methods. The risk of data misuse by the cloud service provider is also removed because the user alone controls the decryption process. This framework improves security without slowing down the virtual machine's processing speed since the two cryptographic algorithms utilized are optimized for low key size, low encryption time, and high speed. In [23], we examine hybrid cryptography from 2014 until the beginning of 2019. Eight are based on a tabular survey format that is easy to use, while the remaining 12 are comprehensive polls. The primary goal of this review paper is to expand the knowledge base of novice cryptography researchers, students, and practitioners. The lack of attention paid to user authentication and the improper usage of hybrid algorithms is the area where more study is needed. As a result of all this, cloud customers are beginning to worry about the

security of their data while it is being stored on these outside managed servers. Information security is necessary to prevent these data breaches and other risks. Information security relies heavily on encryption. The user employs a couple of different encryption algorithms to keep their cloud data safe. Researchers look at trust and its application in distributed computing [24]. A summary of proposed trust models for various distributed system types is provided. The proposed trust management systems for cloud computing are examined with a focus on their capabilities, realistic heterogeneous cloud applicability, and implementation viability. In actuality, data security refers to the safeguarding of information's availability, completeness, and secrecy.

Research [25] ensures that, when necessary (Availability), only authorized users have access to accurate and comprehensive information (Completeness). Information security aims to shield data and information systems from misuse, failure, disclosure, and unauthorized access. Based on the Security as a Service (SECaaS) concept, they suggested multi-layer and multi-level encryption architecture for cloud computing [26]. This article also makes the point that the implementation of this architecture is adaptable to scalable systems with various needs and can unite heterogeneous networks and different operating systems. K-anonymity, an encryption model to safeguard personal information, is provided in [27]. By employing this technique, user data can be protected from unauthorized disclosure. The term "data" in this article refers to specific, unique information that is conceptually arranged as a table with rows for reports and columns for strings. An outline of the cloud's security issues and major difficulties can be found in [28]. By the end of this piece, they have determined that a significant portion of data security may be achieved by the employment of cryptographic methods.

Additionally, it has been found that using both symmetric and asymmetric encryption at once can speed up message transmission and identity verification considerably. The employment of extensible identity recognition protocol in cloud computing is plagued by issues listed in [29]. Cloud computing authentication issues have been resolved using EAP. This technique nullifies dangers from data manipulation, DoS assaults, and identity theft. However, in this manner, a powerful algorithm and encryption are required for the cloud environment, ensuring that the client's data and the data transferred between the client and the cloud provider are encrypted. The EAP approach also has additional issues. A framework for the authentication choice was supplied by the study conducted in [30], referred to as the "Trust Cube" in this study. A high-level framework of authentication procedures is offered in this solution. The client device, the data collector, the authentication engine, and the authentication consumers are taken into account in this architecture. Each of these participants must be validated before data can be sent through the authentication engine.

III. Suggested Method

The suggested approach is broken up into two stages. The first stage focuses on the transfer and secure storage of data on the cloud. Data is only made available to the authorized user after passing all security mechanisms in the second step, which handles data retrieval from the cloud and data validation and integrity. A hybrid encryption technique is employed in both phases as the best option for data encryption and decryption, which is done by using the data from the user side to the server or vice versa. The proposed model and method to provide a general framework to guarantee data security and comprehensiveness are shown in the general structure in Fig. 1. You can see the overall layout and the different types of actions done on the data at each level in this structure.

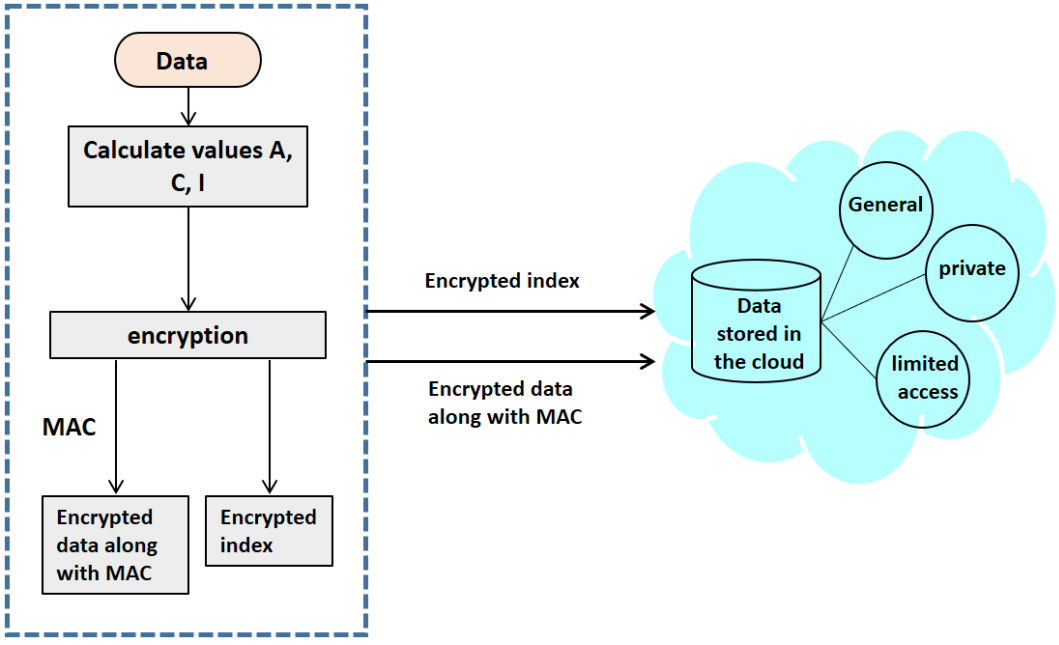

Fig. 1. General outline of the proposed model.

In actuality, this encryption method combines the strengths of AES and RSA, two symmetric encryption methods, to establish the key cryptographic parameters. There was good assurance of confidentiality, integrity, and availability. A systematic and comprehensive method was employed to explore the complexities of cloud data center security and evaluate the effectiveness of our proposed hybrid encryption solution. This method is designed to increase clarity and transparency and allow replication and validation of our research findings. The key components of our research method are as follows: Problem identification, model design, implementation, testing, security analysis, comparison with existing methods, data analysis and presentation of results.

In applying this systematic methodology, our aim was to ensure the robustness and validity of our research. The step-by-step approach brought clarity to our research process and facilitated the evaluation of the effectiveness of our proposed hybrid encryption solution in increasing security and efficiency in cloud data centers.

A. *Combination of two encryption algorithms*

Fig. 2 displays the overall architecture of the suggested method for integrating two techniques employing RSA asymmetric and AES symmetric encryption.

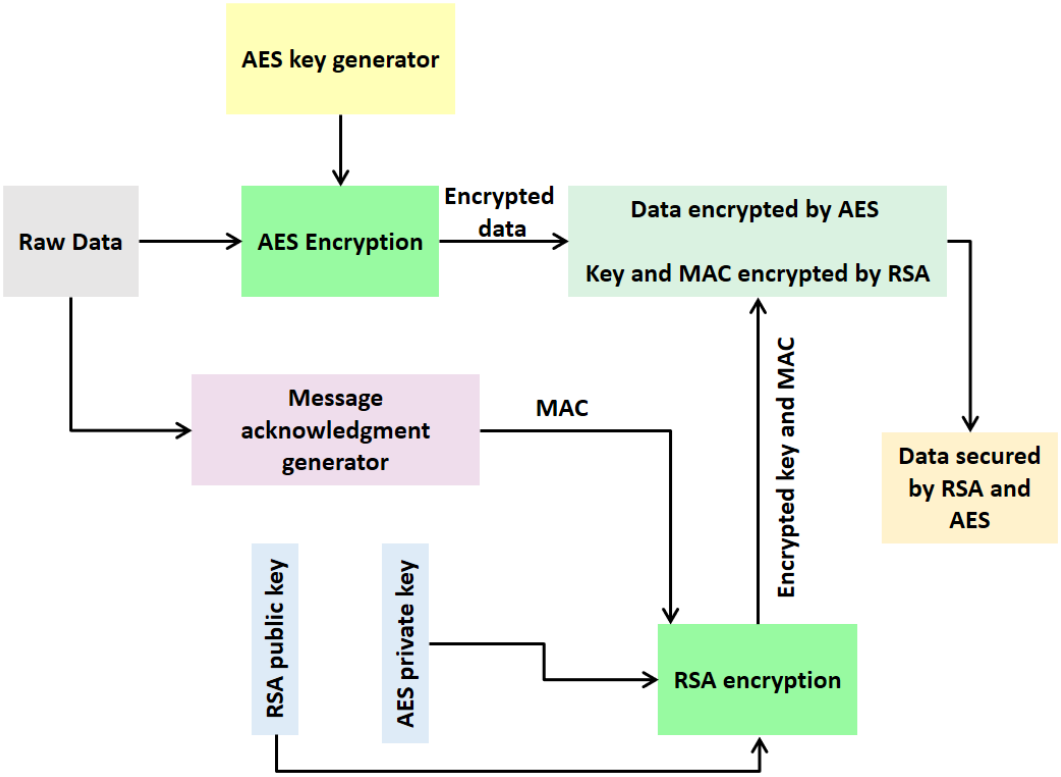

Fig. 2. The general framework of the combination of two cryptographic algorithms.

Proposed hybrid algorithm

Input: Data [] array of n integers storing data for the protection section.
Where Data is an integer array containing the nonnegative integers A, C, I, AES and RSA.
Output: Information sorted by category for the relevant section.
For a to x
C [a] = value of confidentiality.
I [a]= value of integrity
A [a]= value of availability
Calculate AES and RSA [i] = (s [a] + (1/v[a])*10+I [a])/2
For b= 1 to 10
For a= 1 to x
If AES and RSA [a] == 1, 2, 3 then
S [a] = 3
If AES and RSA [a] == 4, 5, 6 then
S [i] = 2
If AES and RSA [a] == 8, 9, 10 then
S [a] = 1

Since cloud storage is preferred, methods are provided for storing various types of data there (public, private, restricted access) according to three cryptographic parameters: privacy, accessibility, and security. The value of "C" (Confidentiality) is determined by how much privacy is needed at each data processing step. In contrast, the value of "I" (Integrity) is determined by how well the data is accurate, reliable, and protected against unauthorized changes. Fig. 1's A (Availability) represents how readily available the data needs to be in response to a user's request.

Users of the aforementioned algorithm are tasked with categorizing information according to the three cryptographic parameters C, I, and A. The user is responsible for providing the values for C, I, and A, where D_{ata} [] is the data. Integrity is also directly tied to security and secrecy, while security is inversely related to availability. This "SR" number determines which of Fig 2's three sections the information belongs in S3 [Public], S2 [Private], or S1 [Owner's Limited Access].

In this solution, the interaction between the user and the cloud servers is considered the main step in which the user must be registered as a cloud client. If he is a registered user, the password check process is performed. Otherwise, you must first register using the service provider, and after confirming the user's authentication process, his public key will be sent to the server to encrypt the data. The scheduling unit of runtime coordination can be categorized as follows from the standpoint of runtime scheduling granularity:

- The complete application framework, which consists of all the entities that are used for execution and collaboration as well as the required external containers. Take a Hadoop , for instance.
- An instance of an application workload, comprising all the entities involved in distributed execution, is referred to as an application instance. Consider a Hadoop job.
- A single executor, which is typically a local OS process, is the internal execution entity of an application instance. Take a Hadoop task, for instance.

IV. Evaluation Results

These tests assess the effectiveness of data encryption at various sizes. Fig. 3 to 11 and Tables I to III exhibit the experiment's outcomes. The differences between the implementation of cryptographic operations in the combined mode and in comparison, with other methods may be readily recognized through the evaluation of the graphs. The comparison of the combined mode's execution times clearly demonstrates improved efficiency and optimal execution. In reality, the encryption procedure has been completed in half the time required by the RSA technique, making the execution time more than two times faster. The high execution time of asymmetric coding techniques is actually one of their key issues, which has a significant impact on performance. Although the throughput is somewhat less optimized when compared to the Blowfish method, the execution time is almost 30 milliseconds less, demonstrating that the combined solution is more optimal.

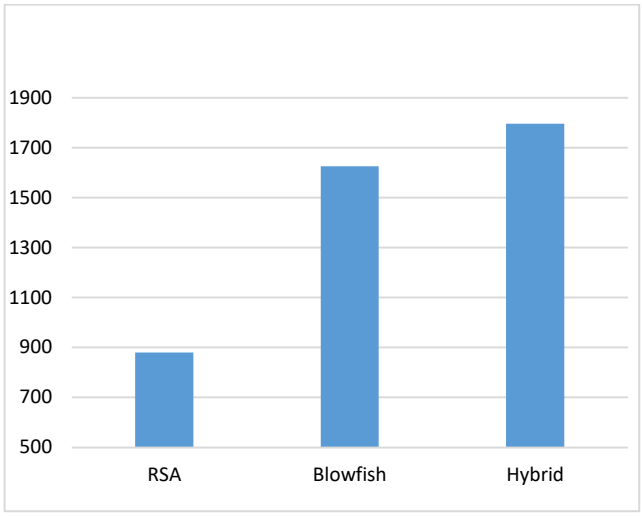

Fig. 3. Throughput - kilobytes per second (data size: 512 kilobytes).

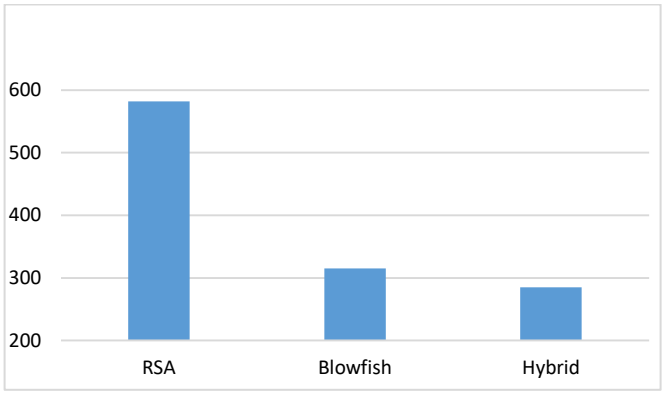

Fig. 4. Execution time - milliseconds (data size: 512 KB).

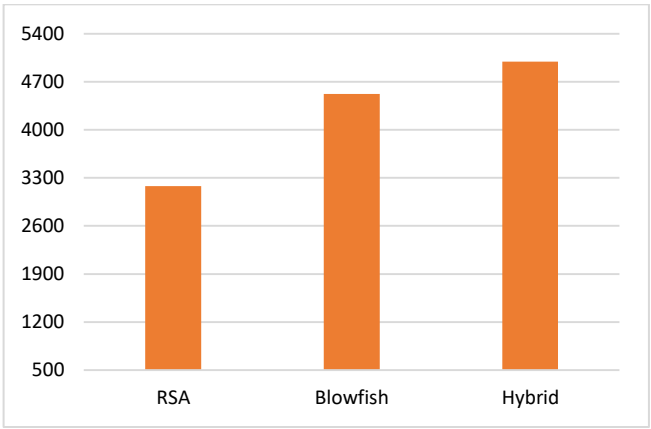

Fig. 5. Throughput - kilobytes per second (data size: 2048 kilobytes).

TABLE I. Execution Time and Throughput (Data Size: 512 KB)

	Hybrid	Blowfish	RSA
Throughput	1796	1625	880
Execution time (milliseconds)	285	315	582

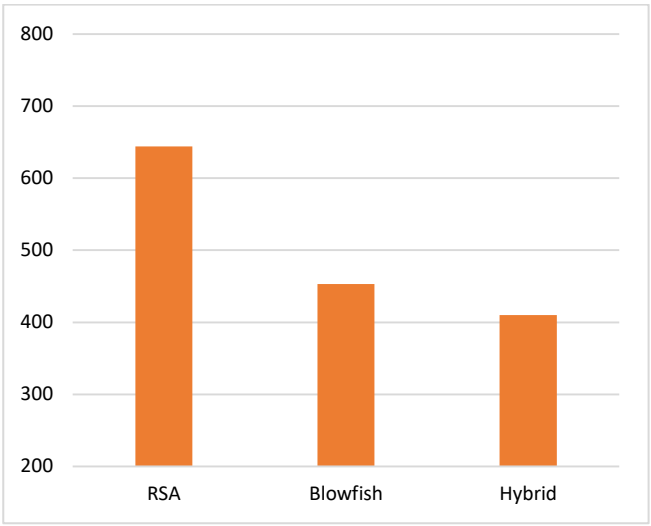

Fig. 6. Execution time - milliseconds (data size: 2048 KB).

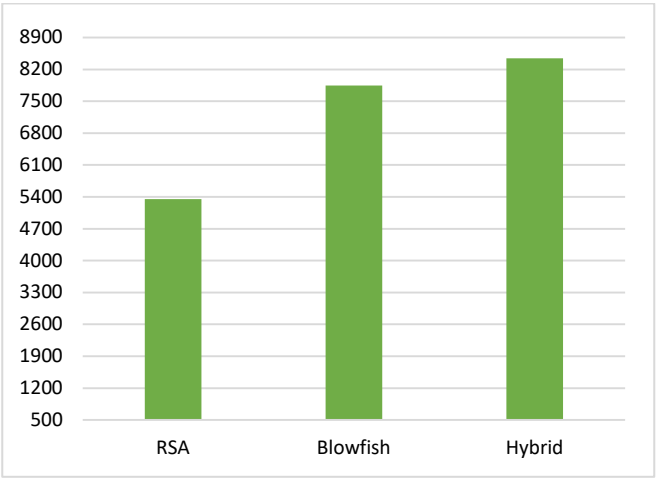

Fig. 7. Throughput - kilobytes per second (data size: 4096 kilobytes).

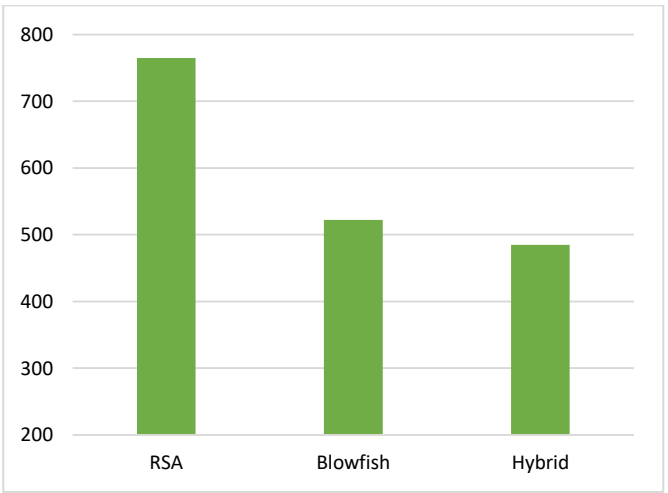

Fig. 8. Execution time - milliseconds (data size: 4096 KB).

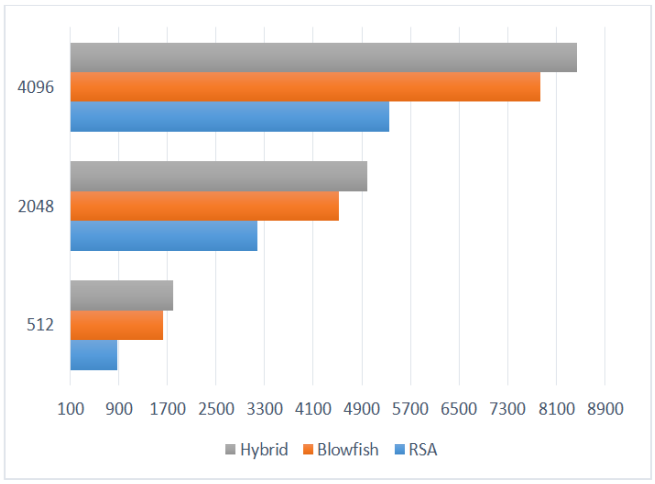

Fig. 9. Throughput rate in different data sizes - kilobytes per second.

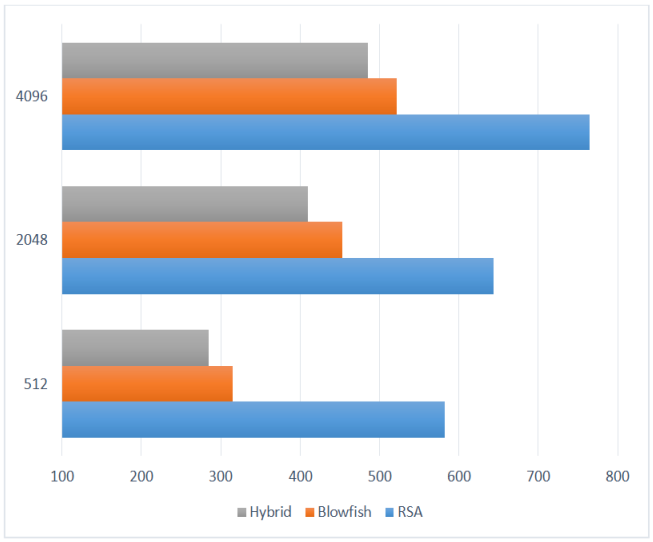

Fig. 10. Execution time in different data sizes - in milliseconds.

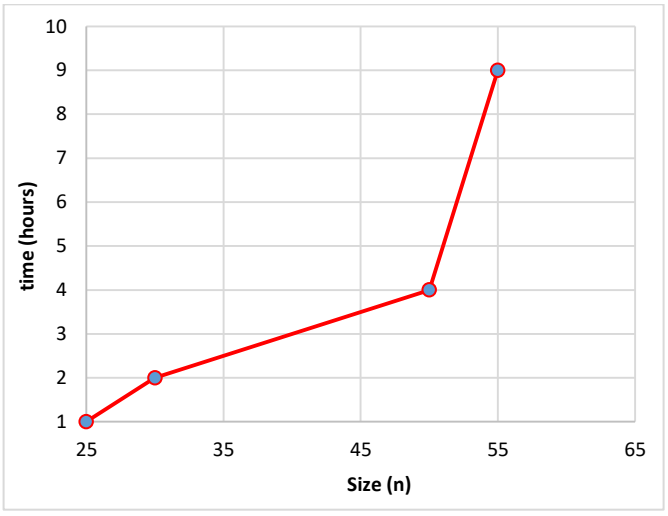

Fig. 11. The time required to decompose n into prime factors.

Due to the fact that the size of the data that can be encrypted can affect the time of the operation, in the second test, the size of the data has been increased to two megabytes (2048 kilobytes) to compare the efficiency of algorithms in this volume of data. As can be seen from Table II, the proposed solution performs much better than the RSA algorithm. In addition to the better throughput, the execution time is reduced by about 214 milliseconds. Compared to the Blowfish algorithm, the execution time has decreased by about 43 milliseconds and at the same time, the transmittance has also increased. This shows that the proposed solution works faster than these two solutions.

TABLE II. RUNNING TIME AND THROUGHPUT (DATA SIZE: 2048 KB)

	Hybrid	**Blowfish**	**RSA**
Throughput	4995	4521	3180
Execution time (milliseconds)	410	453	644

TABLE III. EXECUTION TIME AND THROUGHPUT (DATA SIZE: 4096 KB)

	Hybrid	**Blowfish**	**RSA**
Throughput	8445	7847	5354
Execution time (milliseconds)	485	522	765

Charts 9 and 10 clearly demonstrate that the proposed solution is more optimal in the last test, which compared the desired algorithms with a data amount of 4096 kilobytes. The proposed solution was able to perform the operation in less time than the other two algorithms, as shown in Table III. As the amount of data increases, the throughput likewise increases, but it impacts the encryption time and results in more time required.

The RSA technique is asymmetric, so it naturally takes a lot of time to produce the public and private keys and perform the encryption process, so its execution time is significantly higher. However, the other solution takes less time because of its symmetry. The results also demonstrate how little implementation there is.

Finally, in Fig. 9 and 10, a general comparison between execution times and throughput in different data sizes can be seen.

As can be seen, the processing action requires greater time for the initial execution than for successive executions, which indicates the necessity for warming up or preparation time. This time will be shorter for the application of larger data, yielding the desired outcome. The block size of processable threads was set at 256 for the processing resources that will be accessible for cryptographic operations, albeit this number may change depending on the resources' makeup. The greatest quantity of data that can be processed at any given time for encryption and decryption operations will be equivalent to 256 gigabytes. On the other hand, each processable thread can hold 64 bits of data (8 bytes).

A. Time of Failure

This section evaluates the failure time of the suggested solution because, in addition to boosting efficiency, increasing security is one of the key goals of this research. Failure time is the amount of time needed to locate the algorithm's secret key and subsequently decrypt it. It is obvious that the answer is more secure the longer this period is. Although all encryption techniques can indeed be defeated, what matters is how long the information should be decrypted and what tools should be used. In relation to the suggested hybrid approach, since the RSA technique encrypts the AES encryption key, if the RSA can be decrypted, the AES key may also be retrieved, allowing for the discovery of the original content.

It should be noted that there are only a few ways to decrypt text using the RSA algorithm, the primary one being the breakdown of n into prime factors. In this instance, n and e are likewise presumed to be provided together with the RSA public key. Decomposing n into its prime factors, p and q, is the initial step at this point. Actually, the main and most challenging element of decrypting an RSA key is this step. Mathematical computations have demonstrated that it would take more than seven months to find the prime factors of a number with 155 digits, for instance, even using the fastest computers. The crucial point is that by choosing a larger key (choosing larger p and q numbers) when employing RSA, the work of analyzing n can be made much more challenging for new computers, regardless of how fast and adept they are at handling huge numbers. The outcomes of the decomposition of various values of n into prime components, along with the time needed to complete it, are presented.

According to evaluations, if the number of digits is taken to be equal to 200, four million years of time, as can be seen in the above figure, the time of decomposition might increase exponentially the greater the value of n. The ability to decode the data is required. Fig. 11 makes it evident that picking larger digits can actually make the analysis time; as a result, the RSA decryption is unfeasible in a short amount of time.

B. Efficiency Comparison

An effective cloud data security model will solve all cloud computing's potential issues, allowing its advantages to soar to new heights while shielding its owner's data from as many threats as feasible. In Table IV, we can see how the suggested model stacks up against competing approaches to data protection.

After the security parameters for the contrasted approaches were implemented, Fig. 12 depicts the level of security. As can be seen, the security factor of the suggested method is higher compared to previous ways due to the rise in the volume of data in the horizontal axis, as well as the usage of data classification and encryption technology.

TABLE IV. FACTORS OF THE PROPOSED METHOD COMPARED TO SIMILAR WORKS

Factors	[3]	[31]	[9]	[13]	[21]	[24]	[25]	[29]	This work
Authentication of Storage Providers	n	y	y	y	n	n	y	n	y
Confidentiality	n	y	n	y	y	n	y	n	y
Non-repudiation	n	n		y	y	y	n	n	y
Safe even if the user's credentials are compromised	y	n	y	n	n	y	n	y	y
Authorization	y	y	y	n	n	y	y	y	y
Encryption	y	n	y	y	y	y	n	n	y
Identifying and verifying	y	y	n	n	n	n	y	y	y
Integrity	n	n	y	n	n	n	y	n	y
File indexing	y	y	n	y	y	n	n	y	y
Lookup by Keywords	n	n	y	n	y	y	n	y	y

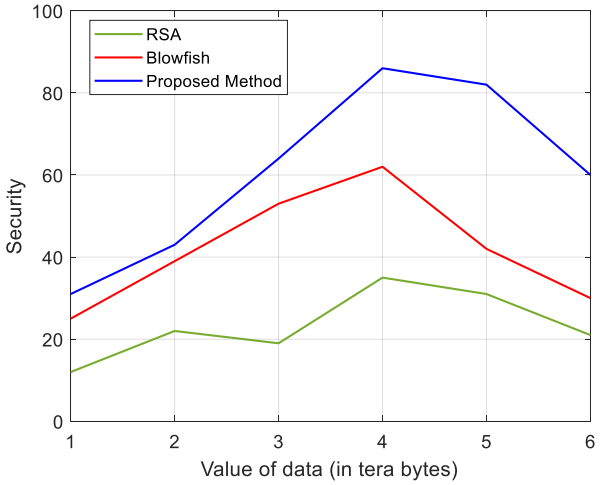

Fig. 12. Comparison of methods for security assessments.

This research includes a hybrid approach using both RSA and AES algorithms for encryption in cloud data centers, which is summarized as follows:

1) Execution time and throughput analysis: 512 KB data size: The proposed hybrid solution performs significantly better than the RSA algorithm when encrypting 512 KB data. In particular, it notes that the execution time is cut in half compared to RSA, which represents a significant improvement in speed. Furthermore, although the throughput is slightly lower than the Blowfish method, it is noted that the execution time is approximately 30 milliseconds shorter, indicating that the hybrid solution offers a more optimal balance between speed and security.

2048 KB data size: When the data size increases to 2048 KB (2 MB), the hybrid solution still outperforms RSA by a large margin. The execution time is reduced by approximately 214 ms compared to RSA, indicating that the hybrid approach is significantly faster. The article also mentions improvements in throughput.

4096 KB data size: In the final evaluation with a data size of 4096 KB (4 MB), the hybrid solution still shows its superiority over RSA and Blowfish. Runtimes are reported to be approximately 280ms faster than RSA and 37ms faster than Blowfish, showing consistent and significant speed benefits.

2) Breakdown time analysis: In this research, an analysis of the RSA algorithm's failure time is presented, which is a measure of how long it takes to decrypt RSA-encrypted data. This emphasizes the importance of choosing large key sizes to make computational decryption impossible in a reasonable amount of time. The presented results show that with a large enough key size (e.g., 200 digits), the decryption time can increase to millions of years, which highlights the security advantage of using larger key sizes in RSA encryption.

3) Efficiency comparison: In the following, the proposed hybrid model has been compared with other methods in terms of security evaluation. It uses a security factor as a benchmark and shows that the hybrid model consistently shows higher security levels compared to alternatives. This suggests that the hybrid approach provides a superior balance between security and efficiency.

The proposed method points out that the hybrid approach combines the strengths of AES and RSA for cryptographic parameters, which results in ensuring confidentiality, integrity, and availability. While providing runtime and throughput data, it is helpful to include specific numerical results, charts, or graphs to visually demonstrate these performance improvements.

V. CONCLUSION

In addition to ensuring security, this article's major objective is to make encryption and decryption processes more effective. As a result, a hybrid approach based on the RSA and AES algorithms was presented to speed up encryption procedures while boosting security. As a result, the message confirmation code and the RSA method are used to secure the key after the main data has been encrypted using the AES technique. With this technique, the potential for decreasing the time required for the encryption procedure is also well established, in addition to improving security. Finally, a hybrid algorithm was constructed and tested using the Java programming language and the Eclipse programming environment, and all tests conducted on the provided solution show that it is superior to alternative approaches; there were strategies. Therefore, the encryption procedure was completed in half the time required by the RSA technique in the first test with a data volume of 512 kilobytes. Although the throughput was higher than the Blowfish method, the execution time was almost 30 milliseconds lower, demonstrating the combined solution's superiority. The proposed approach outperformed the

RSA technique in the second trial, which used a data volume of 2048 KB. Compared to the Blowfish algorithm, throughput increased more than two times, and execution time dropped by more than 214 milliseconds.

Additionally, the throughput is higher, and the execution time is decreased by 43 milliseconds, showing that the proposed method operates more quickly than the two alternatives. The proposed solution outperformed the other two algorithms in the final evaluation with a data volume of 4096 kilobytes, resulting in execution times that were about 280 milliseconds faster than those of the RSA algorithm and 37 milliseconds faster than those of the Blowfish algorithm, respectively. In comparison to these three methods, it demonstrates total optimality. The failure time of the RSA method, which is in charge of protecting the private key, was also examined to assess the solution's level of security. The findings of the experiment demonstrate that by using large keys, it can be decoded in a fair amount of time.

It is feasible to add to the described algorithm for future research and to broaden the suggested solution; it also makes use of a trusted third-party (TTP) based model because TTPs are frequently used in both commercial and cryptographic digital transactions, particularly in those involving a CA that gives a digital identity certificate to one of the two parties. Accordingly, the user is first verified using TTP-based protocols and then, after receiving the authentication certificate, can access the system and view and decode the encrypted data. At that point, the CA becomes a trusted third party for issuing certificates.

ACKNOWLEDGMENT

Anhui Province Scientific research project "Research on the Innovative Development Path of Anhui Cross-border E-commerce under the Background of Free Trade Zone Construction" Project No.: SK2021A0814.

REFERENCES

[1] M. Kaur and R. Aron, “Energy-aware load balancing in fog cloud computing,” Mater Today Proc, 2020.

[2] O. Y. Abdulhammed, “Load balancing of IoT tasks in the cloud computing by using sparrow search algorithm,” J Supercomput, vol. 78, no. 3, pp. 3266–3287, 2022.

[3] M. M. S. Maswood, M. D. R. Rahman, A. G. Alharbi, and D. Medhi, “A novel strategy to achieve bandwidth cost reduction and load balancing in a cooperative three-layer fog-cloud computing environment,” IEEE Access, vol. 8, pp. 113737–113750, 2020.

[4] M. Trik, S. P. Mozaffari, and A. M. Bidgoli, “Providing an adaptive routing along with a hybrid selection strategy to increase efficiency in NoC-based neuromorphic systems,” Comput Intell Neurosci, vol. 2021, 2021.

[5] Fabin Cheng, Ben Niu, Ning Xu, Xudong Zhao, and Adil M. Ahmad. Fault Detection and Performance Recovery Design With Deferred Actuator Replacement Via A Low-Computation Method, IEEE Transactions on Automation Science and Engineering, DOI: 10.1109/TASE.2023.3300723, 2023

[6] A. Celesti, M. Fazio, A. Galletta, L. Carnevale, J. Wan, and M. Villari, “An approach for the secure management of hybrid cloud–edge environments,” Future Generation Computer Systems, vol. 90, pp. 1–19, 2019.

[7] Ning Xu, Zhongyu Chen, Ben Niu, and Xudong Zhao. Event-Triggered Distributed Consensus Tracking for Nonlinear Multi-Agent Systems: A Minimal Approximation Approach, IEEE Journal on Emerging and Selected Topics in Circuits and Systems, DOI: 10.1109/JETCAS.2023.3277544, 2023.

[8] M. K. Neha, “Enhanced security using hybrid encryption algorithm,” International Journal of Innovative Research in Computer and Communication Engineering, vol. 4, no. 7, pp. 13001–13007, 2016.

[9] Arefanjazi, H., Ataei, M., Ekramian, M., & Montazeri, A. (2023). A robust distributed observer design for Lipschitz nonlinear systems with time-varying switching topology. Journal of the Franklin Institute, 360(14), 10728-10744.

[10] P. Crocker and P. Querido, “Two factor encryption in cloud storage providers using hardware tokens,” in 2015 IEEE Globecom Workshops (GC Wkshps), IEEE, 2015, pp. 1–6.

[11] S. Guo, X. Zhao, H. Wang, N. Xu, Distributed consensus of heterogeneous switched nonlinear multiagent systems with input quantization and dos attacks, Applied Mathematics and Computation 456 (2023) 128127.

[12] B. Seth, S. Dalal, and R. Kumar, “Hybrid homomorphic encryption scheme for secure cloud data storage,” Recent Advances in Computational Intelligence, pp. 71–92, 2019.

[13] Wenjing Wu, Ning Xu, Ben Niu, Xudong Zhao and Adil M. Ahmad, Low-Computation Adaptive Saturated Self-Triggered Tracking Control of Uncertain Networked Systems, Electronics, 12(13), 2771, 2023.

[14] M. Samiei, A. Hassani, S. Sarspy, I. E. Komari, M. Trik, and F. Hassanpour, “Classification of skin cancer stages using a AHP fuzzy technique within the context of big data healthcare,” J Cancer Res Clin Oncol, pp. 1–15, 2023.

[15] Chen Cao, Jianhua Wang, Devin Kwok, Zilong Zhang, Feifei Cui, Da Zhao, Mulin Jun Li, Quan Zou. webTWAS: a resource for disease candidate susceptibility genes identified by transcriptome-wide association study. Nucleic Acids Research.2022, 50(D1): D1123-D1130.

[16] J. Sun, Y. Zhang, and M. Trik, “PBPHS: a profile-based predictive handover strategy for 5G networks,” Cybern Syst, pp. 1–22, 2022.

[17] M. Trik, H. Akhavan, A. M. Bidgoli, A. M. N. G. Molk, H. Vashani, and S. P. Mozaffari, “A new adaptive selection strategy for reducing latency in networks on chip,” Integration, vol. 89, pp. 9–24, 2023.

[18] Haoyan Zhang, Xudong Zhao, Huangqing Wang, Ben Niu, Ning Xu, Adaptive Tracking Control for Output-Constrained Switched MIMO Pure-Feedback Nonlinear Systems with Input Saturation, Journal of systems science & complexity, 36: 960–984, 2023.

[19] Abouzarkhanifard, A., Chimeh, H. E., Al Janaideh, M., & Zhang, L. (2023). Fem-inclusive transfer learning for bistable piezoelectric mems energy harvester design. IEEE Sensors Journal, 23(4), 3521-3531.

[20] M. Trik, A. M. N. G. Molk, F. Ghasemi, and P. Pouryeganeh, “A hybrid selection strategy based on traffic analysis for improving performance in networks on chip,” J Sens, vol. 2022, 2022.

[21] Wang, Z., Jin, Z., Yang, Z., Zhao, W., & Trik, M. (2023). Increasing efficiency for routing in Internet of Things using Binary Gray Wolf Optimization and fuzzy logic. Journal of King Saud University-Computer and Information Sciences, 101732.

[22] H. Abroshan, “A hybrid encryption solution to improve cloud computing security using symmetric and asymmetric cryptography algorithms,” International Journal of Advanced Computer Science and Applications, vol. 12, no. 6, pp. 31–37, 2021.

[23] C. Kota, “Secure File Storage in Cloud Using Hybrid Cryptography,” Available at SSRN 4209511, 2022.

[24] V. Goyal and C. Kant, “An effective hybrid encryption algorithm for ensuring cloud data security,” in Big Data Analytics: Proceedings of CSI 2015, Springer, 2018, pp. 195–210.

[25] A. Kumar, V. Jain, and A. Yadav, “A new approach for security in cloud data storage for IOT applications using hybrid cryptography technique,” in 2020 international conference on power electronics & IoT applications in renewable energy and its control (PARC), IEEE, 2020, pp. 514–517.

[26] Khalafi, M., & Boob, D. (2023, July). Accelerated Primal-Dual Methods for Convex-Strongly-Concave Saddle Point Problems. In International Conference on Machine Learning (pp. 16250-16270). PMLR.

[27] P. B. Regade, A. A. Patil, S. S. Koli, R. B. Gokavi, and M. S. Bhandigare, “SURVEY ON SECURE FILE STORAGE ON CLOUD USING HYBRID CRYPTOGRAPHY,” International Research Journal of

Modernization in Engineering Technology and Science, vol. 4, no. 06, 2022.

[28] S. Gokulraj, P. Ananthi, R. Baby, and E. Janani, "Secure File Storage Using Hybrid Cryptography," Available at SSRN 3802668, 2021.

[29] S. Rehman, N. Talat Bajwa, M. A. Shah, A. O. Aseeri, and A. Anjum, "Hybrid AES-ECC model for the security of data over cloud storage," Electronics (Basel), vol. 10, no. 21, p. 2673, 2021.

[30] J. Lei, Q. Wu, and J. Xu, "Privacy and security-aware workflow scheduling in a hybrid cloud," Future Generation Computer Systems, vol. 131, pp. 269–278, 2022.

[31] L. Huang, K. Feng, and C. Xie, "A practical hybrid quantum-safe cryptographic scheme between data centers," in Emerging Imaging and Sensing Technologies for Security and Defence V; and Advanced Manufacturing Technologies for Micro-and Nanosystems in Security and Defence III, SPIE, 2020, pp. 30–35.