The CSI Journal on Computer Science and Engineering

Journal homepage: www.csionjcse.ir

Improving Data Protection in Cloud Environments via Multilayered Cryptographic Mechanisms

Alireza Chamkoori[1*], Hassan Dirsanji[2]

[1] Department of Computer Engineering, Bu.C., Islamic Azad University, Bushehr, Iran
[2] Department of Computer Engineering, Bu.C., Islamic Azad University, Bushehr, Iran

*** Corresponding author email address**: chamkoori@iau.ac.ir

A r t i c l e I n f o

Article type:
Original Research

How to cite this article:
Berihun, A. M., Jemere, Z., Admassu, B., Abebe, W., Kassa, S., Wondie Mekonen, A., Takele, D., Getahun Feleke, M., Berrie, K., Mekasha, Y. & Demessie, Y. (2026). Improving Data Protection in Cloud Environments via Multilayered Cryptographic Mechanisms. *The CSI Journal on Computer Science and Engineering, 20*(2), 115-123.
http://dx.doi.org/10.22034/jcse.2026.583142.1084

A B S T R A C T

This paper presents a multilayer encryption framework for enhancing security in cloud computing environments. The proposed approach combines AES-256, Twofish-256, and Serpent-256 symmetric encryption algorithms with RSA-based key exchange and SHA-256 hashing to improve confidentiality, integrity, and secure data transmission. The framework applies sequential encryption layers using independent cryptographic keys in order to increase resistance against unauthorized access and cryptanalytic attacks. Data integrity is verified through comparison of the generated hash values before encryption and after decryption. The proposed model was implemented and evaluated using CloudSim and the TON-IoT dataset under different storage conditions. Experimental results indicate that the proposed multilayer mechanism significantly improves the security margin while introducing only limited computational overhead compared with conventional AES-based approaches. The evaluation also demonstrates acceptable quality-of-service performance for cloud applications and secure IoT environments.

Keywords: *AES, Twofish, Serpent, RSA, ECC.*

Article history:
Published online 07 August 2026

1 Introduction

Cloud computing has become an essential platform for large-scale storage, distributed processing, and remote access to digital resources. Despite its scalability and flexibility, cloud environments remain vulnerable to various security threats, including unauthorized access, data leakage, and cyberattacks targeting sensitive information. Protecting cloud data therefore requires strong cryptographic mechanisms capable of preserving confidentiality, integrity, and secure communication between users and cloud infrastructures. [1-3]

Many existing cloud security approaches rely on single-layer encryption techniques, which may become insufficient against modern attack models and large-scale distributed threats. Although algorithms such as AES provide high computational efficiency, depending on a single cryptographic layer may reduce resistance against advanced cryptanalytic attacks and key compromise scenarios. Consequently, recent studies have focused on multilayer and hybrid encryption frameworks to improve security robustness in cloud environments. [4-6]

To address these limitations, this paper proposes a multilayer encryption framework integrating AES-256, Twofish-256, Serpent-256, RSA, and SHA-256 mechanisms. The proposed architecture aims to enhance cloud data security through sequential encryption layers, secure key exchange, and hash-based integrity verification while maintaining acceptable computational overhead and quality-of-service performance.

Encryption is the ability to transform data into an undetectable form, which reduces the risk of unauthorized access, ensuring the integrity, confidentiality and availability of data in the cloud ecosystem [7]. The nature of encryption lies in its ability to transform data into an undetectable format, reduce the risk of unauthorized access, and ensure the integrity, confidentiality and availability of data in a multifaceted cloud ecosystem [8]. Encryption also plays a vital role in security in cloud environments, ensuring that data from different users remains isolated and inaccessible to each other, despite being used on shared infrastructure.

Scientifically, encryption mechanisms, whether symmetric (in which the same key is used to encrypt and decrypt data), or asymmetric (in which a pair of public and private keys are used), convert data into ciphertext, resulting in an impenetrable barrier against attackers to exploit sensitive data [9]. In the cloud platform, this technological strategy is used not only to protect data for data storage, but also to increase system security during data transfer and processing. Also, encryption is used as a basic tool in ensuring compliance with regulations, organizational sustainability, strengthening user trust and increasing the security of cloud service models [10].

Businesses, governments, and individuals need strong encryption, access control, and regular auditing to protect their digital assets from potential breaches, data leaks, or unauthorized manipulations, thereby protecting not only data integrity and confidentiality, but also privacy. They also protect the privacy of institutions and people involved [1]. Secure data in cloud storage is not just a technical necessity, but a comprehensive and multifaceted requirement that combines the aspects of cyber security, legal compliance, ethical considerations, and organizational survival and prosperity in the interconnected digital global ecosystem [11]. In addition, from an economic and reputational point of view, secure data storage is instrumental in preventing the potentially catastrophic consequences of a data breach, which can include financial losses, reputational damage, and legal consequences.

Protecting the valuable asset "data" is necessary to maintain competitive advantage and ensure the continuation of digital transformation efforts [12]. Legal and ethical dimensions also significantly affect the importance of secure cloud data storage, as organizations are increasingly mandated by regulatory frameworks (such as GDPR, HIPAA, etc.) to protect consumer data and Stakeholders to protect, and requires them to use strict security measures and show the necessary care in the safe management of digital information [13].

The goals of this paper include determining their strengths, vulnerabilities and suitability in various cloud computing scenarios. To create a structured framework that assists stakeholders in skillfully selecting and implementing encryption strategies those align with specific cloud data security requirements and operational contexts.

A survey conducted by IDC (International Data Corporation) in August 2018 among senior business managers and IT professionals on the challenges and issues that mainly affect cloud computing performance [4]. The results of this survey showed that while 58% of the general population and 86% of senior business leaders are very excited about the potential of cloud computing, more than 90% of these people are extremely concerned about security,

access and privacy. are The survey results show that compared to other parameters, such as efficiency, availability, integration and cost, security is the main challenge among all the parameters that affect the performance and growth of cloud computing.

The modern era has seen the creation of strong cryptographic standards such as the Advanced Encryption Standard (AES) and has now entered the realm of post-quantum cryptography as the next frontier [5].

In the article done in [6], attention was paid to strengthening the inherent security protocols of cloud storage, especially through searchable encryption models, acknowledging the challenges and limitations caused by individual cloud service provider systems. . The authors introduced a new searchable cryptography scheme designed for a multi-cloud environment that underlies blockchain technology. A multi-cloud system model is defined using a consortium chain to attract multiple cloud service providers for data storage. The proposed scheme ensures that the encrypted documents and directories are stored securely in the Interplanetary File System (IPFS), while the hash value and IPFS address of the documents are archived in the blockchain. This innovative approach facilitates the recovery of outsourced encrypted data based on multiple keywords and also incorporates verification systems to verify the integrity of the recovered files. The security and high performance of this design were proved through theoretical analysis and practical experiments using real world data.

In [14], researchers investigate the pivotal role of updatable encryption in the field of cloud storage, and attribute its importance to its capabilities in providing updatable capabilities for ciphertext data and strengthening defenses against malicious key attacks. The most common encryption approaches are updatable, which mainly use partitioning strategies such as leaky sets or firewalls.

In this paper, we are looking for a comprehensive presentation of different encryption methods that are precisely designed to secure data in cloud environments. The goals of this research include determining their strengths, vulnerabilities and suitability in various scenarios used in cloud computing. To create a structured framework that assists stakeholders in skillfully selecting and implementing encryption strategies that align with specific cloud data security requirements and operational contexts.

In addition, this paper aims to contribute to the scholarly dialogue on cloud data security by identifying potential gaps and future research directions in the field of cloud data encryption, thereby paving the way for innovative approaches that will Expertly guides the evolving challenges in securing cloud environments.

The structure of this paper is as follows: In the second part of this research, we examine encryption methods in cloud computing systems. In the third part, we present the proposed algorithm using a multilayer encryption mechanism to increase the security of cloud computing systems. In the fourth part, we will evaluate the performance of our proposed method. Conclusion and suggestions are presented in the fifth section.

The main contributions of this work are summarized as follows:

• Design of a multilayer encryption framework for cloud computing systems.

• Integration of symmetric and asymmetric cryptographic techniques for secure data transmission.

• Implementation of SHA-256-based integrity verification.

• Performance evaluation using CloudSim and the TON-IoT dataset.

• Comparative analysis between the proposed approach and conventional AES-based encryption methods.

2 Encryption and hash algorithms

2.1 Symmetric encryption

Symmetric cryptography refers to an encryption method in which both parties involved have a common secret key that is used for encryption and decryption operations (Figure 1). A symmetric key scheme can be implemented as a block cipher or stream cipher [15]. A block cipher consists of repeating rounds of basic transformations, which are easy to determine and implement. When these transformations are repeated several times, it becomes difficult to reverse without having the right key. The block cipher is determined by the block size and the key size, on which the security of the algorithm depends. Blocks should not be too large to minimize both the length of the encrypted text and the computation overhead. However, they should not be too small, otherwise they are susceptible to codebook attacks.

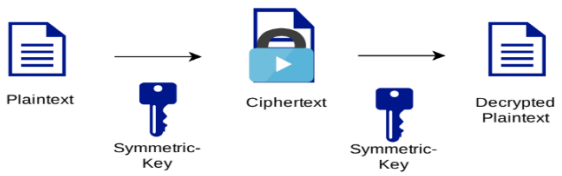

Figure 1. Symmetric key encryption diagram

AES-256 is one of the most widely used symmetric encryption algorithms in modern cloud security systems because of its computational efficiency and strong security characteristics. In the proposed framework, AES is selected as the first encryption layer due to its fast execution speed and hardware acceleration support available in modern processors. The algorithm provides efficient high-throughput encryption while maintaining acceptable computational overhead for cloud environments [16].

Twofish-256 is employed as the second encryption layer in the proposed framework to increase cryptographic diversity and strengthen resistance against brute-force and differential attacks. Compared with conventional single-layer encryption approaches, the integration of Twofish improves the overall security robustness of the proposed multilayer architecture while maintaining acceptable processing performance for cloud environments [16].

Serpent-256 is utilized as the final symmetric encryption layer in the proposed framework because of its high security margin and strong resistance against advanced cryptanalytic attacks. Although the algorithm introduces additional computational overhead compared with AES, its integration significantly enhances the overall confidentiality and robustness of the multilayer encryption architecture for cloud-based environments [17, 18].

Asymmetric encryption

Asymmetric cryptography is applied in the proposed framework to provide secure key exchange and digital signature verification between cloud users and servers. Unlike symmetric encryption methods, asymmetric algorithms utilize separate public and private keys to improve communication security and protect sensitive cryptographic information during data transmission. In the proposed model, asymmetric encryption is mainly used to secure multilayer encryption keys and verify data authenticity across the cloud environment [17].

RSA-2048 is employed in the proposed framework to protect multilayer encryption keys and support secure digital signature verification between the client and cloud server. The asymmetric encryption mechanism improves confidentiality during key exchange and reduces the risk of unauthorized access to cryptographic information during data transmission. In addition, RSA enhances authentication and integrity verification within the proposed cloud security architecture [17].

Elliptic Curve Cryptography (ECC) provides strong security with shorter key lengths and lower computational overhead compared with traditional asymmetric encryption methods. Because of its efficiency, ECC is considered a suitable solution for lightweight cloud and IoT security applications where computational resources are limited [19].

Hash functions

Hash functions are applied in cloud security systems to verify data integrity during storage and transmission processes. In the proposed framework, SHA-256 is used before encryption and after decryption to ensure that the recovered data exactly matches the original plaintext without unauthorized modification. The generated hash values also support integrity verification and secure digital signature validation within the multilayer encryption architecture [20].

3 Proposed model of multilayer encryption in the cloud

The encryption process can occur in different places, but the two main encryption methods are server-side and receiver-side [21]. The distinction between these two is necessary to understand the security of cloud data.

In server-side encryption, the data is encrypted after reaching the infrastructure of the cloud provider. The main advantage of this method is its ease of use for the end user. They simply upload their data and the cloud provider manages the encryption process. However, since encryption is done on the provider side, they have control over the encryption keys. This can be a potential security concern if the provider's systems are compromised.

On the other hand, server-side encryption refers to data that is encrypted at the end of the user before being transmitted to the cloud. This ensures that data is always encrypted during transit and encrypted on the server. One of the notable advantages of this approach is that the encryption keys are usually managed by the user or their organization, ensuring that the cloud provider cannot decrypt the data even if it wants to. However, this added security comes with the responsibility of key management on the part of the user.

Presentation of the proposed method

In the proposed framework, cloud data is protected through a sequential multilayer encryption and decryption mechanism designed to enhance confidentiality, integrity, and secure communication between the client and cloud server. The architecture combines SHA-256 hashing, AES-256, Twofish-256, Serpent-256, and RSA-2048 encryption mechanisms to provide multiple security layers against

unauthorized access and cryptanalytic attacks. Figure 2 illustrates the communication process between the client and server through the cloud communication channel.

To verify data integrity, a SHA-256 hash value is generated from the plaintext before the encryption process begins. The generated hash is digitally signed using the client's RSA private key and transmitted together with the encrypted data. This mechanism enables integrity verification and authentication during the decryption stage at the server side.

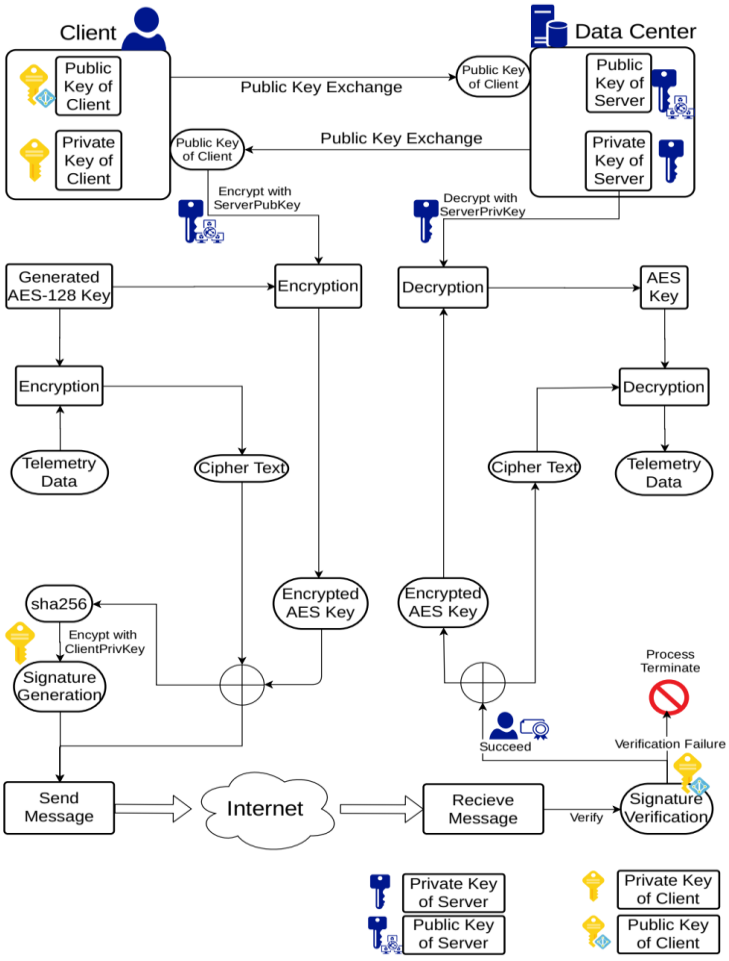

Figure 2. AES+Twofish+serpent and RSA multilayer combination encryption structure

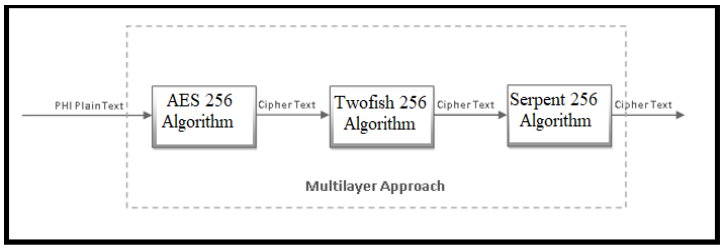

Figure 3. Multi-layer encryption

After hash generation, the plaintext is sequentially encrypted using three independent symmetric encryption layers consisting of AES-256, Twofish-256, and Serpent-256. Each layer utilizes a separate 256-bit cryptographic key in order to increase resistance against brute-force attacks and unauthorized key recovery. The multilayer structure enhances confidentiality by ensuring that compromising a single encryption layer does not expose the original plaintext data.

RSA-2048 public and private key pairs are generated for both the client and server to support secure key exchange and digital signature verification. The server public key is used to encrypt the symmetric encryption keys, while the client private key is applied to digitally sign the generated hash value. During decryption, the server private key decrypts the

symmetric keys, and the client public key verifies the integrity and authenticity of the received data.

Using the RSA key pair to encrypt the hash and keys creates an additional security layer. The encrypted data, hashes, and keys are then either transmitted or securely stored, ensuring that all three are protected during each data transmission or storage process.

The decryption process, which is very important for data analysis, involves using the RSA private key to decrypt the keys and the hash, which is then used to decrypt the data.

The integrity verification process compares the original SHA-256 hash value with the reconstructed hash obtained after decryption. Matching hash values confirm that the transmitted data has not been modified during communication or storage operations within the cloud environment.

4 Implementation and review of the obtained results

The proposed multilayer encryption framework was implemented and evaluated using CloudSim and Java SE Runtime Environment 8 under Windows 10. Experimental analysis was performed using the TON-IoT dataset, which contains heterogeneous cloud and IoT traffic data suitable for evaluating secure cloud storage and communication mechanisms. The simulation environment was configured to analyze encryption performance, computational overhead, and quality-of-service behavior under different storage conditions.

The TON IoT dataset [22], which is a rich and diverse collection of cloud system data, has been used to integrate various sources to facilitate comprehensive analysis in the IoT (Internet of Things) and IIoT (Industrial Internet of Things) domains.

5 Speed comparison of encryption algorithms

Figure 4 compares the encryption and decryption performance of AES, Twofish, Serpent, and the proposed multilayer encryption framework using 200 MB of input data in RAM. The experimental results indicate that AES achieves the highest processing speed because of hardware acceleration support available in modern processors. Although the proposed multilayer architecture introduces additional computational overhead, the increase remains limited compared with the achieved improvement in security robustness.

Algorithm	Encryption	Decryption	Mean
AES	3.4 GB/s	4.0 GB/s	3.7 GB/s
Twofish	628 MB/s	689 MB/s	658 MB/s
AES-Twofish	534 MB/s	591 MB/s	562 MB/s
Serpent	387 MB/s	379 MB/s	383 MB/s
Serpent-AES	348 MB/s	338 MB/s	343 MB/s
Twofish-Serpent	227 MB/s	239 MB/s	233 MB/s
AES-Twofish-Serpent	225 MB/s	226 MB/s	225 MB/s
Serpent-Twofish-AES	218 MB/s	225 MB/s	221 MB/s

Figure 4. Speed comparison of AES, Twofish, Serpent algorithms and their combinations.

For a real comparison of the speed of the algorithms, their implementation was done on a hard drive (HDD) and a solid-state drive (SSD). In practice, most of the data are located on the hard drive or SSD, not in the RAM memory. Therefore, due to the much lower reading and writing speed of drives compared to RAM, the speed of decoding and encryption decreases significantly.

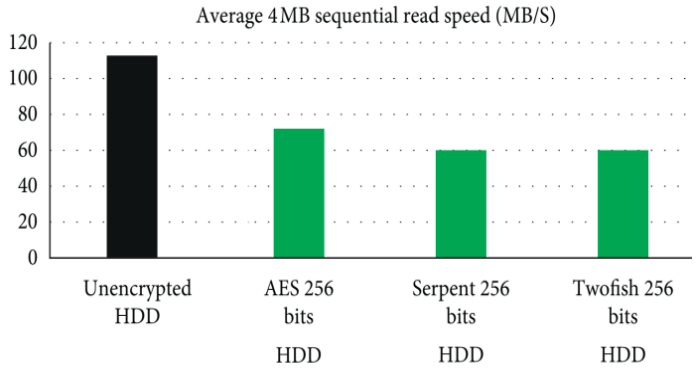

Figure 5. Sequential reading speed of 4MB hard drive and three encryption algorithms AES, Twofish and Serpent

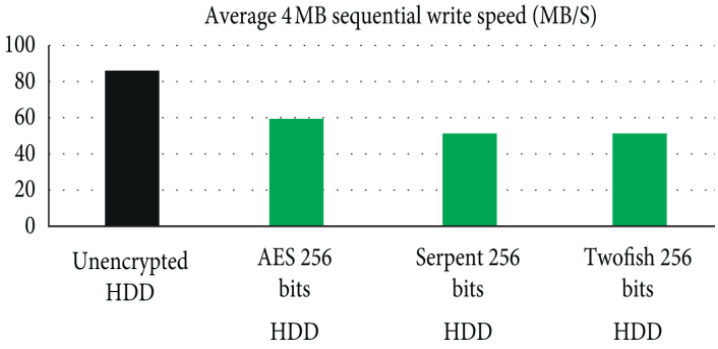

Figure 6. Sequential writing speed of 4MB hard drive and three encryption algorithms AES, Twofish and Serpent

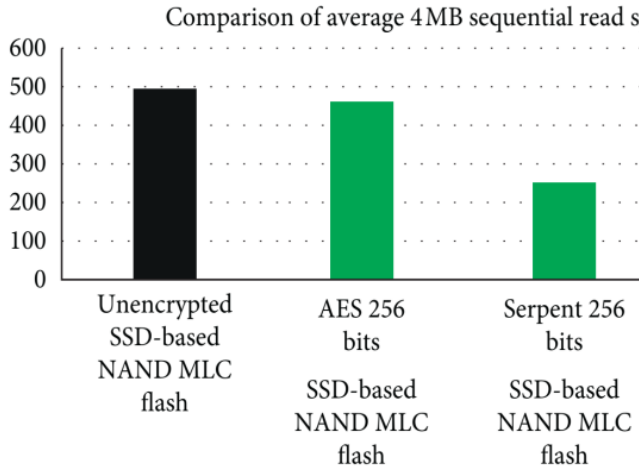

Figure 7. Sequential reading speed of 4MB SSD drive and three encryption algorithms AES, Twofish and Serpent

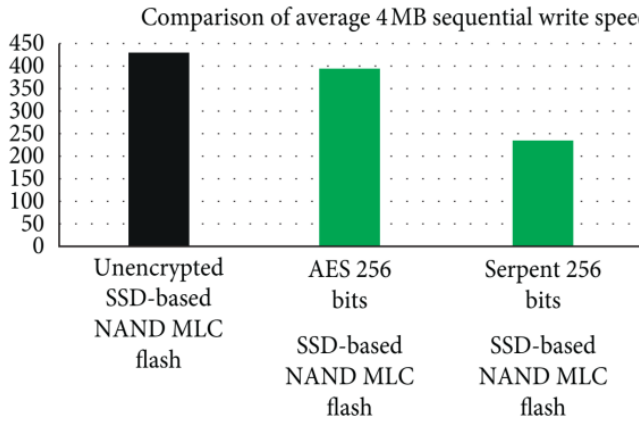

Figure 8. Sequential writing speed of 4MB SSD drive and three encryption algorithms AES, Twofish and Serpent

Figures 5 and 6 present the sequential read and write performance of HDD-based storage under different encryption mechanisms. The results demonstrate that encryption operations reduce storage throughput because of additional cryptographic processing requirements. However, the performance degradation remains within acceptable limits for practical cloud storage environments. AES maintains higher throughput compared with Twofish and Serpent because of optimized hardware-level execution support.

The same process is repeated for the writing mode on the hard drive. In Figure 6, the sequential writing speed of a 4MB hard drive without encryption is 83MBps, which reaches 59MBps, 52MBps, and 51MBps during AES, Twofish, and Serpent encryption, respectively.

Figures 7 and 8 illustrate the read and write performance of SSD-based storage systems under different encryption algorithms. Compared with HDD environments, SSD storage significantly improves encryption throughput because of faster data access operations. The experimental evaluation confirms that the proposed multilayer encryption framework maintains acceptable performance levels while providing enhanced security protection for cloud data storage applications.

In comparison with the SSD, the speed is much higher, and again, due to the high speed of the encryption algorithms by the CPU, especially for AES, there is not much reduction in speed either in the reading or writing mode, and the use of encryption in daily processes is completely justified.

Simulating the CloudSim environment

The proposed framework was further evaluated within the CloudSim environment using file sizes ranging from 100 KB to 50 MB. The simulation included five users and seven distributed data centers with different storage capacities to analyze encryption scalability and communication performance under heterogeneous cloud conditions.

Figure 9. Comparison of the execution time of the proposed multilayer algorithm with the standard AES method

Figure 9 compares the proposed multilayer encryption framework with conventional AES-based encryption in terms of execution time. Although the proposed approach introduces additional processing overhead because of multiple encryption stages, the observed increase remains relatively small compared with the achieved security enhancement. Experimental results demonstrate that the proposed framework preserves acceptable quality-of-service characteristics while significantly improving data protection and resistance against unauthorized access.

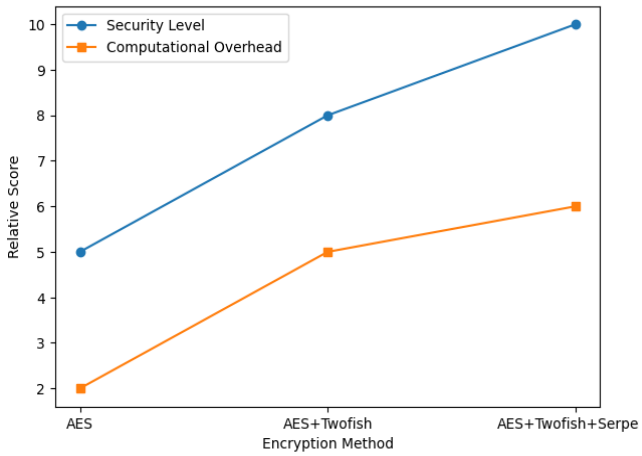

Figure 10. Security level versus computational overhead for different encryption approaches

Figure 10 illustrates the trade-off between security level and computational overhead for different encryption architectures. The proposed multilayer framework achieves significantly higher security robustness compared with conventional AES-based encryption while introducing only moderate additional computational cost.

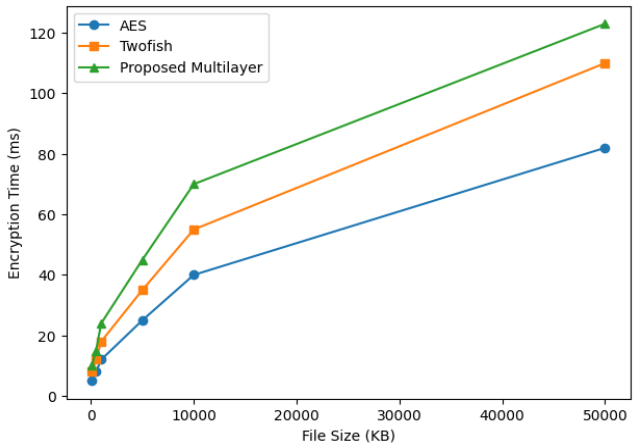

Figure 11. Encryption time scalability under different file sizes

Figure 11 presents the scalability analysis of the proposed encryption framework under different file sizes. The results indicate that encryption time increases gradually with larger datasets while remaining within acceptable performance limits for practical cloud computing applications.

6 Conclusion

This paper presented a multilayer encryption framework for enhancing data security in cloud computing environments. The proposed architecture integrates AES-256, Twofish-256, Serpent-256, RSA-2048, and SHA-256 mechanisms to provide confidentiality, secure key exchange, and integrity verification within cloud communication systems. Experimental evaluation using CloudSim and the TON-IoT dataset demonstrated that the proposed framework significantly improves security robustness while maintaining acceptable computational overhead and quality-of-service performance. Comparative analysis showed that the multilayer approach provides stronger protection than conventional AES-based encryption techniques against unauthorized access and cryptanalytic threats. Future work may focus on lightweight implementations for IoT environments, integration with blockchain-based cloud security systems, and investigation of post-quantum cryptographic mechanisms for next-generation cloud infrastructures.

Ethical Considerations

All procedures performed in this study were under the ethical standards.

Acknowledgments

Authors thank all who helped us through this study.

Conflict of Interest

The authors report no conflict of interest.

Funding

According to the authors, this article has no financial support.

References

[1] M. Ali, "A confidentiality-based data classification-as-a-service (C2aaS) for cloud security," *Alexandria Engineering Journal,* vol. 64, 2023, doi: 10.1016/j.aej.2022.10.056.

[2] K. V. Raipurkar and A. V. Deorankar, "Improve Data Security in Cloud Environment by Using LDAP and Two Way Encryption Algorithm," in *Symposium on Colossal Data Analysis and Networking (CDAN)*, Indore, India, 2016, doi: 10.1109/CDAN.2016.7570934.

[3] G. Yang, "An Efficient Attribute-Based Encryption Scheme with Data Security Classification in the Multi-Cloud Environment," *Electronics,* vol. 12, no. 20, 2023, doi: 10.3390/electronics12204237.

[4] A. Alzahrani, "Hybrid Approach for Improving the Performance of Data Reliability in Cloud Storage Management," *Sensors,* vol. 22, no. 16, 2022, doi: 10.3390/s22165966.

[5] S. G. Domanal, R. M. R. Guddeti, and R. Buyya, "A Hybrid Bio-Inspired Algorithm for Scheduling and Resource Management in Cloud Environment," *IEEE Transactions on Services Computing,* vol. 13, no. 1, 2020, doi: 10.1109/TSC.2017.2679738.

[6] S. Pal, S. Khatua, N. Chaki, and S. Sanyal, "A New Trusted and Collaborative Agent Based Approach for Ensuring Cloud Security," *arXiv preprint arXiv:1108.4100,* 2012, doi: 10.48550/arXiv.1108.4100.
[7] O. Hosam and M. H. Ahmad, "Hybrid design for cloud data security using combination of AES, ECC and LSB steganography," *International Journal of Computational Science and Engineering,* vol. 19, no. 2, 2019, doi: 10.1504/IJCSE.2019.100236.
[8] M. T. Islam, S. Karunasekera, and R. Buyya, "Performance and Cost-Efficient Spark Job Scheduling Based on Deep Reinforcement Learning in Cloud Computing Environments," *IEEE Transactions on Parallel and Distributed Systems,* vol. 33, no. 7, 2022, doi: 10.1109/TPDS.2021.3124670.
[9] L. Malina, J. Hajny, P. Dzurenda, and V. Zeman, "Privacy-preserving security solution for cloud services," *Journal of Applied Research and Technology,* vol. 13, no. 1, 2015, doi: 10.1016/S1665-6423(15)30002-X.
[10] K. K. Almuzaini, "Key Aggregation Cryptosystem and Double Encryption Method for Cloud-Based Intelligent Machine Learning Techniques-Based Health Monitoring Systems," *Computational Intelligence and Neuroscience,* vol. 2022, 2022, doi: 10.1155/2022/3767912.
[11] Y. Harmouch and R. El Kouch, "The benefit of using chaos in key schedule algorithm," *Journal of Information Security and Applications,* vol. 45, 2019, doi: 10.1016/j.jisa.2019.02.001.
[12] A. Shamsuzzoha, "Ensuring Supply Chain Transparency by Deploying Blockchain-Enabled Technology: An Overview With Demonstration," *International Journal of Intelligent Systems,* 2025, doi: 10.1155/int/7304193.
[13] N. H. Mohamad, N. B. Saidin, and M. I. H. Bin Zaidi, "Data Security and Privacy Issues in Cloud Computing: Challenges and Solutions Review," *TechRxiv,* 2023, doi: 10.36227/techrxiv.170327865.59737799/v1.
[14] T. Ali, M. Al-Khalidi, and R. Al-Zaidi, "Information Security Risk Assessment Methods in Cloud Computing: Comprehensive Review," *Journal of Computer Information Systems,* vol. 66, no. 1, 2026, doi: 10.1080/08874417.2024.2329985.
[15] R. Zhang, "The impacts of cloud computing architecture on cloud service performance," *Journal of Computer Information Systems,* vol. 60, no. 2, 2020, doi: 10.1080/08874417.2018.1429957.
[16] M. S. Kumar and M. Pandi, "AI-Enhanced Distributed Multi-Agent Framework for Cloud Security: Real-Time Threat Detection, Automated Policy Enforcement, and Strategic Proactive Defense," 2025, doi: 10.13140/RG.2.2.31919.80807.
[17] H. Abbas, M. S. Hayat Khial, and M. D. Awan, "Comparing the Security and Efficiency of Keyless Data Encryption and Decryption Techniques for Big Data Processing," in *Computational Intelligent Systems*, 2026, doi: 10.1007/978-981-95-2212-5_25.
[18] X. Dong and Y. Xie, "Research on cloud computing network security mechanism and optimization in university education management informatization based on OpenFlow," *Systems and Soft Computing,* vol. 7, 2025, doi: 10.1016/j.sasc.2025.200225.
[19] A. Kalidindi and M. B. Arrama, "Feature selection and hybrid CNNF deep stacked autoencoder for botnet attack detection in IoT," *Computers and Electrical Engineering,* vol. 122, 2025, doi: 10.1016/j.compeleceng.2024.109984.
[20] C. P. Navdeti, I. Banerjee, and C. Giri, "Privacy preservation and secure data sharing scheme in fog based vehicular ad-hoc network," *Journal of Information Security and Applications,* vol. 63, 2021, doi: 10.1016/j.jisa.2021.103014.
[21] K. W. Nafi, T. S. Kar, S. A. Hoque, and M. M. A. Hashem, "A Newer User Authentication, File encryption and Distributed Server Based Cloud Computing security architecture," *International Journal of Advanced Computer Science and Applications (IJACSA),* vol. 3, no. 10, 2012, doi: 10.14569/IJACSA.2012.031029.
[22] R. Ghasemi, "Resolving a common vulnerability in secret sharing scheme-based data outsourcing schemes," *Concurrency and Computation: Practice and Experience,* vol. 35, no. 2, 2020, doi: 10.1002/cpe.5363.