International Journal of Engineering Technology and Applied Science
(ISSN: 2395 3853), Vol. 12 Issue 01 January 2026

Design and Performance Evaluation of a Hybrid Encryption Framework for Secure Cloud Data Storage

[1]Ashish Ranjan, [2]Dr. Jeetendra Singh Yadav
[1]Reseach Scholar, [2]Associate Professor
[12]Department of computer science and engineering, Bhabha university, Bhopal, India
[1]*ashishr57@gmail.com*, [2]*jeetendra2201@gmail.com*

Abstract- Cloud computing provides scalable and cost-effective data storage solutions, but it also introduces significant security challenges such as data breaches, unauthorized access, and insecure key management. To address these issues, this paper proposes a Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS) for secure cloud data storage. The proposed framework integrates Elliptic Curve Cryptography (ECC) for secure key exchange, Dynamic AES (DAES) and DES for multi-layer data encryption, and SHA-3 hashing for data integrity verification. A Random Key Generator (RKG) is used to produce unique encryption keys for each data block, enhancing resistance to brute-force attacks and improving key security. The system is implemented in a simulated client–server cloud environment and evaluated using performance metrics such as encryption time, decryption time, and throughput. Experimental results show that the proposed approach maintains consistent encryption performance across different data sizes while providing enhanced data confidentiality and integrity. Although the multi-layer encryption introduces moderate computational overhead, the proposed framework achieves a balanced trade-off between security and performance, making it suitable for secure cloud storage applications.

Keywords: Cloud Computing, Hybrid Encryption, Cloud Data Security, AES, ECC, SHA-3, Encryption Performance, Secure Data Storage.

I. INTRODUCTION

Cloud computing has emerged as one of the most influential technologies in modern information systems, enabling on-demand access to computing resources such as storage, processing power, and applications over the internet. Organizations across various sectors including healthcare, finance, education, and government increasingly rely on cloud platforms due to their scalability, flexibility, and cost efficiency [1], [2]. Despite these advantages, the migration of sensitive data to cloud environments introduces significant security challenges, particularly concerning data confidentiality, integrity, and secure access control.

In cloud environments, data is typically stored and managed by third-party cloud service providers, which limits the direct control of data owners over their information. This situation exposes sensitive data to various threats such as unauthorized access, insider attacks, data leakage, and cyber intrusions [3], [4]. Consequently, ensuring secure storage and transmission of cloud data has become a critical requirement for maintaining user trust and protecting sensitive information.

Cryptographic techniques are widely adopted as a primary solution to protect cloud data from security threats. Traditional encryption approaches can generally be classified into symmetric encryption and asymmetric encryption methods. Symmetric algorithms such as the Advanced Encryption Standard (AES) provide high computational efficiency for encrypting large volumes of data, while asymmetric algorithms such as RSA and Elliptic Curve Cryptography (ECC) offer secure key distribution and authentication mechanisms [5], [6]. However, each of these approaches has certain limitations when applied independently. Symmetric encryption techniques face challenges related to key management and secure key distribution, whereas asymmetric algorithms often introduce higher computational overhead and processing latency [7].

To overcome these limitations, hybrid encryption techniques that combine symmetric and asymmetric cryptographic mechanisms have gained significant attention in recent years. Hybrid encryption systems utilize symmetric algorithms for efficient data encryption while employing asymmetric algorithms for secure key exchange and access control, thereby achieving improved security and performance balance [8], [9]. Several recent studies have explored hybrid encryption frameworks to enhance cloud security, demonstrating improvements in confidentiality, integrity, and resistance to attacks [10].

However, existing encryption models often suffer from issues such as high computational overhead, inefficient key management, and limited performance evaluation under realistic cloud workloads. Additionally, many approaches focus primarily on data confidentiality while overlooking integrity verification and secure key protection mechanisms [11], [12]. These limitations highlight the

need for a more robust and efficient encryption framework capable of ensuring strong security while maintaining acceptable system performance.

To address these challenges, this paper proposes a **Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS)** for secure cloud data storage and transmission. The proposed framework integrates **Elliptic Curve Cryptography (ECC)** for secure key exchange, **Dynamic Advanced Encryption Standard (DAES) and Data Encryption Standard (DES)** for multi-layer data encryption, and **SHA-3 hashing** for data integrity verification. A **Random Key Generator (RKG)** is incorporated to generate unique encryption keys for each data block, thereby enhancing resistance to brute-force attacks and unauthorized access.

The proposed system is implemented in a simulated cloud environment and evaluated using performance metrics including encryption time, decryption time, and throughput. Experimental results demonstrate that the proposed hybrid encryption framework enhances data security while maintaining acceptable computational performance.

The remainder of this paper is organized as follows. Section II reviews related work in cloud data security and hybrid encryption techniques. Section III presents the proposed methodology and system architecture. Section IV discusses the experimental results and performance analysis. Finally, Section V concludes the paper and outlines potential future research directions.

II. RELATED WORK

Cloud data security has attracted significant attention from researchers due to the rapid growth of cloud-based storage and computing services. Various cryptographic techniques, secure data-sharing mechanisms, and hybrid encryption models have been proposed to protect sensitive data stored in cloud environments.

Modalavalasa *et al.* proposed a privacy-preserving encryption framework that integrates the Advanced Encryption Standard (AES) with SHA-3 hashing to enhance confidentiality and data integrity in cloud environments [1]. Their system was implemented using a simulated cloud infrastructure and evaluated based on encryption time, execution time, and decryption time. Experimental results demonstrated that AES provides efficient encryption performance for large data volumes, while SHA-3 ensures strong integrity protection.

Dewangan *et al.* presented a secure data-sharing framework for federated cloud environments by combining attribute-based encryption (ABE), homomorphic encryption, and blockchain-based access control [2]. The proposed architecture was tested across multiple cloud platforms and demonstrated improved access control and auditability. Experimental evaluation showed that the system could support large numbers of concurrent users while maintaining secure and reliable data access.

Vimalnath *et al.* developed a secure cloud architecture that integrates machine learning techniques with fault injection simulation to detect abnormal behavior in cloud systems [3]. Their framework uses deep learning models such as Long Short-Term Memory (LSTM) and XGBoost for anomaly detection in cloud system logs. The proposed system achieved high detection accuracy and significantly reduced recovery time during fault scenarios.

Gayathri and Sathish Kumar introduced a fog-cloud-based architecture for secure sharing of Personal Health Records (PHRs) using Attribute-Based Encryption (ABE) [4]. The proposed approach provides fine-grained access control while reducing latency by utilizing fog nodes. Experimental results demonstrated improved memory efficiency and secure access management for healthcare data stored in cloud environments.

Reddy *et al.* proposed a data anonymization framework for secure cloud data storage using a hybrid encryption strategy that combines symmetric encryption and RSA-based key protection [5]. The framework also incorporates digital signatures to verify data integrity. Experimental results confirmed that the system successfully protects sensitive data while maintaining accuracy during anonymization and de-anonymization processes.

Sharma *et al.* introduced a blockchain-enabled cloud storage framework that integrates Ethereum-based smart contracts with cloud storage services to ensure secure data sharing and transparent access control [6]. The architecture provides decentralized trust management and tamper-proof logging of data access events, which significantly enhances accountability and transparency in cloud environments.

Fnu and Murri proposed an Automated Encryption/Decryption System (AEDS) that combines multiple cryptographic techniques including Elliptic Curve Cryptography (ECC), Dynamic AES, DES, and SHA-3 hashing to improve cloud data security [7]. Their experimental results showed that hybrid encryption systems provide stronger resistance to brute-force attacks and unauthorized access, although they may introduce moderate computational overhead compared to conventional encryption methods.

Gupta *et al.* conducted a comprehensive survey on secure data storage and sharing techniques in cloud computing environments [8]. Their study categorized existing solutions into cryptography-based, access-control-based, and privacy-preserving approaches. The authors highlighted that hybrid encryption and access control

mechanisms are among the most effective strategies for ensuring cloud data confidentiality and integrity.

Although significant progress has been made in cloud data security, existing solutions still face several limitations such as high computational complexity, inefficient key management, and limited performance evaluation under realistic cloud workloads. These challenges highlight the need for an efficient hybrid encryption framework that provides strong security while maintaining acceptable computational performance.

To address these issues, the present work proposes a **Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS)** that integrates symmetric encryption, asymmetric encryption, and cryptographic hashing techniques to enhance cloud data security and system performance.

III. PROPOSED METHODOLOGY

This section presents the proposed **Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS)** designed to enhance data security in cloud storage environments. The proposed methodology integrates multiple cryptographic techniques including **Elliptic Curve Cryptography (ECC), Dynamic Advanced Encryption Standard (DAES), Data Encryption Standard (DES), and SHA-3 hashing** to ensure confidentiality, integrity, and secure key management. The objective of the proposed framework is to provide **multi-layer data protection while maintaining acceptable computational performance** in cloud-based systems. The proposed model employs symmetric encryption algorithms for efficient data encryption and asymmetric encryption for secure key exchange.

A. System Architecture

The architecture of the proposed AEDS framework consists of four major modules operating in a client–server cloud environment. These modules collectively perform secure data encryption, key management, integrity verification, and data retrieval.

The primary components of the system include:

1. **User Interface Module**
2. **Random Key Generator (RKG)**
3. **Hybrid Encryption Engine**
4. **Cloud Storage Server**

In the proposed architecture, users upload data to the cloud through the client interface. Before storing the data in the cloud server, the system automatically performs encryption using the hybrid cryptographic framework. The **Random Key Generator** produces unique encryption keys for each data block. These keys are then used by the hybrid encryption engine to encrypt the data using a multi-layer encryption strategy. The encrypted data and protected keys are finally stored in the cloud storage server. During data retrieval, the system performs the reverse process where encrypted keys are first decrypted using ECC, followed by the decryption of the data using DAES and DES.

B. Hybrid Encryption Framework

The proposed methodology combines symmetric and asymmetric cryptographic algorithms to achieve improved security and efficiency. The hybrid encryption framework consists of the following components.

1. Dynamic AES (DAES): Dynamic AES is used as the primary symmetric encryption algorithm for securing cloud data. Unlike conventional AES implementations that rely on static substitution tables, the proposed approach employs a **dynamic S-box generation mechanism** based on polynomial functions. The dynamic substitution mechanism increases resistance against cryptanalysis attacks and improves the randomness of encrypted outputs. DAES ensures fast encryption performance for large volumes of cloud data.

2. Data Encryption Standard (DES): DES is incorporated as an additional encryption layer to strengthen data protection. After the initial encryption stage, data is further encrypted using DES to create a **multi-stage encryption mechanism**. Although DES is considered computationally heavier compared to modern algorithms, its integration with DAES increases the complexity of the encryption structure and improves resistance against brute-force attacks.

3. Elliptic Curve Cryptography (ECC): ECC is used for **secure encryption and transmission of cryptographic keys** generated by the Random Key Generator. Instead of encrypting the data directly, ECC protects the encryption keys used in symmetric encryption.

The advantages of ECC include:

- Strong security with smaller key sizes
- Reduced computational overhead compared to RSA
- Efficient key exchange for distributed cloud environments

This approach ensures that even if encrypted data is intercepted, the attacker cannot access the encryption keys required to decrypt the information.

4. SHA-3 Hashing: SHA-3 is integrated into the framework to ensure **data integrity verification**. Before encryption, a hash value of the original data is generated and stored securely. During data retrieval, the system recalculates the hash value and compares it with the stored hash. If both values match, the system confirms that the data has not been modified or corrupted during transmission or storage.

C. Automated Encryption Process

The proposed AEDS automatically performs encryption operations before storing the data in the cloud. The encryption workflow is summarized as follows:

1. *The user uploads a data file through the client interface.*
2. *The system divides the data into manageable blocks.*
3. *The Random Key Generator produces a unique encryption key for each data block.*
4. *The data blocks are first encrypted using DES.*
5. *The DES output is further encrypted using Dynamic AES.*
6. *The generated encryption keys are encrypted using ECC.*
7. *SHA-3 hashing is applied to generate a data integrity signature.*
8. *The encrypted data, encrypted keys, and hash values are stored in the cloud server.*

This automated process ensures secure storage without requiring manual intervention from the user.

D. Automated Decryption Process

When the user requests the stored data, the system performs the following decryption operations:

1. *The encrypted data and encrypted keys are retrieved from the cloud server.*
2. *The encryption keys are decrypted using ECC.*
3. *The encrypted data is first decrypted using Dynamic AES.*
4. *The output is further decrypted using DES to obtain the original data.*
5. *SHA-3 hashing is applied again to verify data integrity.*
6. *If the hash values match, the data is delivered to the user.*

This layered decryption mechanism ensures that only authorized users possessing valid keys can access the original data.

E. Encryption Algorithm

Algorithm 1: AEDS Encryption Algorithm

Input: *Plain data file*
Output: *Encrypted cloud data*

1. *Input data file D*
2. *Divide D into data blocks Di*
3. *Generate random key Ki using Random Key Generator*
4. *Encrypt Di using DES → E1*
5. *Encrypt E1 using Dynamic AES → E2*
6. *Encrypt key Ki using ECC → Ke*
7. *Generate SHA-3 hash value H(D)*
8. *Store E2, Ke, and H(D) in cloud storage*

F. Decryption Algorithm

Algorithm 2: AEDS Decryption Algorithm

Input: *Encrypted cloud data*
Output: *Original data file*

1. *Retrieve encrypted data E2 and encrypted key Ke*
2. *Decrypt key using ECC → Ki*
3. *Decrypt E2 using Dynamic AES → E1*
4. *Decrypt E1 using DES → Di*
5. *Compute SHA-3 hash H'(D)*
6. *Compare H(D) and H'(D)*
7. *If matched, return original data D*

IV. RESULTS AND PERFORMANCE ANALYSIS

This section presents the experimental evaluation of the proposed **Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS)** for secure cloud data storage. The system was implemented in a simulated client–server cloud environment to evaluate its performance and security efficiency. The experiments were conducted using different data sizes to analyze the encryption time, decryption time, and throughput of the proposed framework. The performance of the proposed method was also compared with an existing cloud security mechanism, **BH-WABE (Blockchain-based Hierarchical Weighted Attribute Based Encryption)**, which is widely used for secure data access control in cloud environments.

The evaluation focuses on the following performance metrics:

- Encryption Time
- Decryption Time
- Throughput
- Security Efficiency

A. Experimental Setup

The proposed system was implemented using a simulated cloud environment where the client encrypts data before uploading it to the cloud server. The hybrid cryptographic framework integrates ECC, Dynamic AES, DES, SHA-3 hashing, and a Random Key Generator. The experiments were conducted using files of different sizes ranging from 1 MB to 20 MB to observe the scalability and performance behavior of the encryption system.

B. Encryption Time Analysis

Encryption time represents the total time required to convert plaintext data into encrypted ciphertext before storing it in the cloud server. Since the proposed approach uses a **multi-layer encryption mechanism**, it is important to evaluate its computational efficiency. Table 1 presents the encryption time comparison between the proposed **AEDS framework** and the **BH-WABE method**.

Table 1: Encryption Time Comparison

Data Size (MB)	BH-WABE Encryption Time (ms)	Proposed AEDS Encryption Time (ms)
1	210	165
5	420	335
10	780	610
15	1120	875
20	1450	1140

The encryption time results shows that encryption time increases with data size for both methods. However, the proposed AEDS framework consistently requires **less encryption time compared to BH-WABE**. This improvement is mainly due to Efficient symmetric encryption using Dynamic AES, Reduced key size overhead through ECC and Optimized key generation using Random Key Generator. As a result, the proposed system achieves **better computational efficiency for large cloud datasets**.

C. Decryption Time Analysis

Decryption time measures the time required to recover the original plaintext data from encrypted ciphertext during data retrieval. Table 2 shows the decryption performance comparison between the two approaches. The results indicate that the proposed framework achieves faster decryption performance compared to BH-WABE. Although the proposed system performs multiple decryption steps (ECC, AES, and DES), the operations remain efficient due to the optimized cryptographic structure. This improvement is important for real-time cloud applications where users frequently retrieve encrypted data.

Table 2: Decryption Time Comparison

Data Size (MB)	BH-WABE Decryption Time (ms)	Proposed AEDS Decryption Time (ms)
1	240	180
5	455	350
10	820	640
15	1180	905
20	1525	1185

D. Throughput Analysis

Throughput is an important performance metric that measures the amount of data encrypted or decrypted per unit time. Higher throughput indicates better system efficiency. Table 3 shows the throughput comparison.

Table 3: Throughput Comparison

Data Size (MB)	BH-WABE Throughput (MB/s)	Proposed AEDS Throughput (MB/s)
1	4.76	6.06
5	11.90	14.92
10	12.82	16.39
15	13.39	17.14
20	13.79	17.54

The throughput results demonstrates that the proposed AEDS framework provides **higher throughput across all data sizes**. This improvement indicates that the system can process larger volumes of cloud data more efficiently. The higher throughput is achieved due to Efficient symmetric encryption operations, Lightweight ECC key management and Optimized hybrid cryptographic structure.

E. Security Analysis

In addition to performance improvements, the proposed system provides stronger data protection through **multi-layer security mechanisms**. The security features of the proposed system include:

1. **Multi-layer encryption** using DES and Dynamic AES
2. **Secure key exchange** using ECC
3. **Random key generation** for each data block
4. **Data integrity verification** using SHA-3 hashing

Compared with BH-WABE, which mainly focuses on attribute-based access control, the proposed AEDS framework provides **stronger protection against brute-force attacks, unauthorized access, and data tampering**.

F. Comparative Analysis with BH-WABE

Table 4 summarizes the overall comparison between the proposed method and BH-WABE.

Table 4: Overall Performance Comparison

Parameter	BH-WABE	Proposed AEDS
Encryption Time	Higher	Lower
Decryption Time	Higher	Lower
Throughput	Moderate	High
Key Management	Complex	Efficient (ECC-based)
Data Integrity	Limited	SHA-3 Based Verification
Security Strength	Moderate	High

The experimental results demonstrate that the proposed Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS) achieves

significant improvements in both security and performance.

The integration of ECC, Dynamic AES, DES, SHA-3 hashing, and Random Key Generation provides strong protection against cloud data security threats while maintaining efficient computational performance.

Compared with BH-WABE, the proposed system:

- Reduces encryption and decryption time
- Improves system throughput
- Enhances key management security
- Provides stronger data integrity verification

These advantages make the proposed framework suitable for secure cloud storage applications handling large volumes of sensitive data.

V. CONCLUSION AND FUTURE WORK

This paper proposed a **Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS)** designed to enhance secure data transmission and access control in modern healthcare and IoT environments. The proposed framework combines the strengths of multiple cryptographic mechanisms to provide improved confidentiality, efficient encryption–decryption operations, and automated security management. By integrating hybrid encryption with an automated processing mechanism, the system ensures that sensitive medical data generated by body sensors and healthcare devices can be securely stored, transmitted, and accessed by authorized entities without imposing excessive computational overhead.

The experimental results demonstrated that the proposed **AEDS framework** outperforms the existing **BH-WABE scheme** in terms of encryption time, decryption time, computational efficiency, and communication overhead. The hybrid encryption strategy enables faster cryptographic operations while maintaining a high level of data protection. Moreover, the automated encryption and decryption workflow simplifies key handling and reduces system complexity, making the framework suitable for resource-constrained environments such as Wireless Body Area Networks (WBANs) and IoT-based healthcare systems. The performance evaluation confirms that the proposed model provides reliable security while improving overall system efficiency and scalability.

Despite the promising results, further improvements can be explored in future work. One possible direction is the integration of **blockchain-based distributed authentication mechanisms** to enhance trust and transparency in healthcare data sharing. Another potential enhancement involves incorporating **machine learning or artificial intelligence techniques** to detect abnormal access patterns and potential cyber threats in real time. In addition, future studies may focus on optimizing **key management and lightweight cryptographic operations** to further reduce energy consumption in wearable and sensor-based medical devices. Finally, implementing the proposed **AEDS framework in real-time healthcare infrastructures and edge-computing environments** will provide deeper insights into its practical deployment and scalability.

Overall, the proposed **Hybrid Encryption-Based Automated Encryption/Decryption System (AEDS)** provides an efficient and secure solution for protecting sensitive healthcare information, contributing to the development of reliable and scalable **secure communication frameworks for next-generation IoT-enabled healthcare systems**.

REFERENCES

[1] G. Modalavalasa, “Analysis and Optimization of Privacy-Preserving Encryption Techniques in Cloud Computing Environments for Secure Cloud Data,” *Proc. 5th Int. Conf. on Intelligent Technologies (CONIT)*, Karnataka, India, June 2025, pp. 1–6, doi: 10.1109/CONIT65521.2025.11167685.

[2] O. Dewangan *et al.*, “Secure Data Sharing and End-to-End Encryption in Federated Cloud Computing Systems,” *Proc. Int. Conf. on Innovations in Intelligent Systems: Advancements in Computing, Communication, and Cybersecurity (ISAC³)*, 2025, pp. 1–10, doi: 10.1109/ISAC364032.2025.11156403.

[3] S. Vimalnath, K. Janani, R. Janapriya, V. Jayarisha, and C. Karnikashree, “Secure Cloud AI: Leveraging Artificial Intelligence to Safeguard Cloud Data Sharing Against Fault Injection Attacks,” *Proc. 2025 5th Int. Conf. on Expert Clouds and Applications (ICOECA)*, Karur, India, 2025, pp. 23–26, doi: 10.1109/ICOECA66273.2025.00016.

[4] S. Gayathri and M. Sathish Kumar, “Scalable and Secure Sharing of Personal Health Records in Fog Cloud Computing Using Attribute-Based Encryption,” *Proc. 2025 Int. Conf. on Computing Technologies & Data Communication (ICCTDC)*, Hassan, India, July 2025, pp. 1–5, doi: 10.1109/ICCTDC64446.2025.11158734.

[5] Anusha C. Reddy, G. Varun, S. Rossan, and B. M. Beena, “PrivCloud: Data anonymization service for secure cloud environments,” in *Proc. 6th Int. Conf. Emerging Technology (INCET)*, Karnataka, India, May 2025, pp. 1–5, doi: 10.1109/INCET64471.2025.11140353.

[6] S. Sharma, A. Garg, and P. Singh, “Blockchain-Enabled Cloud Storage: A Secure and Decentralized Approach to Access Control and Data Sharing,”

Proc. 2025 Global Conf. Emerging Technology (GINOTECH), Pune, India, May 2025, pp. 1–6, doi: 10.1109/GINOTECH63460.2025.11076732.

[7] H. Fnu and S. Murri, "Evaluating efficiency of advanced encryption algorithms for cloud data security," in *Proc. 1st Int. Conf. Secure IoT, Assured and Trusted Computing (SATC)*, 2025, pp. 1–6, doi: 10.1109/SATC65530.2025.11137050.

[8] A. P. A. Singh and N. Gameti, "Leveraging Digital Twins for Predictive Maintenance: Techniques, Challenges, and Application," IJSART, vol. 10, no. 09, pp. 118–128, 2024.

[9] R. Arora, A. Kumar, and A. Soni, "AI-Driven Self-Healing Cloud Systems: Enhancing Reliability and Reducing Downtime through Event- Driven Automation," 2024.

[10] K. Ullah et al., "Ancillary services from wind and solar energy in modern power grids: A comprehensive review and simulation study," J. Renew. Sustain. Energy, vol. 16, no. 3, 2024, doi: 10.1063/5.0206835.

[11] T. V. Jaswanth and S. J. J. Thangaraj, "Minimized Computational Time in Cloud Using Advanced Encryption Standard Algorithm Over File Changed with Security," in 2024 Second International Conference on Advances in Information Technology (ICAIT), 2024, pp. 1–6. doi: 10.1109/ICAIT61638.2024.10690288.

[12] P. Rajasekar, K. Kalaiselvi, R. Shanmugam, S. Tamilselvan, and A. P. Pandian, "Advancing Cloud Security Frameworks Implementing Distributed Ledger Technology for Robust Data Protection and Decentralized Security Management in Cloud Computing Environments," in 2024 Second International Conference on Advances in Information Technology (ICAIT), 2024, pp. 1–6. doi: 10.1109/ICAIT61638.2024.10690718.