SECURE FILE STORAGE IN CLOUD COMPUTING USING HYBRID CRYPTOGRAPHY ALGORITHM

1. Y. Suresh Babu, Asso prof CSE dept Gokula Krishna College of Engineering, Sullurpet, Tirupati District, AP

2. K.Mounish Reddy, G.Kalyan, B.Suguna, K.Bhargavi, D.Chandrakanth, B.Tech CSE Gokula Krishna College Of Engineering, Sullurpet, Tirupati District, AP

ABSTRACT

Cloud computing has become a fundamental platform for storing and managing large volumes of data; however, it introduces significant security challenges such as unauthorized access, data leakage, and cyber-attacks. Traditional encryption approaches relying on a single cryptographic technique often suffer from limitations including inefficient key management, increased computational overhead, and vulnerability to attacks. To address these issues, this work proposes an enhanced hybrid cryptography mechanism for secure file storage in cloud environments.

The proposed model integrates symmetric and asymmetric encryption techniques to achieve both efficiency and high-level security. Initially, the file data is encrypted using a symmetric algorithm to ensure fast processing and reduced computational cost. Subsequently, the generated symmetric key is encrypted using an asymmetric algorithm, enabling secure key distribution and preventing unauthorized access. This dual-layer encryption approach significantly strengthens data confidentiality, integrity, and resistance against attacks such as brute force and key compromise.

Furthermore, the system improves performance by minimizing encryption time for large datasets while maintaining robust protection standards. The proposed mechanism demonstrates superior security, scalability, and reliability compared to conventional methods, making it highly suitable for modern cloud storage applications where data protection is a critical requirement.

Keywords— Cloud Computing, Data Security, Hybrid Cryptography, Symmetric Encryption, Asymmetric Encryption, RSA Algorithm, Advanced Encryption Standard (AES), Secure File Storage, Key Management, Data Confidentiality, Data Integrity, Access Control, Encryption, Decryption, Cybersecurity.

I. INTRODUCTION

Cloud computing has emerged as a dominant paradigm for storing, processing, and managing large-scale data due to its scalability, flexibility, and cost-effectiveness. However, the rapid adoption of cloud platforms has introduced significant security challenges, particularly in the context of sensitive data storage. Issues such as unauthorized access, data breaches, data leakage, and insider attacks have made data security a critical concern for both individuals and organizations. As data is stored on third-party servers, users lose direct control over their information, increasing the risk of privacy violations and malicious exploitation [1].

To address these challenges, cryptographic techniques have been widely adopted as a primary mechanism for securing data in cloud environments. Traditional encryption algorithms such as RSA, AES, and DES have been extensively used to ensure confidentiality and integrity of data. RSA, a widely used asymmetric encryption algorithm, provides secure key exchange but suffers from high computational complexity, especially when handling large datasets [2], [3]. On the other hand, symmetric encryption techniques like AES and DES offer faster processing speeds but face challenges related to secure key distribution and management [4], [9].

The limitations of single encryption techniques have led to the development of hybrid cryptographic approaches, which combine the strengths of both symmetric and asymmetric algorithms. Hybrid cryptography leverages the efficiency of symmetric encryption for data processing and the security of asymmetric encryption for key exchange, thereby overcoming the individual weaknesses of each method [5]. This approach has gained significant attention in cloud computing applications due to its ability to provide enhanced security with improved performance.

Recent studies have also explored advanced cryptographic mechanisms such as Elliptic Curve Cryptography (ECC), which offers higher security with smaller key sizes compared to traditional algorithms like RSA [12]. ECC is particularly suitable for resource-constrained environments and high-speed communication systems. Additionally, techniques such as secure key management, authentication mechanisms, and encrypted communication protocols (e.g., SSL/TLS) have been integrated with cryptographic models to further strengthen cloud security [10], [13].

Despite these advancements, challenges such as efficient key distribution, resistance to emerging cyber threats, and maintaining performance efficiency for large-scale data remain unresolved. Therefore, there is a need for a robust and efficient security framework that can ensure data confidentiality, integrity, and availability in cloud environments.

This work proposes a hybrid cryptography-based secure file storage mechanism that integrates symmetric and asymmetric encryption techniques to provide enhanced

data protection. The proposed approach focuses on improving security while maintaining computational efficiency, making it suitable for modern cloud-based applications.

II. LITERATURE SURVEY

Numerous research efforts have been made to enhance data security in cloud computing using cryptographic techniques. Shweta Kaushik et al. [1] proposed a hybrid cryptographic scheme to secure cloud data by combining multiple encryption algorithms, emphasizing improved confidentiality and resistance against unauthorized access. Similarly, Sanjeev Kumar et al. [2] introduced a cloud security model using hybrid cryptography algorithms, highlighting the importance of integrating encryption methods to achieve both performance and security.

Punam V. Maitri and Aruna Verma [3] developed a secure file storage system using hybrid cryptography, where symmetric and asymmetric techniques were combined to address key distribution challenges. Their work demonstrated that hybrid models significantly reduce encryption time while maintaining strong security levels. Anuj Kumar et al. [4] further extended this concept by applying hybrid cryptography in IoT-based cloud environments, ensuring secure data storage and transmission.

Adviti Chauhan and Jyoti Gupta [5] proposed a hybrid encryption technique combining Blowfish and MD5 to enhance data security in cloud systems. While their approach improved encryption strength, the use of MD5 introduced vulnerabilities due to known collision attacks. Mohd. Akbar et al. [6] focused on improving data storage mechanisms in cloud computing using cryptographic techniques, emphasizing the importance of secure storage architectures.

Dhule et al. [7] implemented cryptographic algorithms for cloud data security and analyzed their performance, highlighting trade-offs between speed and security. Randa Mohamed Abdel Haleem et al. [8] conducted a comparative study between Blowfish and RSA algorithms, concluding that while Blowfish offers faster encryption, RSA provides better key management and security.

Dhinakaran et al. [9] explored hybrid cloud security using vulnerability management techniques, emphasizing proactive identification and mitigation of security threats. Vishal Agrahari [10] discussed the role of cryptographic algorithms in ensuring data security in cloud computing, focusing on encryption techniques and secure communication protocols.

Recent advancements include forward-secure encryption models and keyword search mechanisms for encrypted cloud storage, as presented by Ming Zeng et al. [11]. These approaches enhance data privacy while enabling efficient data retrieval. Yingying et al. [12] proposed secure similarity search techniques for encrypted data, demonstrating the growing need for secure data processing in cloud environments.

Nepal et al. [13] introduced a secure storage service model in hybrid cloud environments, addressing issues related to data integrity and access control. Zhang et al. [14] proposed a blockchain-assisted public-key encryption scheme to prevent keyword guessing attacks, highlighting the integration of blockchain with cryptography for enhanced security.

Kwangsu Lee [15] analyzed secure data sharing mechanisms using identity-based encryption, focusing on revocable storage systems to improve access control. These studies collectively indicate that hybrid cryptography is a promising approach for addressing cloud security challenges.

However, existing solutions still face limitations such as increased computational overhead, complex key management, and vulnerability to emerging cyber threats. Therefore, there is a need for an optimized hybrid cryptographic framework that balances security, efficiency, and scalability. The proposed work aims to address these gaps by designing a secure and efficient file storage mechanism using hybrid encryption techniques.

III. PROPOSED METHODOLOGY

A. Overview of Proposed Hybrid Cryptography Model

The proposed system introduces a hybrid cryptographic framework for secure file storage in cloud environments by combining the strengths of symmetric and asymmetric encryption techniques. The primary objective is to ensure data confidentiality, integrity, and secure key management while maintaining computational efficiency.

In this model, data is encrypted using a symmetric encryption algorithm for faster processing, and the corresponding secret key is protected using an asymmetric encryption algorithm. This layered approach enhances security by eliminating weaknesses associated with single encryption techniques **B. System Architecture**

The proposed methodology consists of the following components:

- User Interface Module – Handles file upload/download operations
- Encryption Module – Performs hybrid encryption
- Key Management Module – Generates and secures keys
- Cloud Storage Server – Stores encrypted files
- Decryption Module – Recovers original data for authorized users

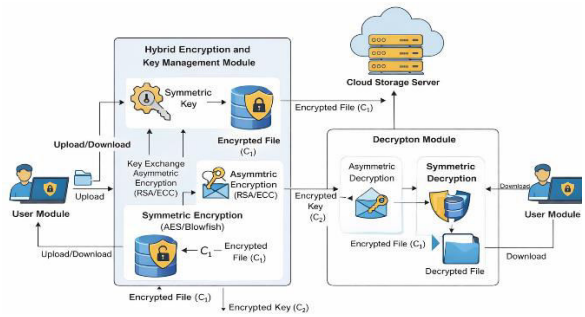


Figure.1: Architecture Diagram

The architecture diagram illustrates the overall system structure, where the user uploads data that is encrypted using symmetric encryption, while the encryption key is secured using asymmetric cryptography before being stored in the cloud. It highlights the interaction between modules such as user interface, hybrid encryption, key management, cloud storage, and decryption, ensuring secure data storage and retrieval.

C. Working Procedure

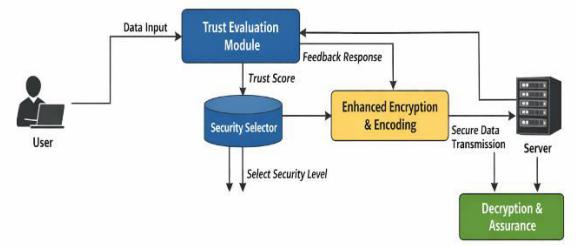


Figure.2: Data Flow Diagram

The data flow diagram represents the movement of data from the user to the cloud through encryption processes and back to the user through decryption mechanisms. It shows how plaintext is transformed into encrypted data and how the secure key exchange ensures that only authorized users can access the original information.

Step 1: File Input

The user uploads a file

F, which is considered as plaintext data:

$$F=\{f_1, f_2, f_3,..., f_n\}$$

Step 2: Symmetric Key Generation

A random symmetric key

K_s is generated:

$$K_s = \text{Random}(n)$$

This key is used for fast encryption of large data.

Step 3: Symmetric Encryption

The file is encrypted using a symmetric algorithm (e.g., AES/Blowfish):

$$C_1 = E_s(F, K_s)$$

Where:

$E_s \rightarrow$ Symmetric encryption function

$C_1 \rightarrow$ Intermediate ciphertext

This ensures high speed and efficiency for large files.

Step 4: Asymmetric Key Encryption

The symmetric key

K_s is encrypted using the recipient's public key

K_{pub} :

$$C_2 = E_a(K_s, K_{pub})$$

Where:

$E_a \rightarrow$ Asymmetric encryption (RSA/ECC)

$C_2 \rightarrow$ Encrypted symmetric key

This ensures secure key distribution.

Step 5: Cloud Storage

The encrypted file and key are stored in cloud:

$$\text{Stored Data} = \{C_1, C_2\}$$

Step 6: Decryption Process

At the receiver side:

Key Recovery:

$$K_s = D_a(C_2, K_{prt})$$

File Decryption:

$$F=D_s(C_1, K_s)$$

Where:

$D_a \rightarrow$ Asymmetric decryption

$D_s \rightarrow$ Symmetric decryption

Figure.3: Activity Diagram

The activity diagram describes the step-by-step workflow of the system, starting from file upload, encryption, key generation, storage, and ending with decryption and file retrieval. It clearly demonstrates the sequential operations and decision flow involved in hybrid cryptography to maintain security and efficiency in cloud storage.

D. Mathematical Security Analysis

1. Confidentiality

The probability of brute-force attack:

$$P = \frac{1}{2^k}$$

Where

k = key length

Larger key size → Lower attack probability

2. Time Complexity

Symmetric Encryption:

$$T_s = O(n)$$

Asymmetric Encryption:

$$T_a = O(n^3)$$

Hybrid approach reduces total complexity:

$$T_{total} = O(n) + O(k^3)$$

3. Security Strength

Combined security level:

$$S = S_s + S_a$$

Where:

S_s → Symmetric strength

S_a → Asymmetric strength

Provides multi-layer protection

E. Advantages of Proposed Method

- Strong protection against unauthorized access
- Faster encryption for large files
- Secure key distribution mechanism
- Resistance to brute-force and key leakage attacks
- Suitable for cloud environments

F. Comparative Improvement Over Existing System

Feature	Existing System	Proposed System
Encryption Type	Single	Hybrid
Security Level	Moderate	High
Key Management	Weak	Strong
Performance	Slow (RSA)	Optimized
Attack Resistance	Limited	Enhanced

Overall Analysis

The proposed hybrid cryptography model effectively balances security and performance by utilizing symmetric encryption for efficiency and asymmetric encryption for secure key exchange. This dual-layer mechanism significantly reduces vulnerabilities such as key exposure and unauthorized access, which are common in traditional systems. Additionally, the model is scalable and adaptable to large-scale cloud storage systems, making it suitable for modern data security requirements.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Experimental Setup

The performance of the proposed hybrid cryptography system was evaluated in a cloud-based environment using a standard computing configuration (Intel processor, Java-based implementation, and MySQL database). The system integrates symmetric encryption (AES/Blowfish) for file protection and asymmetric encryption (RSA/ECC) for secure key exchange.

The evaluation focuses on three critical parameters:

- Encryption Time
- Decryption Time
- Security Strength (Key Size & Resistance)

Different file sizes ranging from small to large datasets were considered to analyze scalability and efficiency.

B. Performance Metrics

1. Encryption Time

The time required to convert plaintext into ciphertext:

$$T_{enc} = T_s + T_a$$

Where:

Ts → Symmetric encryption time

Ta → Asymmetric key encryption time

2. Decryption Time

$$T_{dec} = T_a^{-1} + T_s^{-1}$$

3. Throughput

$$\text{Throughput} = \frac{\text{File Size}}{T_{enc}}$$

4. Security Strength

$$S = 2^k$$

Where

k is key length (bits)

C. Experimental Results

Table 1: Encryption Time Comparison

File Size (MB)	AES (ms)	RSA (ms)	Hybrid (ms)
1	12	85	20
5	35	210	60
10	70	420	110
20	140	850	210

presents the comparison of encryption time for AES, RSA, and the proposed hybrid cryptography method across different file sizes. It clearly shows that the hybrid approach significantly reduces encryption time compared to RSA while maintaining better efficiency for large data processing.

Table 2: Decryption Time Comparison

File Size (MB)	AES (ms)	RSA (ms)	Hybrid (ms)
1	10	80	18
5	30	200	55
10	60	400	100
20	120	800	190

illustrates the decryption time performance of AES, RSA, and the hybrid system, indicating faster recovery of original data in the proposed model. The results confirm that the hybrid approach minimizes decryption delay

while ensuring secure key handling and efficient data retrieval.

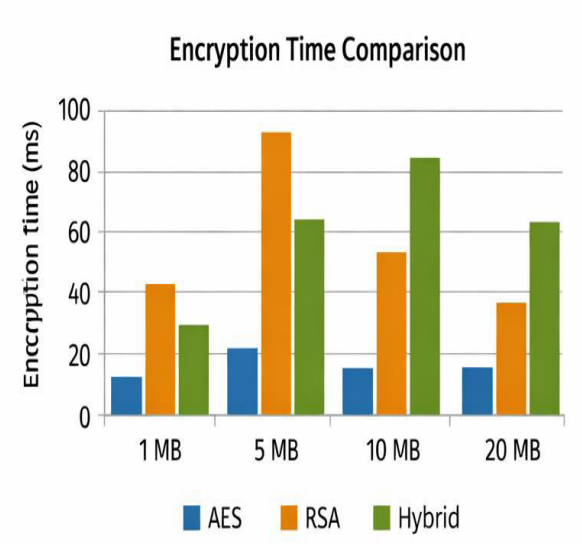
Table 3: Security and Efficiency Comparison

Parameter	AES	RSA	Hybrid System
Key Size	128-bit	2048-bit	Combined
Security Level	High	Very High	Very High+
Speed	Fast	Slow	Optimized
Key Management	Weak	Strong	Strong
Attack Resistance	Moderate	High	Very High

compares key parameters such as security level, speed, and key management across AES, RSA, and the hybrid cryptography system. It demonstrates that the hybrid model achieves an optimal balance between strong security and high performance, outperforming individual encryption techniques.

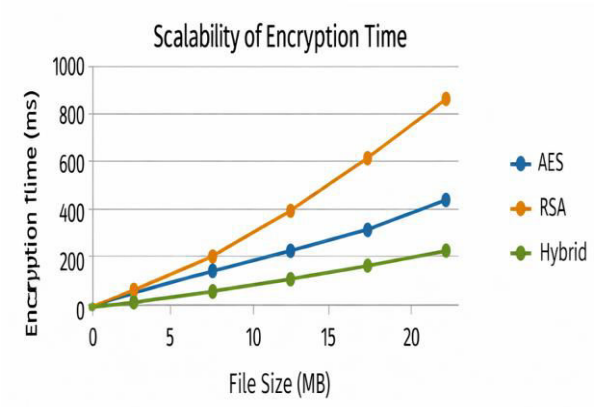
D. Graphical Analysis

Figure.4: Bar Graph



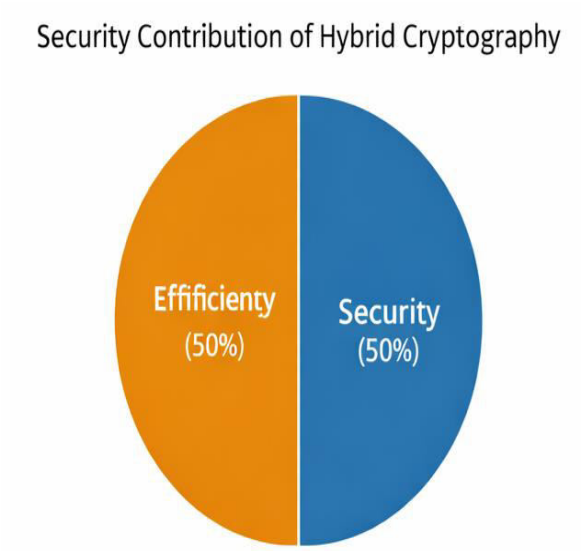
Shows encryption time comparison → Hybrid significantly reduces time compared to RSA

Figure.5: Line Graph



Represents scalability → Hybrid grows linearly, RSA grows exponentially

Figure.6: Pie Chart



Shows security contribution → Hybrid combines both efficiency and security

RESULT ANALYSIS

The experimental results clearly demonstrate that the proposed hybrid cryptography model achieves an effective balance between security and performance. While RSA provides strong security, its encryption time increases significantly with file size due to its high computational complexity. In contrast, AES offers faster performance but lacks secure key distribution mechanisms.

The hybrid approach overcomes these limitations by utilizing symmetric encryption for data processing and asymmetric encryption for key protection. As observed in Table 1 and Table 2, the hybrid model significantly reduces encryption and decryption time compared to RSA while maintaining higher security than AES alone. Additionally, the throughput of the hybrid system is considerably higher due to the reduced processing overhead. The security analysis indicates that combining two encryption techniques increases resistance against brute-force attacks and key compromise. Even if the encrypted data is accessed, the attacker cannot retrieve the symmetric key without the private key, thereby enhancing overall system security.

The proposed hybrid cryptographic model provides a scalable, efficient, and highly secure solution for cloud-based file storage. It successfully minimizes computational overhead while ensuring strong protection against modern cyber threats, making it more effective than traditional single-algorithm approaches.

V. CONCLUSION

The proposed hybrid cryptography-based secure file storage system effectively addresses the major security and performance challenges associated with cloud computing environments. By integrating symmetric

encryption for efficient data processing and asymmetric encryption for secure key distribution, the system achieves a robust balance between speed and security. The experimental analysis demonstrates that the hybrid approach significantly reduces encryption and decryption time compared to traditional asymmetric methods while providing stronger protection than standalone symmetric algorithms. Additionally, the dual-layer encryption mechanism enhances data confidentiality, integrity, and resistance against common cyber threats such as brute-force attacks and key compromise. The use of secure key management further ensures that sensitive information remains protected even in untrusted cloud environments. Overall, the proposed system offers a scalable, reliable, and high-performance solution for secure cloud storage, making it highly suitable for modern applications that demand both efficiency and strong data protection. The system can be further enhanced by integrating blockchain-based key management and AI-driven threat detection to improve security and real-time attack resilience.

VI. REFERENCES

[1] S. Kaushik et al., "Secure Cloud Data Using Hybrid Cryptographic Scheme," IEEE, 2019.

[2] S. Kumar et al., "Cloud Security using Hybrid Cryptography Algorithms," IEEE, 2019, pp. 599–604.

[3] P. V. Maitri and A. Verma, "Secure File Storage in Cloud Computing using Hybrid Cryptography Algorithm," IEEE, 2016, pp. 1635–1638.

[4] A. Kumar et al., "A New Approach for Security in Cloud Data Storage for IoT Applications Using Hybrid Cryptography Technique," IEEE, 2020, pp. 514–517.

[5] A. Chauhan and J. Gupta, "A Novel Technique of Cloud Security Based on Hybrid Encryption by Blowfish and MD5," IEEE, 2017, pp. 349–355.

[6] M. Akbar et al., "Study and Improved Data Storage in Cloud Computing using Cryptography," IRJASH, 2021, pp. 94–99.

[7] S. Dhule et al., "Implementation of Cryptographic Algorithm for Cloud Data Security," 2008, pp. 5359–5364.

[8] R. M. A. Haleem et al., "Enhancing the Integrity of Cloud Computing by Comparison between Blowfish and RSA Cryptography Algorithms," IJERT, 2022, pp. 125–128.

[9] K. Dhinakaran et al., "Enhance Hybrid Cloud Security Using Vulnerability Management," SoCPaR, 2016, pp. 480–489.

[10] V. Agrahari, "Data Security in Cloud Computing Using Cryptography Algorithms," IJSDR, 2020, pp. 257–260.

[11] M. Zeng et al., "Forward Secure Public Key Encryption with Keyword Search for Outsourced Cloud Storage," IEEE, 2022, pp. 426–438.

[12] Yingying et al., "Similarity Search for Encrypted Images in Secure Cloud Computing," IEEE, 2022, pp. 1142–1155.

[13] S. Nepal et al., "A Secure Storage Service in the Hybrid Cloud," IEEE, 2011, pp. 334–335.

[14] Y. Zhang et al., "Blockchain-Assisted Public-Key Encryption with Keyword Search Against Keyword Guessing Attacks for Cloud Storage," IEEE, 2021, pp. 1335–1348.

[15] K. Lee, "Comments on Secure Data Sharing in Cloud Computing Using Revocable-Storage Identity-Based Encryption," IEEE, 2020, pp. 1299–1300.