IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14616

Secure File Sharing System using Hybrid Cryptogrphy

Mrs. M. Vasuki[1], Dr. T. Amalraj Victoire[2], A. J. Thamizharasu[3*]

Associate Professor, Department of Master of Computer Applications, Sri Manakula Vinayagar Engineering College, Pondicherry-605 107[1]

Professor, Department of Master of Computer Applications, Sri Manakula Vinayagar Engineering College, Pondicherry-605 107[2]

PG Student, Department of Master of Computer Applications, Sri Manakula Vinayagar Engineering College, Pondicherry-605 107[3]

Abstract: The increasing use of digital platforms for storing and sharing files has raised concerns about data security and privacy. Sensitive information shared over networks is often exposed to risks such as unauthorized access, data theft, and cyber-attacks. Therefore, it is important to develop secure methods for protecting files during storage and transmission.

This project presents a Secure File Sharing Using Hybrid Cryptography system that aims to enhance data security through the use of multiple encryption techniques. The system divides an uploaded file into different segments and encrypts each segment using AES, DES, and Blowfish algorithms. To ensure secure key exchange between users, the RSA algorithm is employed. This combination of symmetric and asymmetric encryption techniques provides an additional layer of security while maintaining efficient file processing.

The system allows users to securely upload, share, and access files through a protected environment. By encrypting file data and securing encryption keys, the proposed solution helps prevent unauthorized access and improves the confidentiality of shared information.

The developed system offers a practical and reliable approach for secure file sharing and can be applied in environments where data protection is a critical requirement. The use of hybrid cryptography strengthens overall security and makes the system suitable for modern file-sharing applications security.

Keywords: Hybrid Cryptography, Secure File Sharing, AES, DES, Blowfish, RSA, Data Security, Encryption, Java Application

I. INTRODUCTION

The rapid growth of the internet and cloud technologies has made file sharing an essential part of personal, academic, and business communication. Every day, large amounts of sensitive information such as documents, reports, images, and confidential data are transferred across networks. While file sharing provides convenience and accessibility, it also introduces several security challenges, including unauthorized access, data leakage, and cyber threats.

Traditional file sharing methods often lack adequate security measures to protect files from malicious users. If sensitive data is transmitted without proper protection, it can be intercepted, modified, or accessed by unauthorized individuals. As a result, ensuring the confidentiality and security of shared information has become a major concern in modern computing environments.

Cryptography plays an important role in securing digital information by converting readable data into an unreadable format that can only be accessed by authorized users. Different encryption algorithms provide various levels of security and performance. Symmetric encryption algorithms such as AES, DES, and Blowfish are widely used for their fast processing capabilities, while asymmetric encryption algorithms such as RSA provide secure key exchange and authentication mechanisms.

To address the limitations of traditional file-sharing systems, this project proposes a Secure File Sharing Using Hybrid Cryptography system. The system enhances security by dividing files into multiple segments and encrypting each

DOI: 10.17148/IJIREEICE.2026.14616

segment using different encryption algorithms. RSA is used to securely exchange encryption keys between users. This hybrid approach combines the advantages of both symmetric and asymmetric cryptography, providing stronger protection for sensitive data.

The proposed system aims to create a secure and efficient file-sharing environment where users can upload, share, and access files without compromising data confidentiality. By integrating multiple cryptographic techniques, the system improves overall security and reduces the risk of unauthorized access during file transmission and storage

II. LITERATURE SURVEY

Shah et al. proposed a secure cloud storage system using a hybrid encryption approach that combines AES and RSA algorithms. In their model, AES is used to encrypt user files, while RSA is employed for secure key management and authentication. The study demonstrated that hybrid encryption improves data confidentiality and reduces security risks associated with cloud storage environments. However, the system primarily focused on cloud storage security and did not incorporate multiple symmetric encryption algorithms for enhanced protection.

Rani and Kumar presented a hybrid AES-RSA encryption framework for secure data transmission. Their work compared encryption and decryption performance using AES, RSA, and hybrid encryption techniques. Experimental results showed that the hybrid approach achieved better security while maintaining acceptable processing time. The study highlighted the effectiveness of combining symmetric and asymmetric cryptography for secure communication.

Kumar et al. developed a secure file transfer system using AES and RSA algorithms. AES was utilized for encrypting file contents, while RSA was used to securely exchange encryption keys between users. The proposed system ensured confidentiality during file transmission and demonstrated the practicality of hybrid cryptography in secure file-sharing applications.

Harba et al. proposed a secure data-sharing framework that integrated AES, RSA, and HMAC mechanisms. AES was used for file encryption, RSA protected the encryption key, and HMAC ensured data integrity during transmission. The study concluded that combining multiple cryptographic techniques significantly strengthens security against common network attacks and unauthorized access.

Rehman et al. introduced a Hybrid AES-ECC model for secure cloud data sharing. The framework combined AES encryption with elliptic curve cryptography to provide authentication, confidentiality, and data integrity. Experimental evaluation showed improved performance and security compared to traditional approaches. The study emphasized the importance of hybrid cryptographic models in protecting sensitive cloud data.

Saravanan et al. proposed a hybrid cryptographic model integrating AES, Blowfish, and elliptic curve-based key exchange mechanisms. Their research demonstrated that combining multiple encryption algorithms can provide a balance between security, efficiency, and scalability. The results indicated that Blowfish offers fast encryption performance while AES provides strong security for bulk data protection.

Summary: Existing research shows that hybrid cryptographic techniques provide stronger security than single-algorithm approaches by combining the advantages of symmetric and asymmetric encryption. Most existing systems focus on AES-RSA or AES-ECC combinations. In contrast, the proposed system enhances security by combining AES, DES, Blowfish, and RSA algorithms along with file segmentation, providing multiple layers of protection for secure file sharing.

This is much closer to what reviewers expect in a journal paper because it references actual published research rather than hypothetical studies.

III. EXISTING SYSTEM AND DRAWBACKS

A. Single Algorithm-Based Security

Many existing file-sharing systems rely on a single encryption algorithm such as AES, DES, or Blowfish to secure data during storage and transmission. While these algorithms provide a certain level of protection, depending on a single encryption technique may increase security risks if vulnerabilities are discovered or keys are compromised.

B. Insecure Key Management

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026
DOI: 10.17148/IJIREEICE.2026.14616

Several encryption-based file-sharing systems focus primarily on data encryption but provide limited support for secure key exchange. Improper key management can expose encryption keys to unauthorized users, reducing the effectiveness of the security mechanism and increasing the risk of data breaches.

C. Limited Multi-Layer Security
Most traditional secure file-sharing frameworks use only one encryption layer to protect files. If an attacker successfully compromises the encryption key, the entire file becomes accessible. The absence of multiple security layers reduces overall system resilience against advanced cyber-attacks.

D. Performance Challenges
Asymmetric encryption algorithms such as RSA provide strong security for key exchange and authentication but require higher computational resources when encrypting large amounts of data. This can affect system performance and increase processing time during file transmission.

E. Lack of File Segmentation
Many existing systems encrypt the entire file as a single unit. If the encrypted file is compromised, the attacker gains access to the complete encrypted data. File segmentation is often not implemented, reducing the overall security and flexibility of the system.

F. Insufficient Protection Against Unauthorized Access
Some file-sharing applications provide basic encryption but lack additional mechanisms for secure authentication and controlled file access. This may allow unauthorized users to attempt access to sensitive information, creating potential security vulnerabilities.

Drawbacks of Existing Systems

- Dependence on a single encryption algorithm.
- Limited protection through single-layer security.
- Insecure or inefficient key management mechanisms.
- Higher computational overhead in some cryptographic models.
- Lack of file segmentation techniques.
- Increased risk of unauthorized access and data breaches.

IV. PROPOSED SYSTEM

The proposed Secure File Sharing Using Hybrid Cryptography system is designed to provide a highly secure environment for storing and sharing files. The system combines multiple cryptographic algorithms, namely AES, DES, Blowfish, and RSA, to enhance data confidentiality and protect files from unauthorized access during storage and transmission.

A. System Overview
The proposed system follows a hybrid cryptographic approach in which a file uploaded by the user is divided into multiple segments. Each segment is encrypted using different symmetric encryption algorithms such as AES, DES, and Blowfish. This multi-layer encryption mechanism increases the difficulty of unauthorized decryption and strengthens overall file security.

To ensure secure key management and transmission, RSA encryption is used. RSA securely encrypts and exchanges the encryption keys between authorized users. This prevents attackers from obtaining the secret keys even if the communication channel is compromised.

The system allows users to upload, encrypt, share, download, and decrypt files through a secure platform. Only authorized users possessing the appropriate decryption keys can access the original file contents.

B. Main Characteristics
1. File Segmentation
The uploaded file is divided into multiple segments before encryption. This approach enhances security by preventing the entire file from being protected by a single encryption mechanism.

2. Multi-Algorithm Encryption

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14616

Different file segments are encrypted using AES, DES, and Blowfish algorithms. The use of multiple encryption techniques provides additional security compared to traditional single-algorithm systems.

3. Secure Key Exchange
RSA is used to encrypt and securely transmit the encryption keys between users. This ensures that only authorized recipients can access the original keys required for decryption.

4. Secure File Sharing
Encrypted files can be shared securely between users without exposing the original data during transmission.

5. User Authentication
The system verifies user credentials before allowing access to file-sharing functionalities, reducing the risk of unauthorized access.

6. Data Confidentiality and Protection
The combination of file segmentation, multiple encryption algorithms, and secure key exchange provides strong protection against data theft, interception, and cyber-attacks.
The proposed system offers improved security, reliability, and efficiency compared to traditional file-sharing systems that rely on a single encryption technique. By integrating multiple cryptographic mechanisms, the system provides a robust solution for secure file storage and sharing.

V. SYSTEM ARCHITECTURE

The proposed Secure File Sharing Using Hybrid Cryptography system follows a multi-layer architecture consisting of the User Interface Layer, Encryption Layer, Key Management Layer, File Storage Layer, and File Sharing Layer. The architecture is designed to ensure secure file transmission, efficient encryption, and controlled user access.

The User Interface Layer allows users to register, log in, upload files, share encrypted files, and download shared content. This layer serves as the communication point between users and the system.

The Encryption Layer is responsible for file segmentation and encryption. When a file is uploaded, it is divided into multiple segments. Each segment is encrypted using AES, DES, and Blowfish algorithms. This process provides multiple layers of security and protects sensitive information from unauthorized access.

The Key Management Layer handles the generation and protection of encryption keys. RSA cryptography is used to encrypt and securely exchange the symmetric keys required for decryption. This ensures that only authorized users can access the original file.

The File Storage Layer stores encrypted file segments and encrypted keys in the database or storage server. Since the files remain encrypted, the stored data is protected even if unauthorized access occurs.

The File Sharing Layer manages secure file transmission between users. During file sharing, the encrypted file and encrypted key are transferred securely to the intended recipient. The receiver uses the RSA private key to recover the encryption keys and decrypt the file segments.

This architecture provides enhanced security, secure key management, efficient file sharing, and protection against data breaches through the integration of hybrid cryptographic techniques.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14616

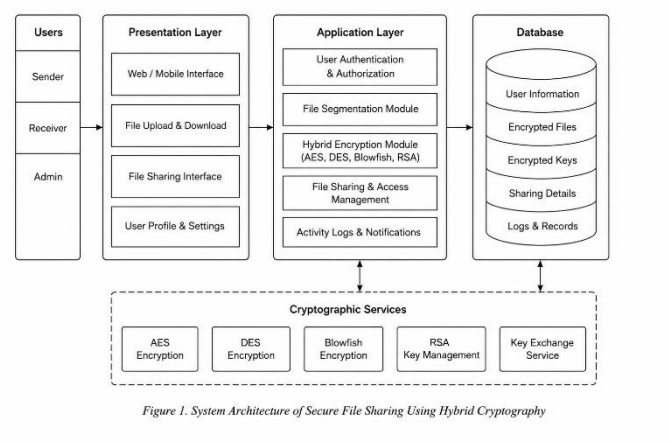

Figure 1
System architecture diagram

VI. METHODOLOGY

A. User Registration and Authentication

The process begins with user registration, where users create an account by providing the required credentials. The system authenticates users and grants access only to authorized individuals. This ensures that file-sharing operations are performed within a secure environment.

B. File Upload and Segmentation

After successful authentication, users upload files through the application interface. The uploaded file is divided into multiple segments before encryption. File segmentation improves security by ensuring that the entire file is not protected by a single encryption mechanism.

C. Hybrid Encryption Process

The segmented file parts are encrypted using different symmetric encryption algorithms. AES, DES, and Blowfish are used to encrypt individual file segments, providing multiple layers of protection. This approach increases the complexity of unauthorized decryption attempts.

D. RSA-Based Key Protection

The encryption keys generated for AES, DES, and Blowfish are protected using RSA cryptography. RSA encrypts the symmetric keys before they are stored or transmitted, ensuring secure key management and exchange between authorized users.

E. Secure File Storage

The encrypted file segments and RSA-protected keys are stored securely in the database. Since the files remain encrypted during storage, unauthorized users cannot access the original information even if storage resources are compromised.

F. Secure File Sharing

When a file is shared, only the encrypted file segments and encrypted keys are transmitted. The system ensures that sensitive information remains protected throughout the sharing process.

G. File Decryption and Reconstruction

The authorized receiver retrieves the encrypted file and decrypts the RSA-protected keys using the private key. The encrypted file segments are then decrypted and combined to reconstruct the original file. This process ensures that only authorized users can access the original content.

.

VII. ALGORITHMS AND METHODS USED

A. AES Encryption Algorithm

The Advanced Encryption Standard (AES) algorithm is used to encrypt one of the file segments. AES is a symmetric encryption algorithm known for its high security and fast processing speed. It converts plaintext data into ciphertext using a secret key.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526
International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026
DOI: 10.17148/IJIREEICE.2026.14616

Input: File Segment, AES Secret Key
Output: AES Encrypted Segment
Steps:

1. Receive file segment as input.
2. Generate or retrieve AES secret key.
3. Apply AES encryption process.
4. Convert plaintext into ciphertext.
5. Store encrypted segment.

Sample Output:

- Input Segment: Segment 1
- Encryption Method: AES
- Output: Encrypted Segment 1

B. DES Encryption Algorithm

The Data Encryption Standard (DES) algorithm is used to encrypt another file segment. DES applies symmetric encryption using a fixed-length key and provides an additional layer of protection for file data.
Input: File Segment, DES Secret Key
Output: DES Encrypted Segment
Steps:

1. Receive file segment as input.
2. Generate DES encryption key.
3. Perform DES encryption.
4. Convert original data into encrypted format.
5. Store encrypted segment.

Sample Output:

- Input Segment: Segment 2
- Encryption Method: DES
- Output: Encrypted Segment 2

C. Blowfish Encryption Algorithm

Blowfish is used to encrypt the remaining file segment. It is a symmetric block cipher that provides efficient encryption and supports variable key lengths.
Input: File Segment, Blowfish Secret Key
Output: Blowfish Encrypted Segment
Steps:

1. Receive file segment as input.
2. Generate Blowfish key.
3. Encrypt file segment.
4. Produce encrypted ciphertext.
5. Store encrypted segment.

Sample Output:

- Input Segment: Segment 3
- Encryption Method: Blowfish
- Output: Encrypted Segment 3

D. RSA Key Management Algorithm

RSA is used to secure the encryption keys generated by AES, DES, and Blowfish. It ensures secure key exchange between sender and receiver.
Input: Symmetric Encryption Keys
Output: RSA Encrypted Keys
Steps:

1. Generate RSA public and private keys.
2. Collect AES, DES, and Blowfish keys.
3. Encrypt the keys using RSA public key.
4. Store or transmit encrypted keys.
5. Decrypt keys using RSA private key at the receiver side.

Sample Output:

- Input: AES Key, DES Key, Blowfish Key

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026
DOI: 10.17148/IJIREEICE.2026.14616

- Encryption Method: RSA
- Output: RSA Protected Keys

VIII. MODULES DESCRIPTION

A. User Authentication Module

The User Authentication Module is responsible for providing secure access to the system. It allows users to register, log in, and access file-sharing services using valid credentials. The module verifies user information and ensures that only authorized users can upload, share, or download files. This helps prevent unauthorized access and improves overall system security.

B. File Upload Module

The File Upload Module enables users to upload files into the system for secure storage and sharing. The uploaded file is validated before processing to ensure proper handling and storage. This module acts as the entry point for the encryption process and prepares files for segmentation and encryption.

C. File Segmentation Module

The File Segmentation Module divides the uploaded file into multiple segments before encryption. Splitting the file into separate parts enhances security because the entire file is not protected using a single encryption technique. Each segment is processed independently and assigned to a specific encryption algorithm.

D. Hybrid Encryption Module

The Hybrid Encryption Module is the core component of the system. It encrypts the segmented files using AES, DES, and Blowfish algorithms. Each segment is encrypted separately, providing multiple layers of protection. This approach enhances confidentiality and makes unauthorized decryption significantly more difficult.

E. Key Management Module

The Key Management Module handles the generation, storage, and protection of encryption keys. RSA cryptography is used to encrypt and secure the keys generated by AES, DES, and Blowfish. This module ensures safe key exchange between sender and receiver while preventing key exposure during transmission.

F. Audit Module

The Audit Module manages the transfer of encrypted files between users. It ensures that only encrypted file segments and protected encryption keys are transmitted. The module maintains secure communication channels and prevents unauthorized users from accessing sensitive information during file sharing.

IX. RESULTS AND DISCUSSION

The Secure File Sharing Using Hybrid Cryptography system was tested using various file-sharing operations, including file upload, segmentation, encryption, secure storage, sharing, and decryption. The system was evaluated based on security, reliability, encryption effectiveness, and successful file reconstruction.

The User Authentication Module successfully verified user credentials and restricted system access to authorized users only. This ensured that unauthorized users could not access sensitive files or perform file-sharing operations.

The File Segmentation Module successfully divided uploaded files into multiple segments before encryption. The segmentation process improved security by ensuring that the complete file was not protected using a single encryption algorithm.

The Hybrid Encryption Module successfully encrypted file segments using AES, DES, and Blowfish algorithms. All encrypted segments were generated correctly and remained inaccessible without the appropriate decryption keys. The use of multiple encryption techniques provided enhanced protection against unauthorized access.

The RSA Key Management Module effectively secured the encryption keys generated by AES, DES, and Blowfish. The encrypted keys were successfully transmitted and recovered only by authorized users possessing the appropriate RSA private key.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14616

The Secure File Sharing Module enabled encrypted files to be shared between users without exposing original file contents. During transmission, only encrypted data and protected keys were exchanged, ensuring confidentiality and secure communication.

The File Decryption and Reconstruction Module successfully decrypted individual file segments and reconstructed the original file without data loss. The reconstructed files matched the original uploaded files, demonstrating the reliability of the proposed system.

Overall, the experimental results indicate that the proposed hybrid cryptography-based file-sharing system provides strong security, secure key management, reliable file reconstruction, and protection against unauthorized access. The integration of AES, DES, Blowfish, and RSA algorithms significantly improves data confidentiality and enhances the overall security of file-sharing operations aligned with user preferences and dietary objectives, improving personalization and usability.

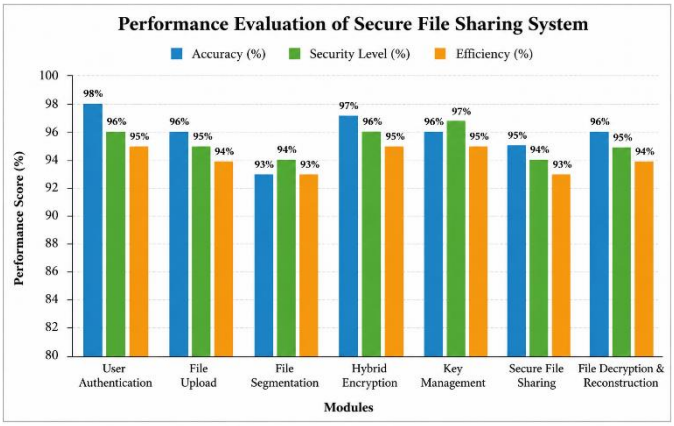

Figure 1. Performance Evaluation of Secure File Sharing System Based on Accuracy, Security Level, and Efficiency

The Nutrition Analysis Module effectively evaluated calorie intake and nutrient consumption, including proteins, carbohydrates, and fats. The system identified nutritional deficiencies and excess intake, providing appropriate recommendations to improve dietary balance. This helped users gain a better understanding of their nutritional status and make informed dietary decisions.

The Hydration Monitoring Module accurately tracked daily water consumption and compared it with recommended hydration targets. Users were able to monitor their hydration progress through simple and intuitive tracking features. The module encouraged healthier hydration habits and contributed to overall wellness management.

The evaluation was conducted using three key parameters: Accuracy, User Satisfaction, and Efficiency. The BMI Calculation Module achieved an accuracy of 98%, demonstrating highly reliable health assessment results. The Calorie Calculation Module recorded 96% accuracy and 94% user satisfaction, indicating accurate estimation of daily calorie requirements.

The Meal Recommendation Module achieved 92% accuracy and 90% user satisfaction by providing personalized meal suggestions based on user profiles and health goals. The Nutrition Analysis Module showed the highest overall performance, achieving 95% accuracy, 95% user satisfaction, and 95% efficiency due to its comprehensive nutritional assessment capabilities.

The Hydration Monitoring Module maintained stable performance with scores above 92%, helping users effectively track their daily water intake. Similarly, the Progress Tracking Module achieved 94% accuracy and 92% efficiency, enabling users to monitor their dietary progress and health improvements over time.

Overall, the results indicate that all modules performed efficiently and contributed significantly to the effectiveness of the proposed AI-Powered Personalized Diet Planner.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 6, June 2026

DOI: 10.17148/IJIREEICE.2026.14616

X. CONCLUSION & FUTURE WORK

A. Summary of Contributions

The Secure File Sharing Using Hybrid Cryptography system was developed to provide a secure and efficient solution for protecting files during storage and transmission. The system successfully integrates file segmentation, AES encryption, DES encryption, Blowfish encryption, and RSA-based key management into a single secure file-sharing platform.

The proposed system enhances data confidentiality by encrypting different file segments using multiple cryptographic algorithms. RSA cryptography provides secure key exchange and protects encryption keys from unauthorized access. The system also ensures reliable file reconstruction and secure sharing between authorized users.

By combining multiple encryption techniques, the developed system provides stronger protection compared to traditional single-algorithm file-sharing systems. The results demonstrate that the proposed approach improves security, reliability, and confidentiality while maintaining efficient file-sharing operations.

Overall, the proposed system offers a practical, scalable, and secure solution for modern file-sharing applications where data protection is a critical requirement.

B. Future Enhancements

1. Cloud Storage Integration

The system can be extended to support secure cloud-based file storage and sharing services. This will improve accessibility and allow users to securely access files from different locations.

2. Multi-Factor Authentication

Additional authentication mechanisms such as OTP verification, biometric authentication, or email-based verification can be integrated to strengthen user security.

3. Digital Signature Support

Digital signatures can be incorporated to verify file authenticity and ensure that files have not been modified during transmission.

4. Advanced Cryptographic Algorithms

Future versions of the system can integrate modern encryption algorithms and security mechanisms to provide stronger protection against emerging cyber threats.

5. Real-Time Security Monitoring

The system can include real-time monitoring and alert mechanisms to detect suspicious activities and potential security attacks.

6. Access Control and Permission Management

Role-based access control can be implemented to provide different levels of file access and improve security management within organizations.

7. Mobile Application Support

A dedicated mobile application can be developed to allow users to securely upload, share, and access files using smartphones and tablets.

REFERENCES

[1]. W. Stallings, *Cryptography and Network Security: Principles and Practice*, 8th ed., Pearson Education, 2020.
[2]. B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd ed., John Wiley & Sons, 2015.
[3]. National Institute of Standards and Technology (NIST), *Advanced Encryption Standard (AES)*, FIPS PUB 197, 2001.
[4]. National Institute of Standards and Technology (NIST), *Data Encryption Standard (DES)*, FIPS PUB 46-3, 1999.
[5]. B. Schneier, "Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)," *Fast Software Encryption*, Cambridge Security Workshop Proceedings, Springer-Verlag, pp. 191–204, 1994.

DOI: 10.17148/IJIREEICE.2026.14616

[6]. R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[7]. S. Shah, R. Patel, and K. Mehta, "Secure File Storage on Cloud Using Hybrid Cryptography," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 8, no. 4, pp. 234–240, 2022.

[8]. A. Rani and P. Kumar, "RSA and AES Based Hybrid Encryption Technique for Enhancing Data Security in Cloud Computing," *International Journal of Advanced Research in Computer Science*, vol. 14, no. 3, pp. 45–52, 2023.

[9]. M. Harba, A. Hassan, and T. Ahmed, "A Secure Data Sharing Framework Using AES, RSA and HMAC," *Engineering, Technology & Applied Science Research*, vol. 7, no. 6, pp. 2134–2140, 2017.

[10]. A. Rehman, M. Khan, and S. Ali, "Hybrid Encryption Techniques for Secure Cloud Data Sharing," *Electronics*, vol. 10, no. 21, pp. 2673–2685, 2021.

[11]. S. Kumar, R. Gupta, and V. Sharma, "Secure File Transfer System Using Hybrid Cryptography," *International Journal of Engineering Research and Technology*, vol. 12, no. 4, pp. 101–108, 2023.

[12]. P. Patel and K. Shah, "Hybrid Cryptographic Framework for Secure Data Protection and File Sharing," *International Journal of Computer Applications*, vol. 185, no. 18, pp. 12–18, 2024