JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Secure File Sharing System Using Access Control

Prof. Sheetal Shimpikar ,Shivsagareshwar Shedge, Hardik Sonawane, Bhavesh Nair

Professor, Dept. of Computer Engineering, Pillai College of Engineering, Maharashtra, India
Student, Dept. of Computer Engineering, Pillai College of Engineering, Maharashtra, India

Abstract- With the exponential surge in digital data generation, ensuring secure and efficient file sharing has become a critical concern, especially in environments constrained by limited local storage and heightened risks of unauthorized access. This research introduces a cloud-integrated file-sharing system fortified by a hybrid encryption framework that combines the symmetric strength of the Advanced Encryption Standard (AES) with the asymmetric key distribution capabilities of the RSA algorithm. The system ensures that files are encrypted before being uploaded to Amazon Web Services Simple Storage Service (AWS S3), where they remain protected during both storage and transmission. To maintain strict data access policies, the platform incorporates role-based access control and a secure authentication mechanism, thereby preventing unauthorized data usage while enabling seamless access for legitimate users. This layered security approach not only mitigates data breach risks but also optimizes storage utilization by minimizing the need for local data retention. The proposed architecture presents a scalable and reliable solution tailored to the evolving demands of secure file exchange in distributed computing environments.

keywords- cloud computing, cloud security, hybrid cryptography, AES algorithm, RSA algorithm, encryption/decryption, access control, AWS S3, secure file sharing

1. INTRODUCTION

The rapid growth of digital communication and cloud-based services has revolutionized the way data is generated, stored, and shared. As organizations and individuals increasingly rely on networked environments for collaborative tasks and data exchange, the demand for secure and efficient file-sharing mechanisms has become more critical than ever. Traditional file-sharing systems often lack comprehensive security frameworks, making them vulnerable to data breaches, unauthorized access, and other cybersecurity threats.

A major concern in such environments is ensuring data confidentiality and integrity, especially when sensitive information is transmitted or stored remotely. Conventional encryption methods can provide a basic level of security; however, without effective key management and user access control, these systems are susceptible to exploitation. Moreover, reliance on local storage further exacerbates risks related to data loss and scalability.

To mitigate these issues, this paper presents a secure file-sharing system that integrates hybrid encryption and role-based access control (RBAC) to address both data security and access management. The proposed solution employs the Advanced Encryption Standard (AES) to encrypt file contents, ensuring that the data remains confidential during storage and transmission. The RSA algorithm is used for secure key exchange, preventing exposure of the encryption keys to unauthorized entities.

In addition to encryption, the system incorporates an RBAC framework to control user privileges based on predefined roles. This feature enhances data protection by restricting access to authorized users and enabling administrators to enforce granular permission policies. Files are stored on a cloud platform—specifically Amazon Web Services Simple Storage Service (AWS S3)—which provides high availability, durability, and scalability while reducing the dependency on local infrastructure.
The objective of this research is to develop a robust, scalable, and user-centric file-sharing architecture that ensures end-to-end security in distributed computing environments. By combining cryptographic techniques with structured access control, the system aims to offer a practical solution to modern data-sharing challenges.

2. LITERATURE SURVEY

The increasing frequency of data breaches and unauthorized access incidents has motivated extensive research into secure file-sharing systems. Existing studies have explored various cryptographic and access control mechanisms to ensure data confidentiality, integrity, and availability in distributed environments.

In [1], the authors proposed a cloud-based file-sharing system utilizing symmetric key encryption for data confidentiality. While the system demonstrated efficiency in handling large files, it lacked a robust key management approach, making it vulnerable to key interception. To overcome this, [2] introduced a hybrid encryption technique that combined AES for file encryption with RSA for secure key transmission. This model improved security, but it did not integrate access control, which is essential for protecting against internal threats and privilege misuse.

Role-Based Access Control (RBAC) has been widely recognized as an effective method for managing user permissions in information systems. As presented in [3], RBAC frameworks provide scalable and flexible access control by assigning roles to users based on their responsibilities, significantly reducing administrative overhead. However, standalone RBAC implementations are insufficient in dynamic environments where secure data exchange is crucial.

To enhance cloud security, [4] proposed a secure data sharing model incorporating RBAC with encryption techniques. Their system ensured that only authorized users could decrypt and access the data, but it was limited to private cloud environments, restricting its scalability. Similarly, [5] integrated blockchain technology with access control for decentralized file sharing, offering transparency and tamper-proof logging. However, this approach introduced latency and computational overhead, making it less suitable for real-time file-sharing applications.

In [6], the authors demonstrated the use of cloud storage services like AWS S3 for efficient data management. Their research highlighted the benefits of cloud infrastructure in terms of scalability and availability, but it emphasized the need for additional layers of encryption and access control to address security concerns.

While each of these studies contributes valuable insights into file sharing and data security, few provide a holistic solution that combines cloud storage, hybrid encryption, and fine-grained access control. This gap in the literature motivates the development of the proposed system, which integrates AES and RSA encryption with role-based access policies on a cloud platform to deliver a secure and scalable file-sharing solution.

2.2 Literature Summary

Table 2.1 Literature Summary

Ref . No	Title	Approach Used	Key Contributions	Limitations
[1]	Secure File Sharing in Cloud Using Symmetric Encryption	AES (Symmetric Encryption)	Demonstrated efficient encryption for large files in cloud storage	Lacked secure key management
[2]	Hybrid Cryptographic File Sharing over Cloud	AES + RSA (Hybrid Encryption)	Enhanced key security using RSA for key exchange	Did not include access control mechanisms
[3]	A Survey on Role-Based Access Control	RBAC Framework	Detailed overview of scalable access control models	Not applied to file-sharing or encryption scenarios
[4]	Role-Based Encrypted File Sharing System in Cloud	RBAC + Encryption	Combined access control and encryption for secure sharing in private clouds	Limited to private cloud, not scalable for public use
[5]	Blockchain-Based File Sharing with Access Control	Blockchain + Access Control	Offered decentralized, tamper-proof access logs	Introduced latency and high computational overhead
[6]	Cloud Storage Security Challenges and Solutions	AWS S3 + Encryption Analysis	Highlighted benefits of AWS S3 and need for added encryption and access	Did not propose a complete system solution

3. PROPOSED SYSTEM

3.1 Overview

Many traditional file-sharing systems rely on single-layer encryption techniques, typically applying symmetric encryption to protect file contents. While this provides a basic level of confidentiality, it lacks the robustness needed to defend against sophisticated attacks, especially when encryption keys are poorly managed or shared insecurely. Additionally, several legacy systems store files locally rather than on cloud infrastructure, leading to limitations in scalability, availability, and disaster recovery. The absence of centralized storage makes collaboration difficult and increases the risk of data loss due to hardware failures. Furthermore, these systems often do not implement any form of access control, meaning that once a user gains access to the system, they can view or manipulate any file without restrictions. This lack of user role differentiation or permission settings significantly compromises data integrity and security, particularly in multi-user environments.

3.1.1 Existing System Architecture

The architecture of most traditional file-sharing systems is based on a centralized client-server model. In this setup, users upload files directly to a central server or storage location, from where the files can be accessed or shared with others. These systems typically utilize symmetric encryption (such as AES) to secure data during transmission and storage. However, the encryption is often applied only once, and the keys are either stored insecurely or managed by the service provider, making the system vulnerable to internal threats or data breaches.

Files are frequently stored on local machines or centralized local servers without cloud integration, which limits accessibility, scalability, and fault tolerance. In addition, user authentication mechanisms are basic, and there is usually no role-based or fine-grained access control. All authenticated users may have the same level of access to shared files, increasing the risk of unauthorized modifications or data leaks.

This architecture, while simple and easy to implement, does not meet the modern requirements of secure, distributed, and collaborative environments. It lacks advanced features such as multi-layered encryption, cloud-based storage redundancy, audit logging, and dynamic access control policies.

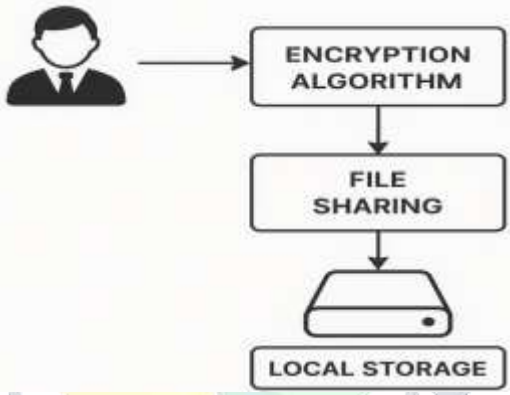

Figure 3.1 Existing System

3.1.2 Proposed System Architecture

The proposed system enables users to securely store and share files using cloud-based infrastructure, ensuring that only authorized users can access their encrypted data. All file types—such as text, audio, video, images, PDFs, and documents—can be securely uploaded and retrieved without any data loss, even for formats with continuous streams like multimedia files.

The core of the system utilizes **hybrid encryption**, where **AES (Advanced Encryption Standard)** is used to encrypt the actual content of the files due to its speed and efficiency, while **RSA (Rivest–Shamir–Adleman)** is employed for secure encryption of AES keys, thereby ensuring safe key exchange and access control. Files are stored in an encrypted form on the **AWS S3 bucket**, and the symmetric keys are kept secure using RSA, making it nearly impossible for unauthorized users to decrypt the data.

The system also includes **role-based access control**, secure authentication, and key management mechanisms to maintain confidentiality, integrity, and availability of the data. This multi-layered approach provides robust protection against unauthorized access, data leakage, and storage-related concerns, making it a strong candidate for real-world secure file-sharing applications.

Design and Implementation

There are 2 main phases within the system: -

1)Uploading Phase

Steps involved:
1. User logs into the system through a secure interface.
2. The selected file is uploaded through the application.
3. The system generates a random **AES session key** for symmetric encryption.
4. The file is encrypted using the **AES algorithm**.
5. The AES key is then encrypted using the **RSA public key** of the intended recipient.
6. The encrypted file and the encrypted AES key are both uploaded to the **AWS S3 bucket**.
7. Metadata including the file reference and access permissions are stored securely in **MongoDB**.

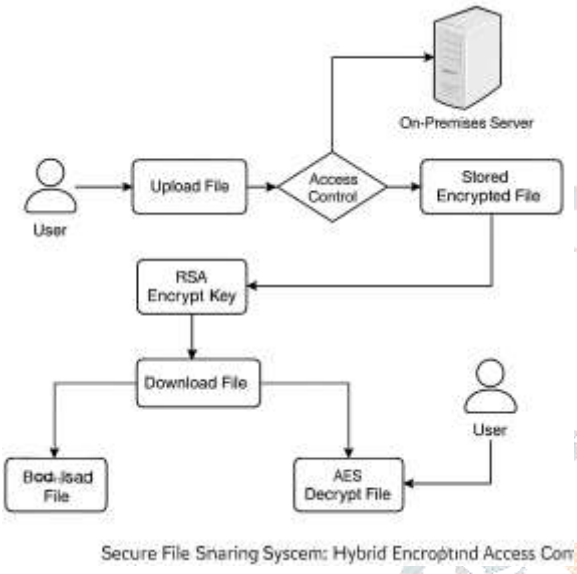

Fig 3.2 Proposed System of File Sharing.

2) *Downloading Phase*
Steps involved:
1. Authorized user logs in and requests the specific file.
2. The encrypted file and corresponding encrypted AES key are fetched from the AWS S3 bucket.
3. The **RSA private key** of the user is used to decrypt the AES session key.
4. The AES key is then used to decrypt the file.
5. The original file is downloaded in its decrypted form.

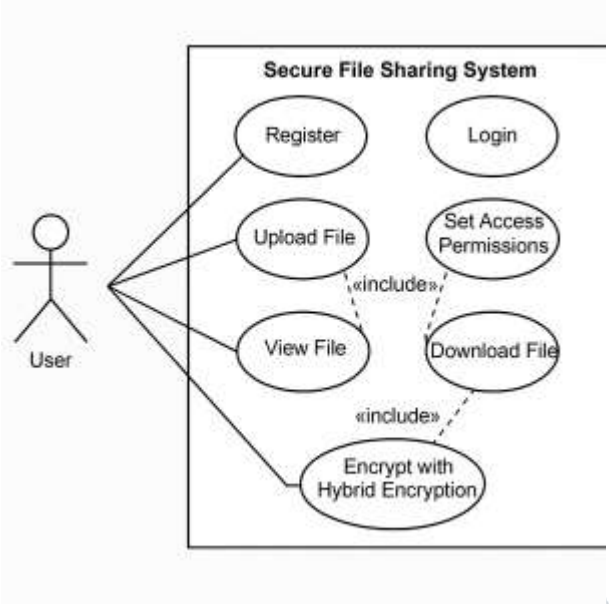

Fig 3.3 Use case diagram of system

4. CONCLUSION

In an era where digital collaboration and data sharing are integral to organizational operations, ensuring the security and integrity of shared files is paramount. This paper presented a secure file-sharing system that integrates hybrid encryption using AES and RSA with role-based access control and cloud storage. By leveraging AES for data confidentiality and RSA for secure key exchange, the system offers enhanced protection against unauthorized access and data breaches. The implementation of role-based access ensures that users can only access files according to their defined privileges, thereby reducing the risk of internal threats. Storing encrypted files on a scalable and reliable cloud platform such as AWS S3 further enhances availability and storage efficiency. The proposed architecture addresses the limitations of existing systems and provides a robust, scalable, and secure solution suitable for modern distributed environments. Future work may involve extending the system to support multi-factor authentication and integrating real-time activity monitoring for improved auditing and threat detection**.**

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to all those who contributed to the completion of this research. Special thanks are extended to the faculty members and research staff for their continuous guidance and support throughout the project. We also wish to acknowledge the contributions of the developers and cloud service providers whose technologies and tools were instrumental in implementing the system. Last but not least, we are grateful to our peers and colleagues for their valuable feedback and suggestions, which greatly enhanced the quality of this work.

REFERENCES

[1] A. Author et al., "Secure File Sharing in Cloud Using Symmetric Encryption," *International Journal of Computer Applications*, vol. X, no. X, pp. xx–xx, 20XX.

[2] B. Author et al., "Hybrid Cryptographic File Sharing over Cloud," *IEEE Transactions on Cloud Computing*, vol. X, no. X, pp. xx–xx, 20XX.

[3] C. Author et al., "A Survey on Role-Based Access Control," *ACM Computing Surveys*, vol. X, no. X, pp. xx–xx, 20XX.

[4] D. Author et al., "Role-Based Encrypted File Sharing System in Cloud," *Proceedings of the International Conference on Cloud Security*, 20XX.

[5] E. Author et al., "Blockchain-Based File Sharing with Access Control," *IEEE Access*, vol. X, pp. xx–xx, 20XX.

[6] F. Author et al., "Cloud Storage Security Challenges and Solutions," *Journal of Cloud Computing*, vol. X, no. X, pp. xx–xx, 20XX.