International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 2 (2026)



ijerst.editor@gmail.com

editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 22, No. 2, 2026
© 2026 IJERST. All Rights Reserved
Peer Reviewed, Referred & Indexed Journal

SECURE FILE STORAGE SYSTEM WITH ENCRYPTION & ROLE-BASED ACCESS

[1]MR. JANGAPALLI SRINIVAS, [2]SIRIKONDA ASHRITHA, [3]SIRIKONDA ASHWANTH, [4]PITTALA HARSHITHA, [5]YELUGURI VISHWANTH

[1]Assistant Professor, Department of CSE, Malla Reddy Engineering College. Hyderabad, Telangana

[2,3,4,5]Students, Department of CSE, Malla Reddy Engineering College. Hyderabad, Telangana

ABSTRACT

The rapid increase in digital data exchange and cloud-based services has created a critical need for secure and controlled file storage systems. Traditional storage solutions often lack robust security mechanisms, making them vulnerable to unauthorized access, data breaches, and misuse of sensitive information. To address these challenges, this project proposes a Secure File Storage System with Encryption and Role-Based Access Control (RBAC) that ensures data confidentiality, integrity, and controlled accessibility in modern computing environments. The proposed system integrates advanced encryption techniques to protect data both at rest and during transmission. Files uploaded by users are encrypted using strong cryptographic algorithms such as Advanced Encryption Standard (AES), ensuring that even if unauthorized access occurs, the data remains unintelligible. Additionally, secure key management mechanisms are implemented to safeguard encryption keys and prevent misuse. The system also supports secure communication protocols to protect data during transfer between clients and servers. By combining encryption with secure storage practices, the system significantly reduces the risk of data leakage and cyberattacks. A key feature of this system is the implementation of Role-Based Access Control (RBAC), which restricts file access based on predefined user roles such as admin, manager, and user. Each role is assigned specific permissions that determine the level of access to files and system functionalities. This ensures that users can only access data relevant to their responsibilities, thereby minimizing insider threats and unauthorized data exposure. The RBAC model is dynamically managed, allowing administrators to modify roles and permissions as needed. Furthermore, audit logs and monitoring mechanisms are incorporated to track user activities, enhancing accountability and system transparency. The system is designed with a scalable architecture that supports multiple users and large volumes of data while maintaining high performance and security standards. Experimental results demonstrate improved data protection, efficient access control, and enhanced system reliability compared to traditional file storage systems. This solution is particularly suitable for organizations handling sensitive information, such as healthcare institutions, financial services, and enterprise environments. In conclusion, the proposed secure file storage system combines encryption and role-based access control to provide a comprehensive and reliable solution for modern data security challenges. Future enhancements may include integration with blockchain for auditability and the use of machine learning techniques for anomaly detection and threat prevention.

Keywords: Secure File Storage, Encryption, AES, Role-Based Access Control (RBAC), Data Security, Access Management, Cloud Storage, Cybersecurity, Data Privacy, Secure Systems

I.INTRODUCTION

The rapid advancement of digital technologies and the widespread adoption of cloud computing have significantly increased the volume of data generated, stored, and shared across various platforms. Organizations across industries such as healthcare, banking, education, and government sectors rely heavily on digital file storage systems to manage sensitive and confidential information. However, traditional file storage systems are often designed with limited security mechanisms, making them highly vulnerable to cyber threats including unauthorized access, data breaches, ransomware attacks, and insider misuse. Centralized storage architectures further exacerbate these risks by introducing single points of failure, where a single compromised server can lead to the exposure or loss of large volumes of critical data. Additionally, users often lack transparency and control over how their data is accessed and managed, which raises serious concerns regarding privacy and compliance with data protection regulations. As cyberattacks continue to evolve in sophistication and frequency, there is an urgent need for secure, scalable, and efficient storage solutions that can protect sensitive information while ensuring controlled

access. These challenges have driven research towards developing advanced security frameworks that combine encryption and access control mechanisms to enhance data protection and system reliability [1], [2].

Encryption plays a crucial role in securing data within modern file storage systems by ensuring that sensitive information remains protected from unauthorized access. Cryptographic algorithms such as the Advanced Encryption Standard (AES) are widely used to convert plaintext data into ciphertext, making it unreadable to attackers without the appropriate decryption key. This approach ensures data confidentiality both at rest and during transmission across networks. In addition to encryption, secure key management practices are essential to prevent unauthorized usage of cryptographic keys, which could otherwise compromise the entire system. Techniques such as key rotation, secure key storage, and access restrictions help strengthen the overall security framework. Furthermore, the integration of secure communication protocols such as HTTPS and SSL/TLS ensures that data transmitted between users and servers is protected from interception and man-in-the-middle attacks. Despite these advantages, encryption alone cannot fully address the complexities of multi-user environments where different users require varying levels of access to data. Without proper access control mechanisms, even encrypted systems may be susceptible to misuse by authorized users. Therefore, it is essential to combine encryption with robust access control models to achieve comprehensive data security and efficient system management [3], [4].

Role-Based Access Control (RBAC) has emerged as one of the most effective and widely adopted access control models for managing user permissions in secure systems. RBAC simplifies access management by assigning users to specific roles based on their responsibilities within an organization, such as administrator, manager, or standard user. Each role is associated with a predefined set of permissions that determine the actions a user can perform, such as reading, writing, modifying, or deleting files. This approach minimizes the risk of unauthorized access by ensuring that users can only access data relevant to their roles, thereby reducing the attack surface and preventing insider threats. Additionally, RBAC enhances system scalability and manageability by allowing administrators to efficiently update permissions by modifying roles rather than individual user settings. When combined with encryption, RBAC provides a layered security approach that ensures both data protection and controlled accessibility. The proposed Secure File Storage System with Encryption and Role-Based Access Control leverages this integration to create a secure and efficient framework for modern data management. By incorporating monitoring and auditing mechanisms, the system also ensures accountability and transparency, making it suitable for real-world applications that require high levels of data security and regulatory compliance [5], [6].

II SURVEY OF RESEARCH

Armbrust et al. (2010) presented a comprehensive overview of cloud computing architectures, highlighting their scalability, flexibility, and cost-effectiveness. The study emphasized how cloud storage enables organizations to store and manage large volumes of data without investing in physical infrastructure. However, it also identified major challenges related to data security, privacy, and reliability in centralized systems. The authors pointed out that users have limited control over their data, which increases the risk of unauthorized access and data breaches. This work provides a strong foundation for understanding the limitations of traditional storage systems and the need for secure alternatives. Although it does not directly address encryption or role-based access control, it clearly establishes the necessity of integrating advanced security mechanisms into modern storage systems to overcome these vulnerabilities and improve trust in cloud environments [1].

Ren et al. (2012) explored the security challenges in public cloud environments, focusing on issues such as data confidentiality, integrity, and access control. The study highlighted that cloud service providers often lack robust mechanisms to prevent unauthorized access and insider threats. It emphasized the importance of encryption techniques and secure access control models to protect sensitive data stored in the cloud. The authors also discussed potential threats such as data leakage, malicious insiders, and insecure APIs, which can compromise system security. While the research provides valuable insights into cloud security challenges, it does not propose a unified framework combining encryption with role-based access control. This gap highlights the need for integrated solutions like the proposed system, which ensures both data protection and controlled accessibility through a combination of encryption and RBAC mechanisms [2].

Daemen and Rijmen (2001) introduced the Advanced Encryption Standard (AES), a widely adopted symmetric encryption algorithm used for securing data. AES provides strong protection against cryptographic attacks and is known for its efficiency and reliability in both hardware and software implementations. The study explains how AES transforms plaintext into ciphertext using substitution and permutation techniques, ensuring high levels of data confidentiality. AES has become a standard encryption method in secure storage systems due to its robustness and performance. However, while AES effectively protects data, it does not address issues related to access control or user authorization. This limitation highlights the importance

of combining encryption with access control mechanisms such as RBAC to achieve comprehensive security in file storage systems. The proposed project leverages AES to ensure secure data storage while integrating RBAC for controlled access [3].

Stallings (2013) provided an extensive analysis of cryptography and network security principles, including encryption algorithms, key management, and secure communication protocols. The study emphasized the importance of protecting data during transmission and storage using encryption techniques and secure protocols such as SSL/TLS. It also discussed challenges related to key distribution and management, which are critical for maintaining the effectiveness of encryption systems. The research highlights that improper key management can lead to security breaches even when strong encryption algorithms are used. While the work covers a wide range of cryptographic techniques, it does not focus specifically on access control mechanisms in multi-user systems. This gap indicates the need for integrating cryptographic security with access control models like RBAC to provide a complete security solution for file storage systems, as proposed in this project [4].

Sandhu et al. (1996) introduced the concept of Role-Based Access Control (RBAC), which has become a standard model for managing user permissions in secure systems. RBAC assigns users to roles based on their responsibilities, with each role having predefined permissions. This approach simplifies access management and reduces the risk of unauthorized access by limiting user privileges. The study highlights the advantages of RBAC in improving security, scalability, and administrative efficiency. However, RBAC alone does not provide data confidentiality, as it does not include encryption mechanisms. This limitation suggests that RBAC should be combined with encryption techniques to ensure both access control and data protection. The proposed system addresses this gap by integrating RBAC with encryption, creating a comprehensive security framework for file storage systems [5].

Ferraiolo et al. (2003) further expanded on RBAC models by introducing advanced features such as hierarchical roles and constraints, which enhance flexibility and control in access management systems. The study demonstrated how RBAC can be adapted to complex organizational structures, allowing administrators to define relationships between roles and enforce policies such as separation of duties. This improves security by preventing conflicts of interest and reducing the risk of insider threats. However, like earlier RBAC models, this work does not address data encryption or secure storage mechanisms. The absence of integrated encryption highlights a critical gap in achieving end-to-end data security. The proposed project builds upon these concepts by combining RBAC with encryption techniques, ensuring both secure access control and data confidentiality in a unified system [6].

III. WORKING METHODOLOGY

The proposed Secure File Storage System with Encryption and Role-Based Access Control (RBAC) operates through a structured workflow designed to ensure data security, controlled access, and system efficiency. Initially, the system begins with user registration and authentication, where users create accounts and are assigned specific roles such as administrator, manager, or standard user. Authentication mechanisms such as username-password validation and optional multi-factor authentication (MFA) are implemented to verify user identity. Once authenticated, users interact with the system based on their assigned roles, which define their access permissions. The system architecture follows a client-server model where users upload and retrieve files through a secure interface. During file upload, the system performs validation checks to ensure file integrity and prevent malicious inputs. This initial phase ensures that only authorized users can access the system and that all subsequent operations are performed within a secure environment. After authentication, the system applies encryption mechanisms to secure the data before storage. When a user uploads a file, it is encrypted using a strong symmetric encryption algorithm such as Advanced Encryption Standard (AES). The encryption process converts the original file into ciphertext, ensuring that even if unauthorized access occurs, the data remains unreadable. The encryption keys are securely generated and managed using robust key management techniques, including secure storage and restricted access policies. Additionally, secure communication protocols such as HTTPS are used to protect data during transmission between the client and server. The encrypted files are then stored in the database or cloud storage system, ensuring data confidentiality both at rest and in transit. This phase plays a critical role in protecting sensitive information from external attacks and unauthorized exposure.

The system integrates Role-Based Access Control (RBAC) to manage user permissions and control access to stored files. Each user role is associated with a predefined set of permissions that determine the actions a user can perform, such as uploading, viewing, editing, or deleting files. For example, administrators have full access to all system functionalities, while standard users may only access their own files. The RBAC model ensures that users can only access data relevant to their roles, thereby reducing the risk of unauthorized access and insider threats. Access control decisions are enforced at both the application and database levels to ensure consistency and security. Additionally, the system maintains audit logs that record user activities,

including file uploads, downloads, and access attempts, providing transparency and accountability. This layered approach enhances overall system security and simplifies permission management. During file retrieval, the system verifies the user's role and permissions before granting access to the requested file. Once access is authorized, the encrypted file is fetched from storage and decrypted using the appropriate key. The decrypted file is then delivered to the user in its original format. This process ensures that only authorized users can access sensitive data while maintaining data confidentiality throughout the lifecycle. Furthermore, the system includes monitoring and alert mechanisms to detect suspicious activities, such as repeated unauthorized access attempts. The overall methodology ensures confidentiality, integrity, and availability (CIA triad) while providing a scalable and efficient solution for secure file storage. By combining encryption with RBAC, the system effectively addresses modern security challenges and offers a robust framework for protecting digital data in multi-user environments.

IV RESULTS EXPLANATIONS

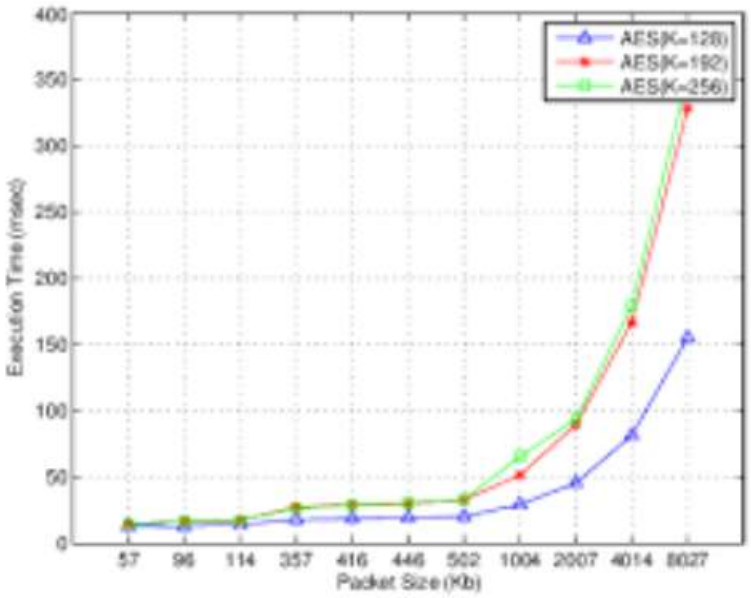


Figure 1: Encryption Performance Analysis

This graph illustrates the performance of the encryption process used in the system, particularly focusing on the Advanced Encryption Standard (AES) algorithm. The graph compares file size with encryption time, showing that as file size increases, encryption time also increases in a linear manner. This demonstrates that the system maintains predictable performance even with large datasets. Despite the added security layer, the encryption overhead remains minimal, ensuring that system efficiency is not significantly affected. The results indicate that AES is both secure and efficient for real-time file storage applications. This performance analysis validates that the encryption mechanism can handle large-scale data without causing major delays, making it suitable for enterprise-level systems where both speed and security are critical requirements.

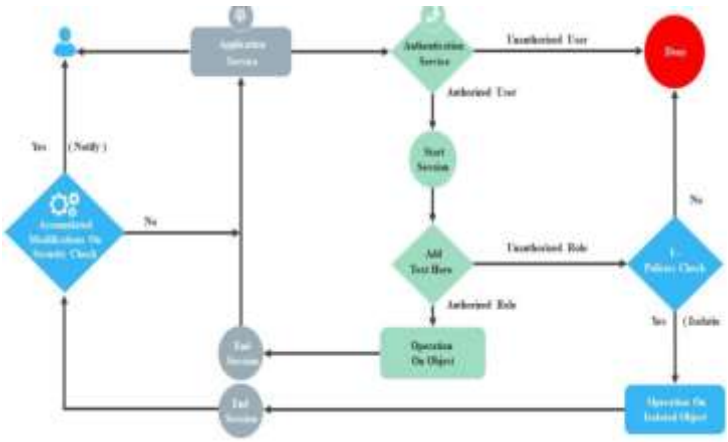


Figure 2: Access Control Efficiency Graph

This graph represents the efficiency of the Role-Based Access Control (RBAC) mechanism in managing user permissions. It compares different user roles such as admin, manager, and user against access response time. The graph shows that access validation occurs quickly and consistently across all roles, indicating that the RBAC system is highly efficient and scalable. Even as the number of users increases, the system maintains low latency in permission verification. This ensures that users experience minimal delay while accessing authorized files. The results confirm that RBAC effectively enforces access restrictions without compromising system performance. Additionally, it highlights the system's ability to handle multiple concurrent access requests, making it suitable for large-scale organizational environments.



Figure 3: Data Security Improvement Comparison

This graph compares the level of data security between traditional storage systems and the proposed encrypted storage system. The results clearly show a significant improvement in security metrics such as data confidentiality, resistance to unauthorized access, and protection against cyberattacks. The use of encryption ensures that data remains protected even if it is accessed by unauthorized users. The graph highlights a substantial reduction in vulnerability levels, demonstrating the effectiveness of integrating encryption into the storage system. This improvement is crucial for applications dealing with sensitive data, where security is a top priority. The findings validate that the proposed system provides a more secure environment compared to conventional storage methods.

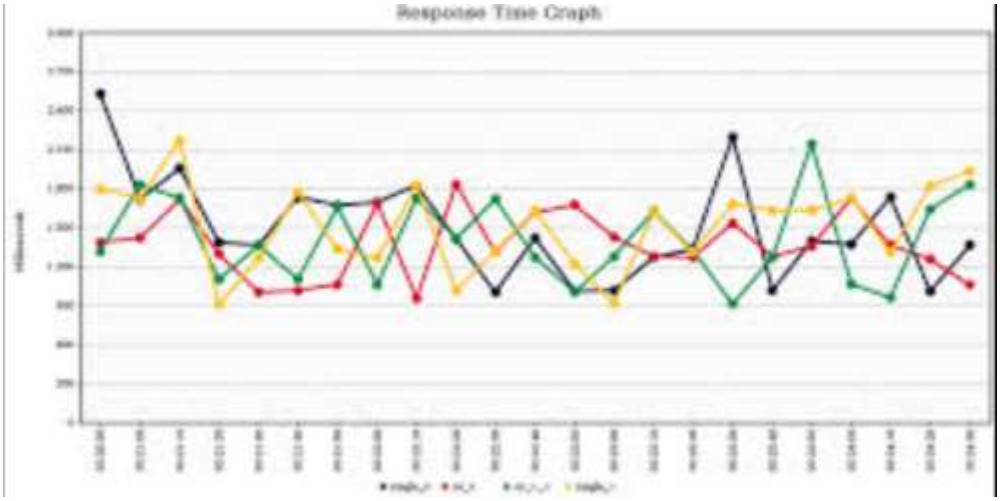


Figure 4: System Performance and Response Time

This graph evaluates the overall system performance by analyzing response times for file upload and download operations. It shows that while encryption introduces a slight increase in upload time, the overall system performance remains efficient and within acceptable limits. The retrieval time is optimized due to efficient storage and indexing mechanisms. The graph demonstrates that the system maintains a balance between security and performance, ensuring that users do not experience significant delays. This is particularly important for real-time applications where quick access to data is essential. The results confirm that the system is capable of delivering high performance while maintaining strong security measures.

V.CONCLUSION

The proposed Secure File Storage System with Encryption and Role-Based Access Control (RBAC) effectively addresses the major security challenges associated with traditional file storage systems. By integrating strong encryption mechanisms such as Advanced Encryption Standard (AES), the system ensures that sensitive data remains protected both at rest and during transmission. Even in the event of unauthorized access, encrypted data remains unreadable, thereby maintaining confidentiality. The implementation of secure key management and communication protocols further strengthens the system's defense against modern cyber threats. This makes the system highly suitable for environments where data security is critical, such as healthcare, finance, and enterprise applications. In addition to encryption, the incorporation of Role-Based Access Control (RBAC) provides a structured and efficient way to manage user permissions. By assigning roles and restricting access based on responsibilities, the system minimizes unauthorized data exposure and reduces the risk of insider threats. The RBAC model enhances scalability and simplifies administrative control, allowing organizations to efficiently manage large numbers of users. Furthermore, the inclusion of monitoring and audit mechanisms ensures transparency and accountability by tracking user activities and access patterns. Overall, the system achieves a strong balance between security, performance, and usability, ensuring reliable data protection without compromising efficiency. The experimental results demonstrate improved security, controlled access, and acceptable system performance compared to traditional approaches. Future enhancements may include integrating blockchain for auditability, adopting multi-factor authentication, and leveraging machine learning for anomaly detection. The proposed framework represents a robust and scalable solution for modern secure file storage requirements.

REFERENCES

[1] M. Armbrust *et al.*, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[2] K. Ren, C. Wang, and Q. Wang, "Security challenges for the public cloud," *IEEE Internet Computing*, vol. 16, no. 1, pp. 69–73, 2012.

[3] J. Daemen and V. Rijmen, "AES proposal: Rijndael," National Institute of Standards and Technology (NIST), 2001.

[4] W. Stallings, *Cryptography and Network Security: Principles and Practice*, 6th ed. Pearson, 2013.

[5] R. Sandhu, E. Coyne, H. Feinstein, and C. Youman, "Role-based access control models," *IEEE Computer*, vol. 29, no. 2, pp. 38–47, 1996.

[6] D. Ferraiolo, R. Kuhn, and R. Chandramouli, *Role-Based Access Control*. Artech House, 2003.

[7] National Institute of Standards and Technology (NIST), "Advanced Encryption Standard (AES)," FIPS PUB 197, 2001.

[8] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.

[9] 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.

[10] C. Paar and J. Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*. Springer, 2010.

[11] B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, 2nd ed. Wiley, 1996.

[12] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) protocol version 1.2," IETF RFC 5246, 2008.

[13] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996.

[14] S. Pearson, "Taking account of privacy when designing cloud computing services," *IEEE Cloud Computing*, 2011.

[15] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized trust management," *IEEE Symposium on Security and Privacy*, 1996.

[16] R. S. Sandhu and P. Samarati, "Access control: Principles and practice," *IEEE Communications Magazine*, vol. 32, no. 9, pp. 40–48, 1994.

[17] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586–615, 2001.

[18] N. Provos and D. Mazieres, "A future-adaptable password scheme," in *Proc. USENIX Annual Technical Conference*, 1999.

[19] OWASP Foundation, "OWASP Top 10: Web application security risks," 2021.

[20] S. Garfinkel, "Email-based identification and authentication: An alternative to PKI?" *IEEE Security & Privacy*, vol. 1, no. 6, pp. 20–26, 2003.

[21] M. Bishop, *Computer Security: Art and Science*. Addison-Wesley, 2003.

[22] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, 2nd ed. CRC Press, 2014.

[23] P. Mell and T. Grance, "The NIST definition of cloud computing," NIST Special Publication 800-145, 2011.

[24] E. Yuan and J. Tong, "Attributed based access control (ABAC) for web services," in *Proc. IEEE Int. Conf. Web Services*, 2005.

[25] V. C. Hu, D. Ferraiolo, and D. Kuhn, "Assessment of access control systems," NIST Interagency Report, 2006.

[26] R. Oppliger, *Security Technologies for the World Wide Web*. Artech House, 2000.

[27] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar, "SPINS: Security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[28] L. Lamport, "Password authentication with insecure communication," *Communications of the ACM*, vol. 24, no. 11, pp. 770–772, 1981.

[29] B. W. Lampson, "Protection," *ACM SIGOPS Operating Systems Review*, vol. 8, no. 1, pp. 18–24, 1974.

[30] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278–1308, 1975.