International Journal of Engineering Research and Science & Technology

# FOG COMPUTING BASED SECURE FILE MANAGEMENT SYSTEM USING AES ENCRYPTION

Mrs.N. Sreeja,[1] Reena Sri Yalamandala[2], Papagari Sandhya Padmavathi[3], Manda Anjali[4], Muppala Deepa[5]
1. Assistant professor, Department of Computer Science and Engineering (Data Science) Vignan's Institute of Management and Technology for Women, Hyderabad
2, 3, 4, 5 B-Tech student, Department of Computer Science and Engineering (Data Science), Vignan's Institute of Management and Technology for Women, Hyderabad
Email: reenasriyalamanadala@gmail.com

**ABSTRACT:**

In recent years, secure data management has become a major concern due to the rapid growth of digital information. Traditional cloud-based systems often face issues such as delay and security risks, making them less efficient for real-time applications. This paper presents a fog computing based secure file management system that performs data processing locally to improve performance and reduce latency. The proposed system uses the Advanced Encryption Standard (AES) algorithm to ensure data security by encrypting files before storage. Role-based access control is implemented to restrict user permissions, allowing only authorized users to perform specific operations. In addition, a multi-layer security mechanism is introduced by combining user authentication with a decryption password for enhanced protection. The system also includes an audit logging feature to record user activities such as login, file upload, and file download. Experimental results demonstrate that the system provides improved security, faster processing, and efficient file handling. The proposed approach offers a reliable and scalable solution for secure data management in various applications.

**Keywords:** Fog computing, AES Encryption, Data Security, File Management, Role-Based Access Control

## I. INTRODUCTION

In the modern digital era, the rapid growth of data has increased the need for secure and efficient data management systems. Organizations and individuals rely heavily on cloud computing for storing and managing data. However, cloud-based systems often face challenges such as high latency, security risks, and dependency on remote servers. These limitations make cloud systems less suitable for applications that require real-time processing and strong data protection. Fog computing has emerged as an effective solution to overcome these limitations by bringing data processing closer to the user. Unlike cloud computing, fog computing performs operations locally, reducing delay and improving system performance. It also enhances data privacy by minimizing the need to transfer sensitive information over the internet. In this paper, a fog computing based secure file management system is proposed to address the challenges of data security and access control. The system uses the Advanced Encryption Standard (AES) algorithm to encrypt files before storage, ensuring that sensitive data remains protected. In addition, role-based access control is implemented to restrict user permissions, allowing only authorized users to access specific functionalities. The proposed system also incorporates a multi-layer security mechanism by combining user authentication with a decryption password. This ensures that even if login credentials are compromised, the data remains secure.

Furthermore, an audit logging feature is included to record user activities, which helps in monitoring system usage and maintaining accountability. The main objective of this work is to develop a secure, efficient, and user-friendly file management system that reduces latency, enhances data protection, and provides controlled access. The proposed approach offers a reliable solution for secure data management in various real-world applications such as educational institutions, organizations, and enterprises.

## II. RELATED WORK

In recent years, several approaches have been proposed to improve data security and file management systems. Traditional cloud-based storage systems are widely used due to their scalability and accessibility. However, these systems often suffer from issues such as high latency, data breaches, and lack of user control over stored data. Many studies have focused on improving cloud security using encryption techniques, but they still rely heavily on remote servers, which increases response time. Encryption techniques such as the Advanced Encryption Standard (AES) have been widely used to secure data in storage systems. AES provides strong encryption and ensures data confidentiality, making it one of the most preferred methods for protecting sensitive information. However, encryption alone is not sufficient if proper access control mechanisms are not implemented. Several research works have introduced role-based access control systems to restrict user permissions and improve data security. These systems ensure that users can only access the data relevant to their role, thereby reducing the risk of unauthorized access. However, many existing systems lack multi-layer security, which makes them vulnerable to attacks if authentication credentials are compromised. Fog computing has recently gained attention as an alternative to cloud computing for reducing latency and improving performance. By processing data closer to the user, fog computing minimizes delays and enhances real-time processing capabilities. Some studies have explored the use of fog computing in secure data management, but they often lack proper integration with encryption and access control mechanisms. The proposed system addresses these limitations by combining fog computing, AES encryption, role-based access control, and multi-layer security mechanisms. This integrated approach ensures improved security, reduced latency, and efficient file management compared to existing systems.

## III. SYSTEM ARCHITECTURE

The proposed system is designed to provide secure and efficient file management using fog computing and encryption techniques. The architecture focuses on performing data processing locally to reduce latency and improve security. The system consists of multiple components that work together to ensure controlled access, secure storage, and efficient file handling.

### A. User Authentication

The system begins with a secure authentication process where users are required to log in using valid credentials. The entered username and password are verified by the system to ensure that only authorized users can access the application. After successful login, the system identifies the user role such as administrator, company, cloud, or auditor. Based on the assigned role, specific permissions are granted to restrict unauthorized access and operations.

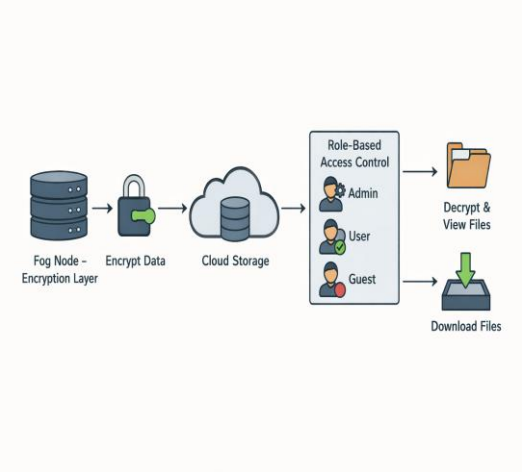
### B. File Encryption

Once the user is authenticated, the system allows file upload. Before storing the file, it is encrypted using the Advanced Encryption Standard (AES) algorithm. This converts the file into an unreadable format, ensuring data confidentiality. Even if

unauthorized users gain access to stored files, they cannot understand the content without proper decryption.

**C. File Storage and Management**

The encrypted files are stored in a structured format within the system. The storage mechanism ensures that files are organized and easily retrievable. The system maintains file integrity and prevents any unauthorized modifications.




**D. File Access and Decryption**

Users can view the list of stored files, but only file names are displayed while the content remains encrypted. When a user selects a file for download, the system verifies the user's permission and prompts for a decryption password. Only after entering the correct password, the system decrypts the file and displays it in its original format.

**E. Role-Based Access Control**

The system implements role-based access control to manage user permissions effectively. Different roles are assigned different levels of access, ensuring that users can perform only the operations permitted for their role. This enhances system security and prevents misuse of data.

**F. Audit Logging**

The system maintains a log of all user activities such as login attempts, file uploads, file access, and downloads. These logs include timestamps and user details, which help in monitoring system usage and detecting suspicious activities. This feature ensures accountability and transparency within the system.

Overall, the system architecture integrates fog computing, encryption, and access control mechanisms to provide a secure, efficient, and reliable file management solution.

## IV. IMPLEMENTATION AND DATASETS

The proposed system is implemented using Python and its associated libraries to provide a secure and efficient file management solution. The implementation integrates encryption techniques, user authentication, and role-based access control within a fog computing environment.

**A. Development Environment**

The system is developed using Python as the primary programming language due to its simplicity and flexibility. The implementation is carried out on a 64-bit operating system using development tools such as IDLE or PyCharm. Libraries such as Tkinter are used for designing the graphical user interface, while PyCryptodome is used for implementing AES-based encryption and decryption.

**B. User Interface Implementation**

A simple and user-friendly graphical interface is designed using Tkinter. The interface includes modules for login, file upload, file viewing, and file download. The design ensures ease of use, allowing users to interact with the system without requiring advanced technical knowledge.

**C. Encryption and Decryption Implementation**

The system uses the Advanced Encryption Standard (AES) algorithm to secure data. During file upload, the system encrypts the file before storing it. During file download, the system prompts the user to enter a decryption password. Only after successful verification, the file is decrypted and displayed in its original format.

**D. Dataset Description**

The system uses user-provided files as the dataset. These files may include documents, images, or text files that are uploaded by users into the system. The dataset is not predefined, as the system is designed to handle real-time file inputs dynamically. This approach ensures flexibility and applicability in real-world scenarios.

**E. Data Handling and Storage**

Encrypted files are stored in a structured format within the system, such as JSON or encoded files. The system ensures that all stored data remains in encrypted form, preventing unauthorized access. File handling operations such as upload, storage, retrieval, and display are performed efficiently.

**F. System Workflow Execution**

The system workflow includes user login, file upload, encryption, storage, file viewing, download request, password verification, and decryption. Each step is executed sequentially to ensure secure and controlled file management.

Overall, the implementation ensures that the system is secure, efficient, and user-friendly. The integration of encryption, access control, and fog computing provides a reliable solution for secure file management in real-world applications.

## V. RESULTS AND DISCUSSION

The proposed fog computing based secure file management system is evaluated based on functionality, security, and performance. Unlike prediction-based systems, this system is analysed through feature comparison and system behaviour. The results demonstrate that the system provides improved data protection, reduced latency, and efficient file management.

**A. Functional Results**

The system successfully performs all core operations including user authentication, file upload, encryption, file storage, file viewing, and file download. The login module validates user credentials correctly, preventing unauthorized access. The encryption module ensures that all uploaded files are converted into unreadable format, while the decryption process restores the original data only when the correct password is provided.

**B. System Comparison**

Table I shows the comparison between the existing cloud-based system and the proposed fog computing system. It is observed that the proposed system provides higher security, lower latency, and better access control due to local processing and encryption mechanisms.

**TABLE I** – *Comparison Between Existing and Proposed System*

| System | Security | Latency | Access Control |
|---|---|---|---|
| Cloud System | Medium | High | Limited |
| Proposed System | High | low | strong |

**C. Interface Results**

The system interface is tested to ensure usability and functionality. Fig. 1 shows the login interface of the system, where users enter valid credentials. Fig. 2 illustrates the file upload and encryption process. Fig. 3 represents the file viewing and decryption functionality. These results confirm that the system is user-friendly and operates efficiently.

The login interface allows users to enter valid credentials. The system verifies the username and password to ensure secure access and restrict unauthorized users.


**Fig. 1.** Login Interface of the System

The file upload module allows users to select and upload files into the system. The uploaded files are encrypted using AES before storage to ensure data security.

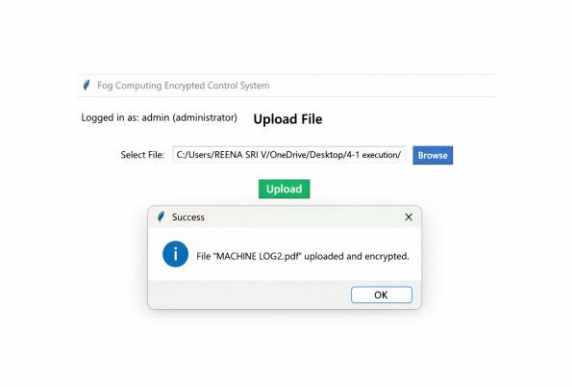



**Fig. 2.** File Upload and Encryption Process

The view files module displays the list of encrypted files. Users can select files for download, and decryption is performed only after verifying the password.

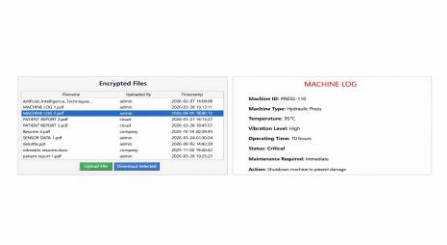



**Fig. 3.** File Viewing and Decryption

### D. Security Analysis

The system ensures strong data security using AES encryption and multi-layer protection. Even if unauthorized users access stored files, the data remains protected due to encryption. The additional decryption password further enhances security.

### E. Discussion

The results clearly indicate that the proposed system overcomes the limitations of traditional cloud-based systems. By using fog computing, the system reduces latency and improves performance. The integration of encryption, role-based access control, and audit logging enhances overall system reliability and security. Therefore, the proposed system provides a more efficient and secure solution for file management compared to existing approaches.

## VI. CONCLUSION

In this paper, a fog computing based secure file management system using AES encryption is proposed and implemented. The system addresses major challenges such as data security, unauthorized access, and latency associated with traditional cloud-based systems. By performing data processing locally, the proposed system reduces delay and improves overall performance. The implementation of the Advanced Encryption Standard (AES) ensures that all files are securely stored in encrypted format, protecting sensitive information from unauthorized users. The integration of role-based access control restricts user permissions and enhances system security. Additionally, the multi-layer security mechanism, which includes user authentication and decryption password verification, provides an extra level of protection. The system also incorporates an audit logging feature that records user activities, improving transparency and accountability. The results demonstrate that the system successfully performs all required operations such as login, file upload, encryption, storage, and secure file access.

Compared to existing systems, the proposed approach offers improved security, reduced latency, and better control over data. Overall, the proposed system provides a reliable, efficient, and secure solution for file management. It can be effectively used in real-world applications where data protection and controlled access are essential. Future enhancements may include cloud integration, advanced authentication methods, and improved scalability to support larger systems.

**REFERENCES**

[1] W. Stallings, "Cryptography and Network Security: Principles and Practice," 7th ed., Pearson, 2017.

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

[3] D. Boneh and V. Shoup, "A Graduate Course in Applied Cryptography," Cambridge University Press, 2020.

[4] NIST, "Advanced Encryption Standard (AES)," FIPS PUB 197, 2001.

[5] M. Chiang and T. Zhang, "Fog and IoT: An Overview of Research Opportunities," IEEE Internet of Things Journal, vol. 3, no. 6, pp. 854–864, 2016.

[6] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and Its Role in the Internet of Things," in Proc. MCC Workshop on Mobile Cloud Computing, 2012, pp. 13–16.

[7] S. Yi, Z. Qin, and Q. Li, "Security and Privacy Issues of Fog Computing: A Survey," Wireless Personal Communications, vol. 95, no. 1, pp. 1–26, 2017.

[8] R. Roman, J. Lopez, and M. Mambo, "Mobile Edge Computing, Fog Computing and Cloud Computing: A Survey," Future Generation Computer Systems, vol. 78, pp. 680–698, 2018.

[9] A. Singh and S. Sharma, "Secure Data Storage Using AES Encryption," International Journal of Computer Applications, vol. 179, no. 18, pp. 1–5, 2018.

[10] X. Sun, P. Zhang, and Y. Chen, "Secure Data Storage in Cloud Computing Using Encryption Techniques," Journal of Cloud Computing, vol. 7, no. 1, pp. 1–12, 2018.

[11] K. Bilal, O. Khalid, A. Erbad, and S. U. Khan, "Potentials, Trends, and Prospects in Edge Technologies: Fog, Cloudlet, Mobile Edge, and Micro Data Centers," Journal of Computing, vol. 15, no. 3, pp. 1–15, 2018.

[12] H. Gupta, A. Vahid Dastjerdi, S. K. Ghosh, and R. Buyya, "iFogSim: A Toolkit for Modelling and Simulation of Resource Management Techniques in Fog Computing," Software: Practice and Experience, vol. 47, no. 9, pp. 1275–1296, 2017.

[13] T. H. Luan, L. Gao, Z. Li, Y. Xiang, and L. Sun, "Fog Computing: Focusing on Mobile Users at the Edge," arXiv preprint arXiv:1502.01815, 2015.

[14] S. Sarkar, S. Chatterjee, and S. Misra, "Assessment of the Suitability of Fog Computing in the Context of Internet of Things," IEEE Transactions on Cloud Computing, vol. 6, no. 1, pp. 46–59, 2018.

[15] Q. Zhang, M. Chen, L. T. Yang, Z. Chen, and M. A. Razzaque, "Recent Advances in Edge Computing for Smart Cities: A Survey," IEEE Access, vol. 6, pp. 67563–67575, 2018.