International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4639 - 4644
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0609

SmartGaurd: Real-Time Behavioral Analysis System for Malicious File Activity Detection

Prachi G Jadhav[1], Sai v Jadhav[2], Neha R Devadiga[3], Vaishnavi S Gaikwad[4], Prof. Manoj Rathod[5]
[1,3,4]UG Scholar, Dept. of Computer Science& Engg. & Satara, Maharashtra, India
[2]UG Scholar, Dept. of Electronics & Telecommunication. ,Sangli, Maharashtra, India
[5]Associate professor, Dept. of Computer Science& Engg. & Satara, Maharashtra, India
Emails: *prachiganeshjadhav96@gmail.com[1], vjsai1143@gmail.com[2], nehardevadiga04@gmail.com[3], vaishnavigaikwad9091@gmail.com[4], manoj.rathod.@kbpcoes.edu.in[5]*

Abstract
The rapid growth of digital documentation in enterprises has created significant challenges in ensuring secure storage, controlled access, and reliable auditing of sensitive information. This research presents a SmartGaurd: Real-Time Behavioral Analysis System for Malicious File Activity Detection designed to protect confidential organizational data using a layered security architecture. The proposed system is developed using a React-based user interface for responsive and dynamic client interaction, while the server-side operations are handled through the Django framework, enabling robust authentication, authorization, and document processing. The platform integrates SQLite3 as the database engine to maintain structured document metadata, user credentials, and audit logs. To strengthen access control, the system incorporates One-Time Password (OTP) verification mechanisms and encryption techniques to ensure that only authorized users can access or download protected files. This architecture improves data confidentiality while minimizing the risk of unauthorized data exposure. The system workflow includes secure user authentication, encrypted document upload, malware inspection, controlled file sharing, and real-time audit logging to maintain transparency in document access activities. Whenever a user attempts to retrieve protected documents, the system dynamically generates an OTP to validate the request and prevent unauthorized downloads. Additionally, administrative and auditing modules monitor user behavior and maintain detailed activity logs that can be analyzed for compliance verification and anomaly detection. By combining modern web technologies (React and Django) with security mechanisms such as encryption and OTP-based validation, the proposed system provides a scalable and efficient approach to enterprise document security. The implementation demonstrates how integrated authentication, encryption, and auditing mechanisms can significantly enhance data protection while maintaining system usability and operational efficiency.

Keywords: *SmartGaurd, One-Time Password(OTP), user authentication, audit logging, activity logs, encryption.*

1. Introduction
The rapid growth of digital technologies has increased the volume of electronic documents generated and shared within organizations. As businesses increasingly rely on digital platforms managing these documents management system often lack strong security measures, making the vulnerable to unauthorized access, data leakage, and malicious activities. Traditional document management systems often provides limited security controls. Consequently, organizations require secure and intelligent document management solutions that not only facilitate efficient document handling but

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4639 - 4644
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0609

also ensure strong protection of sensitive information. To address these issues, the paper focuses on secure document management and auditing system. Th system is developed using React for the frontend, Django for backend services, and SQLite3 for database management. It incorporates security features such as encryption and One-Time Password (OTP) verification to protect sensitive documents and ensure authorized access. Documents uploaded to the platform are securely stored, and access requests are validated through user authentication and OTP verification before document retrieval or sharing is permitted. These measures significantly enhance data confidentiality and reduce risk of information leakage. The platform supports secure document upload, encrypted storage, controlled sharing, and comprehensive audit logging. Whenever a user requests a document, the system verifies the user's identity and generates an OTP for access validation. All activities are recorded in audit logs for monitoring, compliance, and security analysis. By combining modern web technologies with advanced security mechanism, the proposed system provides a scalable, reliable, and secure solution for enterprise document management

2. Literature Survey

With the rapid growth of digital systems, the threat of application-based attacks and malware has significantly increased. Researchers have proposed various techniques such as behavioral modeling, intrusion prevention frameworks, and memory forensics to detect and prevent such attacks effectively. K. Nishitha and P. Usha (2024) suggested employing behavioral modeling techniques to create a real-time detection model for application-based threats. Their method effectively detects fraudulent activity by continuously monitoring application activities and analyzing runtime behavior patterns. By concentrating on aberrant behavioral traits rather than only employing conventional signature-based techniques, the system was able to accomplish better assault detection.[1] Similarly, D. Kushwahaa et al. (2022) introduced a lateral movement detection framework based on user behavioral analysis. Their method analyzes user activities, login patterns, and system interactions to detect suspicious movements within a network environment. The framework improved the identification of insider threats and unauthorized access attempts through behavioral monitoring. However, the system may generate false positives when legitimate users exhibit unusual but non-malicious behavior.[2] In other study, H. Aljihani et al. (2021) presented a standalone behavior based attack detection technique for distributed software systems integrated with blockchain technology. Their approach utilized blockchain mechanisms to ensure secure and tamper-resistant detection of malicious activities in distributed environments. [3] R. Han et al. (2023) proposed a machine learning-based approach for detecting malicious behavior using host process data. Their system analyzed process execution patterns, resource utilization, and system calls to identify abnormal activities associated with malware. The proposed model achieved high detection accuracy through machine learning classification techniques. However, the approach required extensive training datasets and may struggle with zero-day attacks not represented in the training data.[4] Additionally, Z. Maasaoui et al. (2025) developed a scalable framework for realtime network security traffic analysis and attack detection using machine learning and deep learning methods. Their framework processed large-scale network traffic efficiently and detected various cyberattacks in real time. The use of deep learning improved detection performance for complex attack patterns. However, the framework demanded significant computational resources and high-quality labeled datasets for effective training. [5] Role-based access control and password-based authentication are implemented by a number of current document management systems. These techniques offer a minimal degree of protection, but they are still susceptible to insider threats, unwanted access, and credential theft. Moreover, a lot of systems don't have ongoing monitoring features that can spot questionable user behavior after authentication. The importance of One-Time Password (OTP) verification and multi-factor authentication (MFA) in avoiding unwanted access has been emphasized by

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4639 - 4644
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0609

recent cybersecurity studies. By requiring temporary credentials during sensitive operations like document downloads and file sharing, OTP-based systems add an extra layer of security. OTP verification dramatically lowers the risk of account compromise, according to studies. As a result, a comprehensive system that can secure papers and keep tabs on user behavior in real time is still required. By incorporating numerous security levels into a single document management system, SmartGaurd closes this research void. To provide improved protection from both external and internal threats, the suggested system integrates OTP-based authentication, encrypted document storage, malware detection, behavioral monitoring, and audit logging.

3. Proposed Methodology

3.1.System Overview

Through behavioral monitoring, SmartGaurd is a web-based intelligent document management solution[6] that secures critical organizational documents and identifies malicious file activity. The system uses a security architecture that has multiple layers and includes user authentication, OTP verification, encryption, malware detection, behavioral analysis, and audit logging. For the user interface, the application is built using ReactJS, for backend processing, it uses Django, and for database management, it employs SQLite3. These technologies provide scalability, maintainability, and effective handling of document-related tasks[7].

3.2.System Architecture

3.2.1. User Authentication Module

User Credentials are authenticated using SHA-256 hashing to ensure secure password verification and prevent credential exposure.

3.2.2. OTP Verification Module

When a user tries to download or view a sensitive document, the system generates an OTP. Only upon successful OTP validation is access permitted, adding an extra degree of protection against illicit actions.

3.2.3. Encryption Module

Document are protected using AES encryption before storage. Homomorphic encryption is incorporated to support to privacy-preserving processing on encrypted data without exposing the original content.

3.2.4. Malware Detection Module

Before saving uploaded files, the system inspects them for malware using ClamAv and YARA rules. To avoid potential security problems, suspicious files are flagged and isolated.

3.2.5. Behavioral Analysis Module

This module continuously keeps track of user activities like login tries, file uploads, downloads, sharing actions, and access frequency. Administrators get a report on any unusual behavior discovered so that they may investigate it further.

3.2.6. Audit Logging Module

All system activities are recorded in detailed audit logs. These logs support compliance monitoring, forensic analysis, and security auditing[8].

3.3.Workflow of the Proposed System

- Step 1: User registration and authentication
- Step 2: User Uploads documents to the platform
- Step 3: Uploaded files undergo malware inspection.
- Step 4: Documents are encrypted and stores securely.
- Step 5: User requests document access
- Step 6: OTP Verification is performed
- Step 7: Behavioral analysis evaluates user activity.
- Step 8: Access is granted or denied.
- Step 9: Activity details are recorded in audit logs[9]

3.4.Architecture Diagram

The SmartGaurd architecture consists of Employee, Admin, Auditor interfaces connected to an authentication and authorization module that enforces secure access through OAuth 2.0 and RBAC. User requests are processed by backend services, which coordinate file management. Encrypted files are[10] stored in secure object storage, while user information and audit logs are maintained in a database. This architecture ensures secure document storage, controlled access, malware detection, and comprehensive activity monitoring. As shown in Figure 1 Architecture Diagram of Smartgaurd., Table 1 Behavioral Anomaly Detection Algorithm, Table 2 Experimental Results of SmartGaurd System[11].

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4639 - 4644
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0609

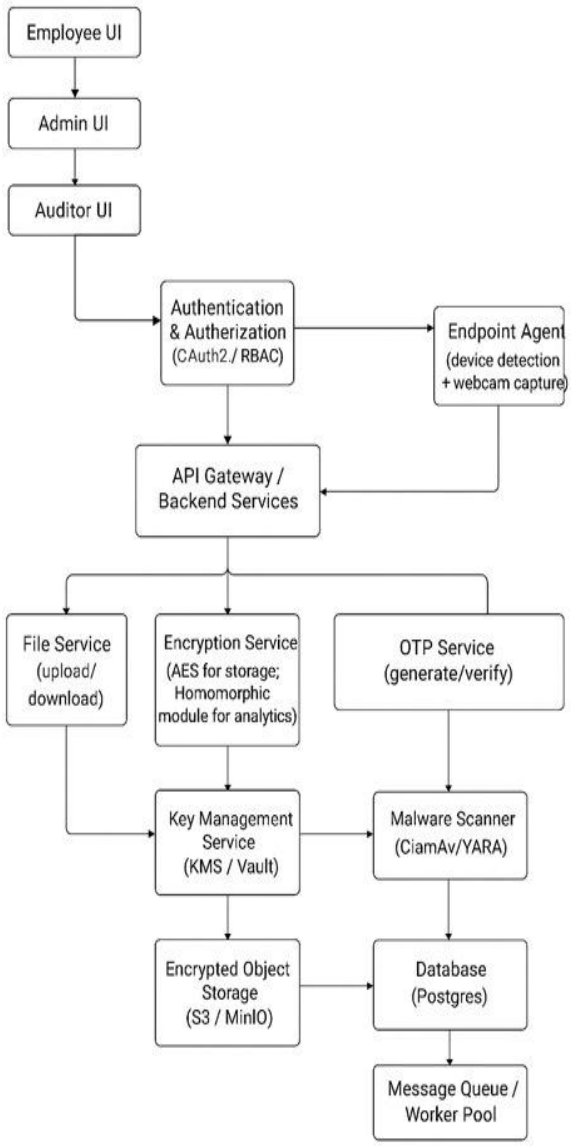

Figure 1 Architecture Diagram of Smartgaurd

3.4.1. Algorithm

Algorithm: SmartGaurd Security workflow
Input: User request and document file
Output: secure access decision[12]

- Start
- Receive user credentials.
- Apply SHA-256 hashing for password verification.
- Authenticate the user
- Verify user authorization level.
- Scan uploaded files using ClamAV and YARA rules.
- If malware is detected:
  - Block file upload, user credential
  - Generate security alert.
  - Record event in audit logs
  - Stop
- Encrypt document Using AES encryption.
- Apply homomorphic encryption for secure processing.
- Store encrypted document securely
- Generate OTP for document access request.
- Validate OTP entered by the user.
- If OTP validation fails:
  - Deny access
  - Record event in audit logs
  - Stop
- Monitor behavioral activity patterns
- If abnormal behavior is detected:
  - Generate alert.
  - Capture evidence
  - Restrict access
- Else:
- Grant secure document access
- Update audit logs
- End

Table 1 Behavioral Anomaly Detection Algorithm

Algorithms	Purpose
AES(Advanced Encryption Standard)	Encrypts sensitive documents before storage and sharing
Homomorphic Encryption	Enables secure processing of encrypted data without exposing the original contents
SHA-256 Hashing	Secures user password and verifies data integrity

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4639 - 4644
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0609

OTP Verification	Provides multi-factor authentication t access

3.4.2. YARA Pattern Matching Files Scanned using signature rules:

```
yara.match(file_path)
```

3.4.3. Encryption Workflow

```
Encrypted_File = AES_Encrypt(Input_File)
Store(Encrypted_File)
IF user_requests_download:
Validate_OTP()
Decrypted_File = AES_Decrypt(Encrypted_File)
Return_File()
```

This ensures complete confidentiality and controlled access.

4. Results and Discussion

4.1.Results

The SmartGaurd system was successfully implemented using ReactJS, Django, SQLite3. Functional testing demonstrated successful extension of all core security modules. The authentication modules accurately validated user credentials and prevented unauthorized login attempts. OTP verification successfully introduced a second layer of authentication for document access. The malware detection module effectively identified suspicious files during upload operations. Encrypted document storage ensured that sensitive files remained protected even in the event of storage compromise. Behavioral monitoring successfully recorded user actions and generated alerts for unusual access patterns. Audit logs maintained complete records of user interactions, enabling efficient compliance monitoring. Performance evaluation demonstrated that the system maintained acceptable response times while executing multiple security operations simultaneously.

Table 2 Experimental Results of SmartGaurd System

Parameter	Result
Total Files Tested	100
Malware Files Detected	98
Detection Accuracy	98%
Average Upload Time	1.8 sec
Average Download Time	1.2 sec
OTP Verification Success rate	100%
Audit Log Accuracy	100%

5. Discussion

The implementation results demonstrate that integrating multiple security mechanisms significantly enhances document protection. OTP-based verification reduced the risk of unauthorized access, while encryption ensured confidentiality of stored documents. Behavioral analysis provided additional protection against insider threats by unusual activities that conventional authentication mechanism may not detect. The audit logging framework improved[13] accountability and enabled detailed investigation of security incidents. Compared with traditional document management system, SmartGaurd offers a comprehensive security framework that combines prevention, detection, and monitoring capabilities within a single platform. The modular architecture facilitates future scalability and integration with advanced machine learning models for predictive threat detection[14] SDFF.

Conclusion

This paper presented SmartGaurd: A Real-Time Behavioral Analysis System for Malicious File Activity Detection. The proposes system integrates user authentication, OTP verification, encryption, malware detection, behavioral monitoring, and audit logging to provide comprehensive protection for sensitive organizational documents. This implementation demonstrated the effectiveness of combining multiple security layers within a unified document management platform. Experimental results indicates that SmartGaurd improves confidentiality, access control, and accountability

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 04 Issue: 07 July 2026
Page No: 4639 - 4644
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2026.0609

while maintaining usability and operational efficiency. Future work will focus on integrating machine learning-based anomaly detection models, cloud-based storage services, and advanced threat intelligence mechanisms to future enhance security performance and scalability.

References

[1]. K. Nishitha and P. Usha, "Real-Time Detection Model for Application-Based Attacks Using Behavioral Modeling Techniques," 2024.
[2]. D. Kushwahaa et al., "Lateral Movement Detection Framework Using User Behavioral Analysis," 2022.
[3]. H. Aljihani et al., "Behavior-Based Attack Detection Technique for Distributed Software Systems via Blockchain," 2021.
[4]. R. Han et al., "Machine Learning-Based Detection of Malicious Behavior Using Host Process Data," 2023.
[5]. Z. Maasaoui et al., "Real-Time Network Security Traffic Analysis and Attack Detection Using Machine and Deep Learning," 2025.
[6]. A. K. Tripathi et al., "Survey on Data Platforms: Concepts, Challenges and Opportunities," 2022.
[7]. J. Tian et al., "Systematic Review of Attack Detection Techniques Used in Cybersecurity Systems," 2014.
[8]. A. Velliangiri et al., "Homomorphic Encryption for Secure Computation in Cloud Computing Environments," 2024.
[9]. J. Singh et al., "Scalable Hybrid Homomorphic Encryption Approach for Cloud Data Privacy," 2024.
[10]. S. Venkatesh and R. Mukesh, "Energy-Aware Multimodal Homomorphic Security Mechanism for Private Cloud Systems," 2023.
[11]. J. Ramasamy and M. Dinesh, "Hybrid Encryption Approach for Advanced Cloud Data Security," 2023.
[12]. M. A. Supriya, "Multi-Layer Evaluation Framework for Encrypted Data Processing Using Fully Homomorphic Re-encryption," 2024.
[13]. T. Al Attar and M. A. Mohammed, "Fully Homomorphic Encryption Scheme for Securing Cloud Data," 2023.
[14]. P. Manazipet et al., "Data Security Enhancement in Cloud Computing Using AES Encryption," 2023.