An Efficient Data Protection Model for Digital Criminal Record Management

E. O. Bennett
Department of Computer Science,
Rivers State University, Port Harcourt, Nigeria.
Bennett.okoni@ust.edu.ng

U. E. Chinenye
Department of Computer Science,
Rivers State University, Port Harcourt, Nigeria.

DOI: 10.56201/ijcsmt.v11.no2.2025.pg40.58

Abstract

The effective management and protection of digital criminal records is a crucial concern for law enforcement agencies, legal authorities, and government organizations. Traditional paper-based record-keeping systems are increasingly being replaced by digital platforms, which offer enhanced accessibility, efficiency, and scalability. However, the transition to digital criminal record management also introduces new challenges in terms of data security, privacy, and compliance with regulatory frameworks. This dissertation presents an efficient data protection model for the management of digital criminal records. The model leverages a combination of advanced encryption techniques, access control mechanisms, and robust data backup and recovery strategies to ensure the confidentiality, integrity, and availability of sensitive criminal justice information. The proposed model is designed to be scalable in handling increased data loads and user traffic, ensuring that the system remains responsive and efficient even as more crime data is added. Its optimization performance reduced the CPU usage from 70% to 50%, Memory usage from 80% to 60%. The encryption time witnessed a significant reduction of 0.08sec for 1000KB file size as against 0.12 sec in the existing system. It is also adaptable and compliant with relevant data protection regulations, such as the General Data Protection Regulation (GDPR) and national-level data privacy laws. The implementation of this model has significantly enhanced the overall security and integrity of digital criminal record management systems, while ensuring the protection of sensitive personal and criminal justice information.

***Keywords* – Crime management, access control, data protection, criminal record**

1. Introduction

In today's digital age, the need to securely manage criminal records has become increasingly important. Digitizing criminal records allows for efficient storage and retrieval of sensitive information. However, this also raises concerns about data privacy and security. To address these challenges, researchers have explored the use of efficient data protection model technology as a potential solution for criminal record management [1], [2].

Data protection model offers several advantages for criminal record management, through CIA triad which represent Confidentiality, Integrity and Availability. To ensure the safeguarding of criminal record data protection management practices such as Data Loss Prevention (DLP), Encryption, Access control, Data backup and Data resiliency can be securely and efficiently managed, reducing the risk of unauthorized access or tampering [3], [4].

While strong privacy protection is essential for criminal record management, it is important to balance this with the need for effective rehabilitation and reintegration of individuals with a criminal history. Overly restrictive policies regarding the dissemination of criminal record information can hinder the successful reintegration of these individuals into society [5]. Therefore, a balanced approach is necessary to ensure the protection of individual privacy while also supporting the rehabilitation and reintegration process.

Furthermore, the use of digital technologies in criminal investigations and prosecutions raises concerns about the potential for abuse and the need for robust safeguards. Researchers have emphasized the importance of designing specific procedural safeguards, such as defense access to training data and algorithms, to ensure the legitimacy and fairness of digital criminal investigations [6]. Additionally, the integration of criminal justice agencies through centralized electronic systems can improve transparency and accountability in the criminal justice system [7].

2. Literature Review

[8] Evaluated the concept of a data protection model in the field of information security some of the key findings also made from the literature is Consent Management: Many data protection models emphasize the importance of obtaining informed consent from individuals for the collection and use of their personal data. Research has highlighted challenges in obtaining meaningful consent and recommended approaches such as using layered notice and consent mechanisms, providing granular choices, and implementing user-friendly consent interfaces.

[9] Discussed Digital Record Management System (DRMS) as a software solution designed to store, manage, and track electronic documents and records in a centralized and organized manner. The systems are used by organizations to efficiently manage their digital records ensure compliance with regulations and improve overall productivity. It further discussed the numerous features and benefits of DRMS in improving efficiency in record retrieval and reduced administrative overhead are commonly reported advantages. Its studies have also emphasized the potential for increased security and compliance with data protection regulations, as well as enhanced collaboration and information sharing within organizations. Efficient metadata management has proven to be very crucial in success of a digital record management system. It provides descriptive information about records, facilitating their identification, retrieval, and context understanding.

[10] Explored data encryption as security method that translates data into a code, or cipher text that can only be read by people with access to a secret key or password. He evaluated how data encryption protects data from being stolen, changed, or compromised.

[11] Evaluated access control as a mechanism that is used to maintain data confidentiality by inspecting the users' rights against a set of authorizations. A security administrator that is responsible for managing a DBS specifies these authorizations. The study compared many types of access control policies, the three traditional access control policies DAC, MAC, and RBAC policy and the modern age, ABAC is a promising alternative to traditional policies of access control (DAC, MAC, and RBAC) which attracts the attention in both recent academic literature and industry application.

[12] "A Review of Cloud Service Security with Various Access Control Methods," discusses the significance of data security in cloud computing, focusing on access control mechanisms that ensure only authorized users can access sensitive data. The authors review different models, including Discretionary Access Control (DAC), Mandatory Access Control (MAC), Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), and Risk-Based Access Control. The paper emphasizes the importance of protecting outsourced data in a multi-tenant cloud environment and compares the effectiveness of these models.

[13] Employs a proxy re-encryption strategy combined with identity-based encryption for securely exchanging data in cloud contexts. In this framework, data owners encrypt their data before sending it to the cloud, and legitimate users can access the data through a proxy re-encryption method. The use of an edge device as a proxy server allows for performing complex calculations, facilitating the re-encryption process. Additionally, the system leverages information-centric networking capabilities to cache data in the proxy, enhancing service quality and network capacity. It also integrates blockchain technology for decentralized data sharing, improving efficiency and enabling fine-grained data access control.

3. Methodology

The architecture diagram in Figure 1 depicts the structural components and interactions within the proposed system for digital criminal record management. At its core is a secure and scalable application layer, housing the system's business logic and functionality. User interactions are facilitated through a user interface layer, providing an intuitive platform for users to register/login, input crime records, and access encrypted data. The system relies on robust authentication mechanisms to ensure secure user access. Data encryption and data loss prevention components are integrated into the architecture to safeguard sensitive information, while an audit logging system tracks user activities for security and compliance purposes. Additionally, external interfaces enable integration with law enforcement databases and other external systems for data exchange and interoperability. Overall, the architecture ensures the confidentiality, integrity, and availability of criminal records data while facilitating seamless user interactions and compliance with regulatory requirements.

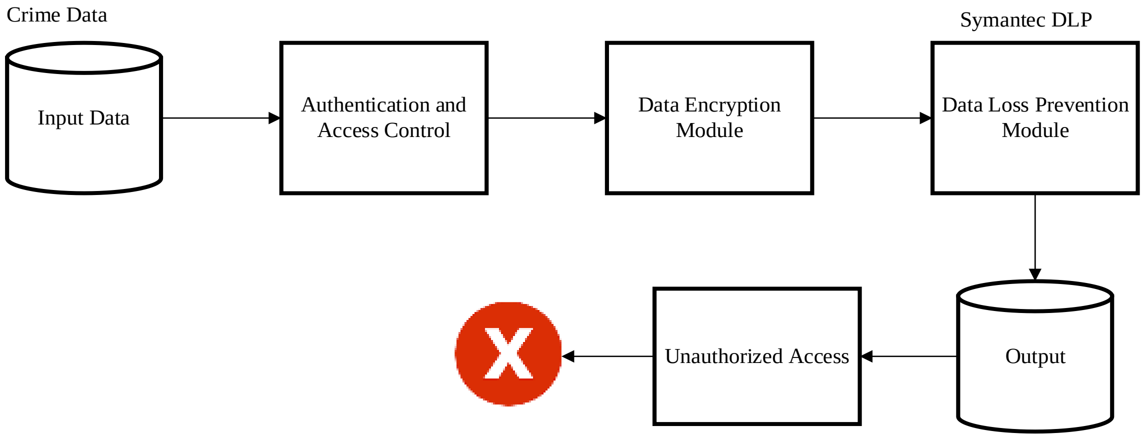


Figure 1: Architectural design

Input Data: The input data for the efficient data protection model in digital criminal record management comprises a comprehensive array of information sources. It includes detailed criminal records encompassing arrest history, charges, court proceedings, convictions, and sentencing details, alongside sensitive personal identifiable information (PII) like names, addresses, social security numbers, and dates of birth. Additionally, the dataset incorporates case information such as case numbers, offense types, and jurisdictional details, as well as law enforcement reports, court documents, victim statements, suspect details, and forensic evidence data.

Authentication and Access Control: User Authentication: Implement mechanisms such as username/password, multi-factor authentication (MFA), or biometric authentication to verify the identity of users accessing the system. Role-Based Access Control (RBAC): Define roles and permissions based on user responsibilities and grant access privileges accordingly. This ensures that users can only access data relevant to their role.

Data Encryption Module: Encrypt sensitive data at rest and in transit using strong encryption algorithms to prevent unauthorized access or tampering. Implement robust key management practices to securely generate, store, and rotate encryption keys, ensuring the confidentiality of encrypted data.

Data Loss Prevention Module: DLP Policies: Define and enforce data loss prevention policies to prevent unauthorized transmission or sharing of sensitive data outside the authorized boundaries. Implement content inspection mechanisms to scan outgoing communications, files, or data transfers for sensitive information and enforce policy-based controls to prevent data leaks.

Output: The output of the efficient data protection model in digital criminal record management includes authorized data, which refers to the subset of information accessible to authorized users based on their roles and permissions. This authorized data may include sanitized criminal records with sensitive PII masked or anonymized, case information relevant to ongoing investigations or legal proceedings, and aggregated statistics for analytical purposes. Access to authorized data is governed by strict authentication and access control mechanisms, ensuring that only authorized personnel with legitimate needs can retrieve, view, or manipulate the data. By delineating and safeguarding authorized data, the model aims to prevent unauthorized access, minimize privacy risks, and uphold confidentiality while facilitating legitimate use cases within the criminal justice system.

3.3.2 Component Design of Data Encryption

The component design elaborates the subcomponents of the encryption module in Figure 2. It shows how various components interact with each other in forming a secured means of protecting email contents.

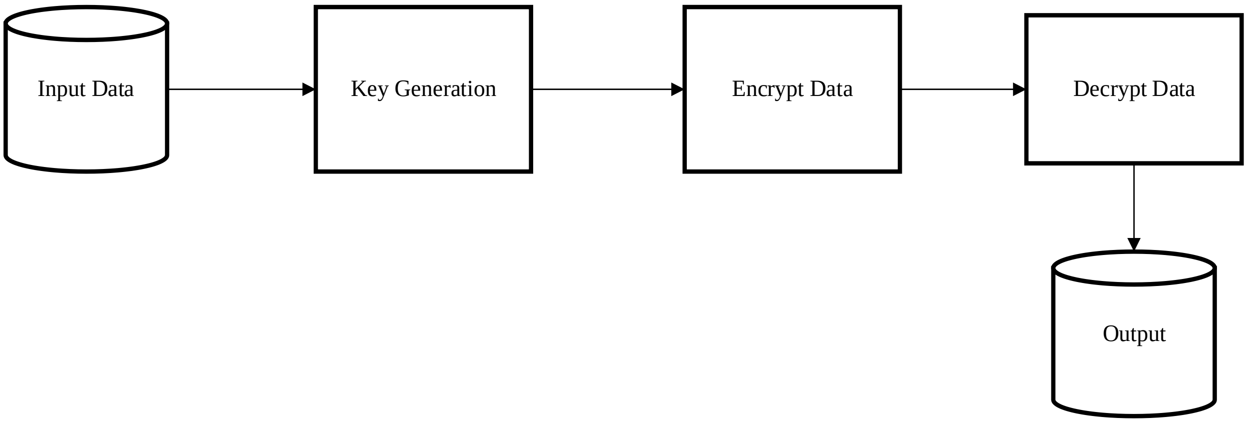


Figure 2: Component Design of Symmetric Encryption Model

Input Data: This component involves the gathering of raw data, which in the context of digital criminal record management, includes information such as arrest records, charges filed, court proceedings, convictions, sentencing details, and other pertinent details related to criminal activities. This data is typically collected from various sources such as law enforcement agencies, courts, correctional facilities, and other relevant entities.

Key Generation: Key generation is the process of creating a cryptographic key that will be used for encrypting and decrypting the data. In this step, a random or pseudo-random key is generated using cryptographic algorithms and techniques. The strength and randomness of the key are crucial factors in ensuring the security of the encryption process. Depending on the encryption algorithm used, the key may have specific requirements such as length, complexity, and entropy.

Encrypt Data: Once the cryptographic key is generated, the next step is to encrypt the data using symmetric encryption algorithms. Symmetric encryption involves using the same key for both encryption and decryption. In the context of digital criminal record management, sensitive data such as personal identifiable information (PII) and case details are encrypted to protect confidentiality and prevent unauthorized access. The encryption process transforms the plaintext data into ciphertext, rendering it unreadable without the corresponding decryption key.

Decrypt Data: Decryption is the reverse process of encryption, where the ciphertext data is converted back into its original plaintext form using the decryption key. This step is essential for authorized users to access and retrieve the encrypted data securely. Using the symmetric key, proper key management practices are crucial to ensure that authorized users have access to the decryption keys while maintaining the security of the system.

Output: The output component involves presenting the decrypted data in a readable format to authorized users or applications. Once the data is successfully decrypted, it can be accessed, analyzed, and utilized for various purposes such as criminal investigations, legal proceedings, statistical analysis, and reporting. The output may be presented through a user interface, reports, APIs, or other means depending on the requirements of the system and its users.

Algorithm 1 outlines the process for implementing robust access control mechanisms within a system. It begins by defining user roles and permissions, organizing them into a structured framework. Access control lists (ACLs) or role-based access control (RBAC) mechanisms are then implemented to regulate data access based on these predefined roles and permissions. Users are authenticated and authorized based on their assigned roles, ensuring that they can only access data relevant to their responsibilities.

Algorithm 1: Implementing Robust Access Control Mechanisms

Inputs:

- DD = Data flow within the system
- RR = User roles and permissions

Outputs:

- $D_{regulated}$ = Regulated data flow
- $D_{monitore}$= Monitored data access

Procedure:

1. Define user roles and permissions: $R = R_1, R_2, \dots, R_n$
2. Implement access control lists (ACLs) or role-based access control (RBAC) mechanisms.
3. Authenticate users and authorize access based on their assigned roles and permissions.
4. Monitor data flow and access logs to detect and prevent unauthorized activities.
5. Enforce encryption and authentication protocols to secure data transmission and access.

Additionally, the algorithm emphasizes the importance of monitoring data flow and access logs to detect and prevent unauthorized activities. This involves continuously monitoring user actions and system logs to identify any anomalies or suspicious behaviour. Finally, encryption and authentication protocols are enforced to secure data transmission and access, safeguarding sensitive information from unauthorized access or tampering.

Algorithm 2 focuses on ensuring the reliability and integrity of data through customized validation checks. It begins by identifying specific data validation requirements, which could include ensuring data accuracy, completeness, or compliance with regulatory standards. Customized validation rules and procedures are then developed to meet these requirements, tailored to the unique characteristics of the data being processed. These validation checks are implemented to verify the integrity and reliability of the data, ensuring that it remains accurate and consistent throughout its lifecycle. Regular audits and integrity checks are performed to detect any deviations from the established validation rules, allowing for prompt corrective actions to maintain data reliability.

Algorithm 2: Establishing Customized Data Reliability Checks

Inputs:

- $D = Data\ validation\ requirements$
- $S = Data\ integrity\ standards$

Outputs:

- $D_{validated} = Validated\ and\ reliable\ data$

Procedure:

1. Identify data validation requirements: $D = D_1, D_2, \ldots D_3$
2. Develop customized validation rules and procedures: $V(D_I)$
3. Implement validation checks: $V(D_I) \rightarrow D_{Validated}$
4. Perform regular audits and integrity checks to ensure data reliability and consistency.
5. Integrate error handling mechanisms to identify and rectify validation failures.

Error handling mechanisms are integrated to identify and rectify validation failures, minimizing the risk of data inconsistencies or inaccuracies.

Algorithm 3 addresses the challenge of enhancing system scalability to accommodate growing workloads and user demands. It begins by assessing the current system workload to understand its capacity limitations and performance bottlenecks. Scalability requirements are then identified, considering factors such as user growth, data volume, and transactional complexity. Scaling strategies are implemented to optimize system performance and resource utilization, ensuring that the system can efficiently handle increasing workloads without sacrificing performance or reliability.

Algorithm 3: Enhancing System Scalability

Inputs:

- WW = Current system workload
- RR = Scalability requirements

Outputs:

- $SASA$ = Scalable system architecture
- $PEPE$ = Efficient performance under heavy workloads

Procedure:

1. Assess current system workload: W
2. Identify scalability requirements: R
3. Implement scaling strategies: $S_A = f(W, R)$
4. Optimize resource utilization: $P_E = g(SA)$
5. Conduct load testing and performance tuning to validate scalability improvements.

This involve horizontal scaling, vertical scaling, or adopting cloud-based solutions to dynamically allocate resources based on demand. Load testing and performance tuning are conducted to validate the scalability improvements, ensuring that the system can seamlessly scale to meet evolving business needs while maintaining optimal performance under heavy workloads.

Algorithm 4A the encryption function E can be any suitable encryption algorithm, such as the Caesar cipher, substitution cipher, or more advanced encryption schemes like AES (Advanced Encryption Standard). Each encryption algorithm will have its own mathematical expressions defining the encryption process, but the general idea remains consistent across different algorithms.

Algorithm 4A: a step-by-step algorithm for a basic data encryption process

Input: Crime data

Output: Encrypted data

Step 1: Key Generation Generate a random encryption key K of sufficient length. This key will be used to encrypt and decrypt the data.

Step 2: Data Encryption For each character x_i in the plaintext data P, apply the encryption function E using the encryption key K:

$Ci=E\ (x_{i,}\ K)$

Where C_i represents the ciphertext corresponding to $x_{i.}$

Step 3: Output Combine all encrypted characters C_i to form the encrypted message C.

Algorithm 4B outlines the process for decrypting encrypted data using the encryption key KK and a decryption function DD. Each encrypted character $CiCi$is decrypted individually, resulting in the reconstruction of the original plaintext data PP.

Algorithm 4B: Basic Data Decryption Process

Input: Encrypted data CC, Encryption key KK

Output: Decrypted data PP

Step 1: Key Input

a. Input the encryption key KK used during the encryption process.

Step 2: Data Decryption

a. For each encrypted character $CiCi$ in the ciphertext CC, apply the decryption function DD using the encryption key KK: $xi=D(Ci,K)xi=D(Ci,K)$ Where $xixi$ represents the corresponding plaintext character.

Step 3: Output

a. Combine all decrypted characters $xixi$ to reconstruct the original plaintext data PP.

4. Results

4.1 System Requirements

4.1.1 Hardware Requirements

In order to implement the proposed system, the security agency must first have computer systems of about of 8GB ram minimum, 512 SSD and windows operating system (windows 10 or greater versions) installed on it. The minimum hardware requirements of the system include:

1. A microprocessor with minimum of 64 bits Clock speed of 20H2
2. 8GB of RAM
3. Compatible mouse, keyboards.
1. Hard disk of about 512 SSD.

4.1.2 Software Requirements

The software Requirement for the design of the proposed system involves

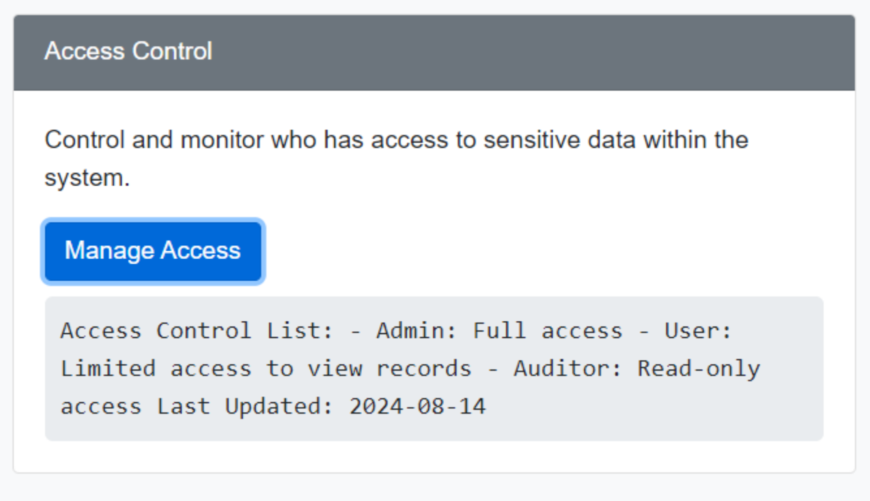
i. Microsoft Windows 8 and above.

ii. A web browser

iii. Anaconda (Python Distribution)

4.2 Implementing Role-Based Access Control (RBAC)

A critical component of the crime data protection system is the implementation of role-based access control (RBAC) to ensure that only authorized personnel can access sensitive data. The RBAC mechanism is integrated with the Flask backend and controls user access based on their assigned role. For instance, a "Detective" may have full access to add and view crime data, while an "Analyst" may only generate reports. This ensures that crime data is only accessible to individuals with the necessary clearance, minimizing the risk of unauthorized data exposure and maintaining the integrity of the system.



Access Control

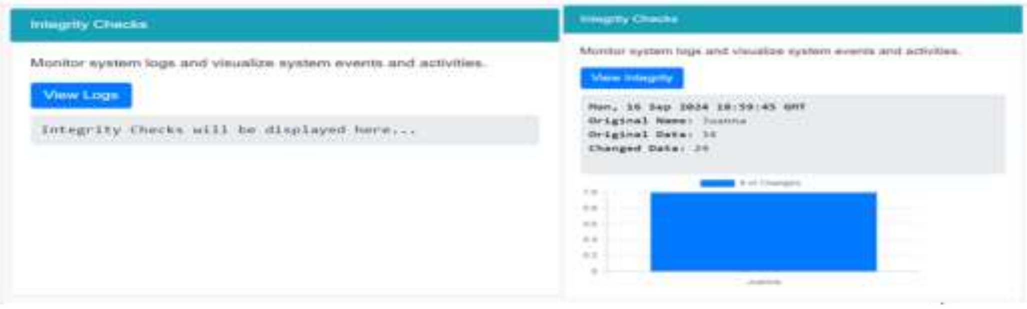
Control and monitor who has access to sensitive data within the system.

```
Access Control List: - Admin: Full access - User: Limited access to view records - Auditor: Read-only access Last Updated: 2024-08-14
```

Figure 3: Users with full and limited access to the database

Figure 3 showcases the distinction between users with full and limited access to the database. The experiment demonstrated that the system could effectively manage user roles, granting different levels of access based on permissions. This ensures that only authorized users have full access to sensitive data, while other users are restricted to limited functions, enhancing overall system security.

Figure 4. illustrates the data integrity checks performed within the system. It highlights the modified data and the time taken to complete these modifications.



Integrity Checks

Monitor system logs and visualize system events and activities.

```
Integrity Checks will be displayed here...
```

Integrity Checks

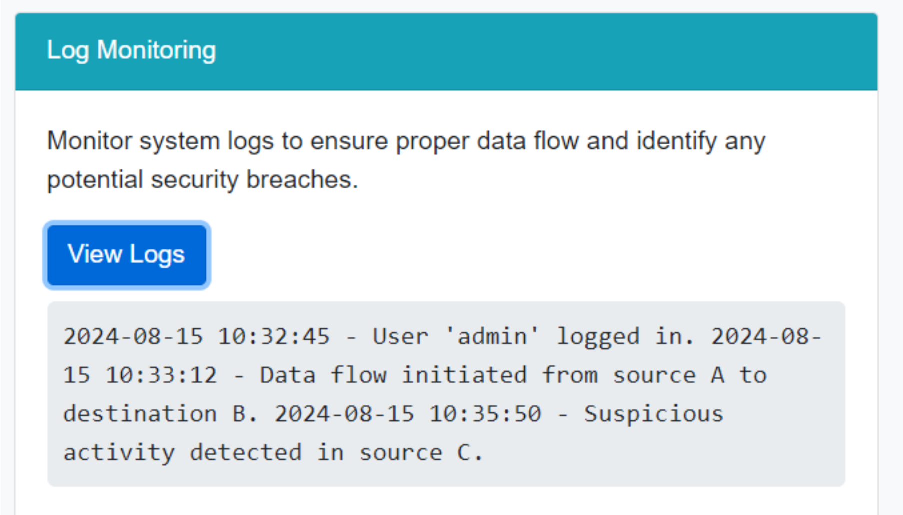
Monitor system logs and visualize system events and activities.

```
Mon, 16 Sep 2024 18:59:45 GMT
Original Name: Joanna
Original Data: 16
Changed Data: 24
```

Figure 4: Data integrity checks.

The experiment confirmed that the system accurately tracked all changes made to the data, providing crucial information such as timestamps and the nature of the modifications. These integrity checks are essential for ensuring that the data remains consistent, trustworthy, and unaltered by unauthorized users.

Figure 5 presents an overview of the log monitoring system, which tracks user logins.



Log Monitoring

Monitor system logs to ensure proper data flow and identify any potential security breaches.

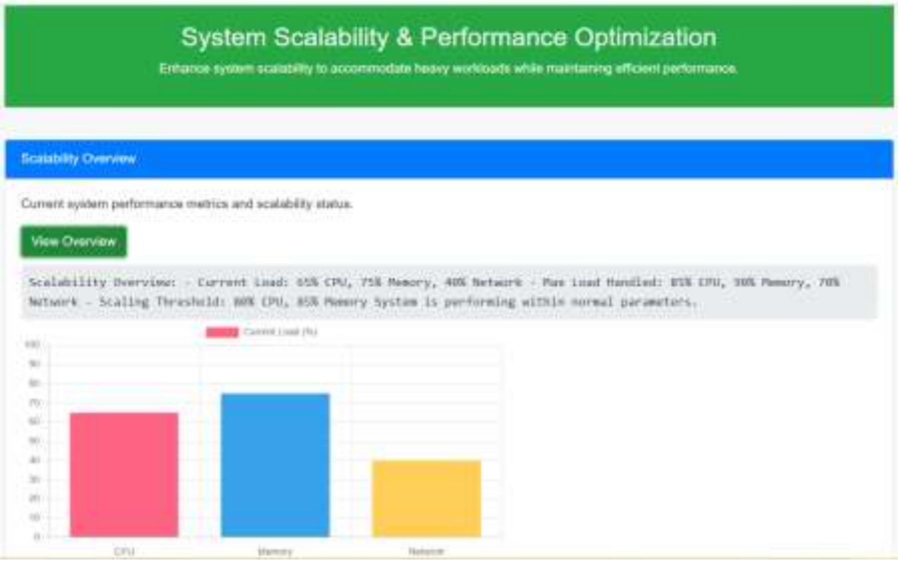
```
2024-08-15 10:32:45 - User 'admin' logged in. 2024-08-
15 10:33:12 - Data flow initiated from source A to
destination B. 2024-08-15 10:35:50 - Suspicious
activity detected in source C.
```

Figure 5 Overall Log Monitoring.

The figure displays when users logged into the system and the exact time of their login activity. Log monitoring is a key security feature that ensures accountability and transparency, allowing administrators to detect any suspicious login activity or unauthorized access attempts in real-time.

4.2 Scalability and System Expansion

Scalability was a key consideration during the system's implementation to accommodate future growth and additional features. The backend was designed to handle increased data loads and user traffic, ensuring that the system remains responsive and efficient even as more crime data is added. The use of Flask's modular architecture allows for easy integration of new functionalities, such as advanced analytics tools or additional encryption methods, without disrupting the existing system. The database schema was also designed with scalability in mind, supporting the storage of large volumes of encrypted crime data while maintaining quick retrieval times.



System Scalability & Performance Optimization

Enhance system scalability to accommodate heavy workloads while maintaining efficient performance.

Scalability Overview

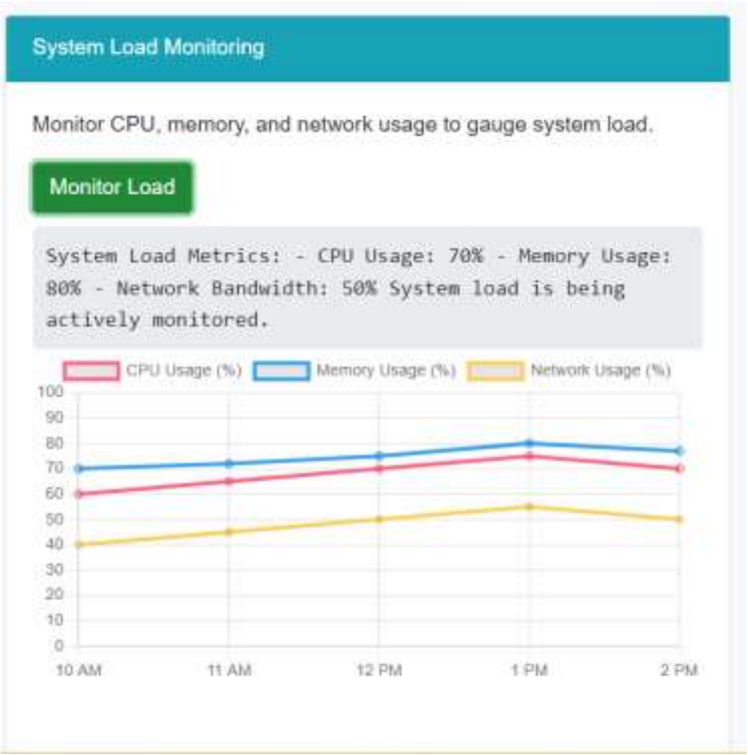
Current system performance metrics and scalability status.

```
Scalability Overview: - Current Load: 65% CPU, 75% Memory, 40% Network - Max Load Handled: 85% CPU, 90% Memory, 70% Network - Scaling Threshold: 80% CPU, 85% Memory System is performing within normal parameters.
```

Figure 6: Overview of system scalability

Figure 6 emphasize the system's capability to handle future growth. The backend was designed to support an increasing load of crime data and user traffic without compromising system performance. Flask's modular architecture facilitates easy integration of new features, such as advanced analytics and additional encryption, while the database was structured to maintain efficient retrieval times even with a growing volume of encrypted data.

Figure 7 shows the system load usage monitoring, depicting how the system managed varying workloads. This feature is crucial for understanding how the system performs under different conditions, allowing administrators to adjust resources as needed to prevent bottlenecks.



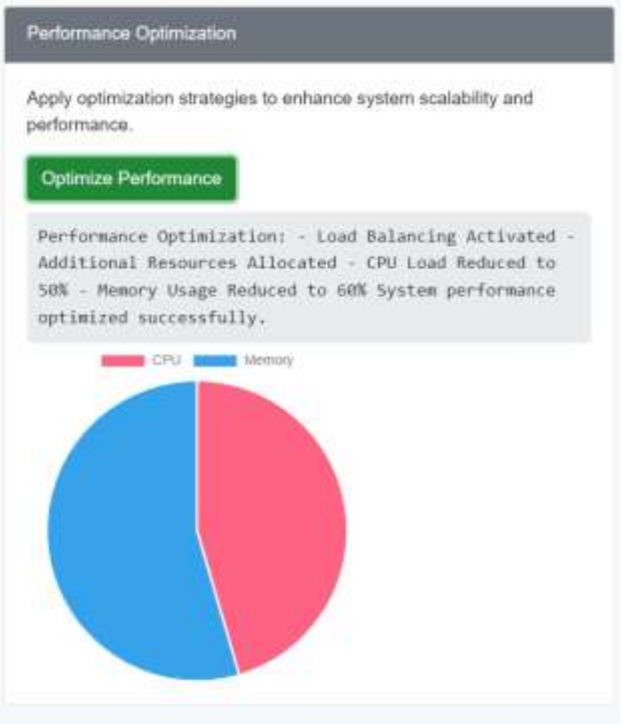
System Load Monitoring

Monitor CPU, memory, and network usage to gauge system load.

```
System Load Metrics: - CPU Usage: 70% - Memory Usage:
80% - Network Bandwidth: 50% System load is being
actively monitored.
```

Figure 7: System load usage monitoring

Figure 8 highlights the optimized performance of the system after load balancing was activated. It demonstrates the efficient use of memory and CPU, leading to an overall improvement in system responsiveness and stability during high-demand periods.



Performance Optimization

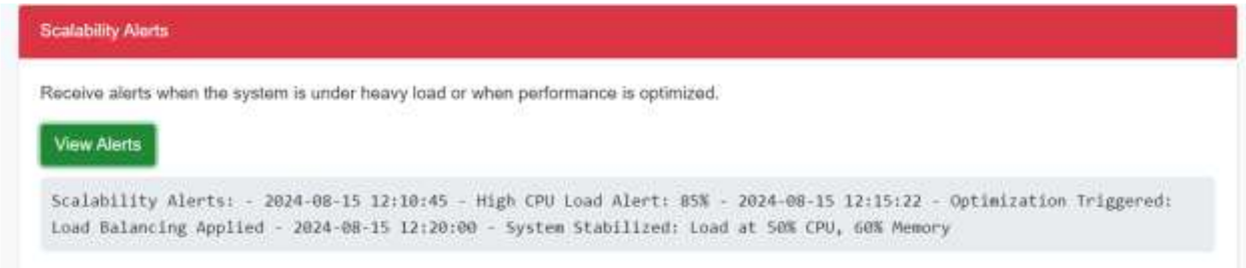
Apply optimization strategies to enhance system scalability and performance.

```
Performance Optimization: - Load Balancing Activated -
Additional Resources Allocated - CPU Load Reduced to
50% - Memory Usage Reduced to 60% System performance
optimized successfully.
```

Figure 8: Optimized performance

This shows the optimized usage of the memory and CPU when load balancing was activated

Figure 9 further illustrates the optimized load monitoring, showing how the system maintained balanced performance by distributing the computational load across different resources. This ensures consistent and reliable operation, especially during periods of increased activity.



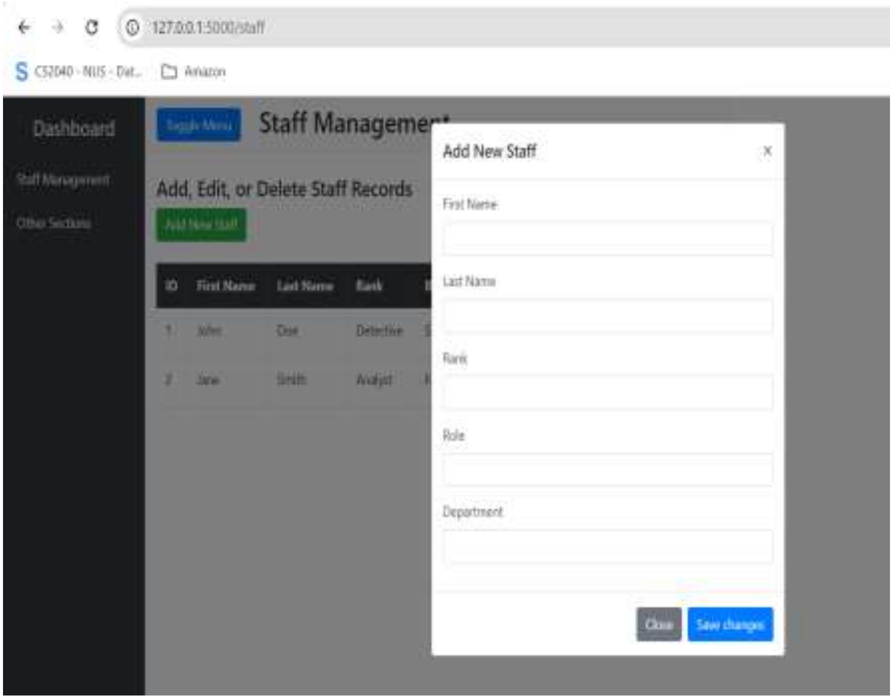
Scalability Alerts

Receive alerts when the system is under heavy load or when performance is optimized.

```
Scalability Alerts: - 2024-08-15 12:10:45 - High CPU Load Alert: 85% - 2024-08-15 12:15:22 - Optimization Triggered:
Load Balancing Applied - 2024-08-15 12:20:00 - System Stabilized: Load at 50% CPU, 60% Memory
```

Figure 9: Optimized load monitoring

Figure 10 focuses on staff management within the system, highlighting the ability to track and manage personnel involved in system operations. This feature ensures that staff-related tasks, such as user permissions and responsibilities, are organized and managed efficiently.



Staff Management

Add, Edit, or Delete Staff Records

ID	First Name	Last Name	Rank
1	John	Doe	Detective
2	Jane	Smith	Analyst

Add New Staff

First Name

Last Name

Rank

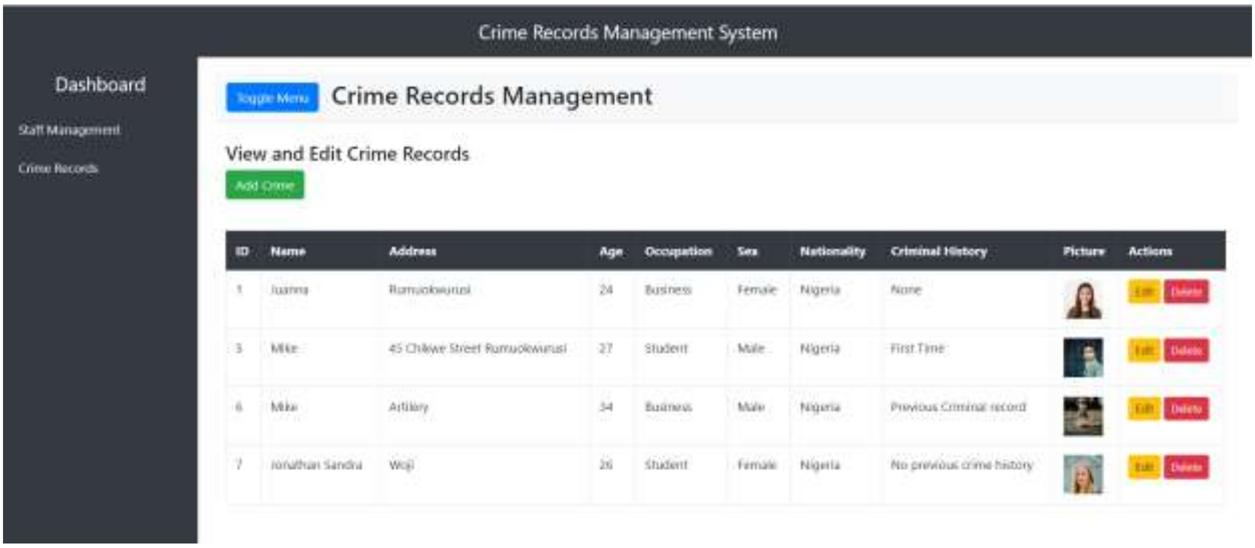
Role

Department

Figure 10: Staff Management

It holds the information about the incident. It contain the detail information of the officer in-charge of the incident name, rank, department etc.

Figure 11 displays the crime management feature, which showcases the system's capability to handle and organize crime data effectively. This aspect is critical for ensuring that the system can store, retrieve, and analyze crime-related information in an efficient and secure manner.



Crime Records Management System

Crime Records Management

View and Edit Crime Records

ID	Name	Address	Age	Occupation	Sex	Nationality	Criminal History	Picture	Actions
1	Juanna	Rumuokwurusi	24	Business	Female	Nigeria	None		Edit Delete
3	Mike	45 Chikwe Street Rumuokwurusi	27	Student	Male	Nigeria	First Time		Edit Delete
6	Mike	Artillery	34	Business	Male	Nigeria	Previous Criminal record		Edit Delete
7	Jonathan Sandra	Woji	26	Student	Female	Nigeria	No previous crime history		Edit Delete

Figure 11: Crime Management

This hold the information about the criminal such as the name, address, age, occupation, nationality criminal history picture etc.

Figure 12 and Figure 13 shows the encrypted and decrypted file based on the file load and its encryption and decryption time.

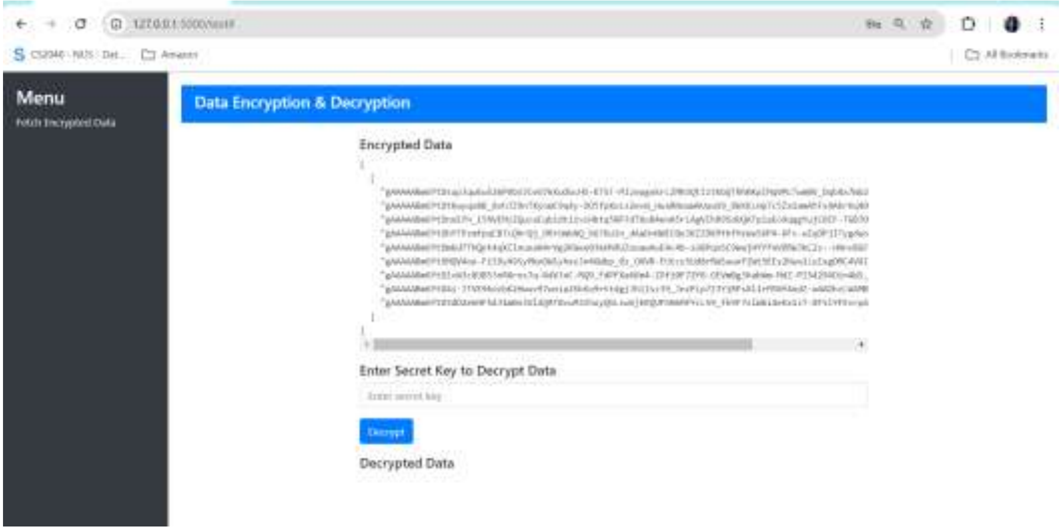


Figure 12: Encrypted data

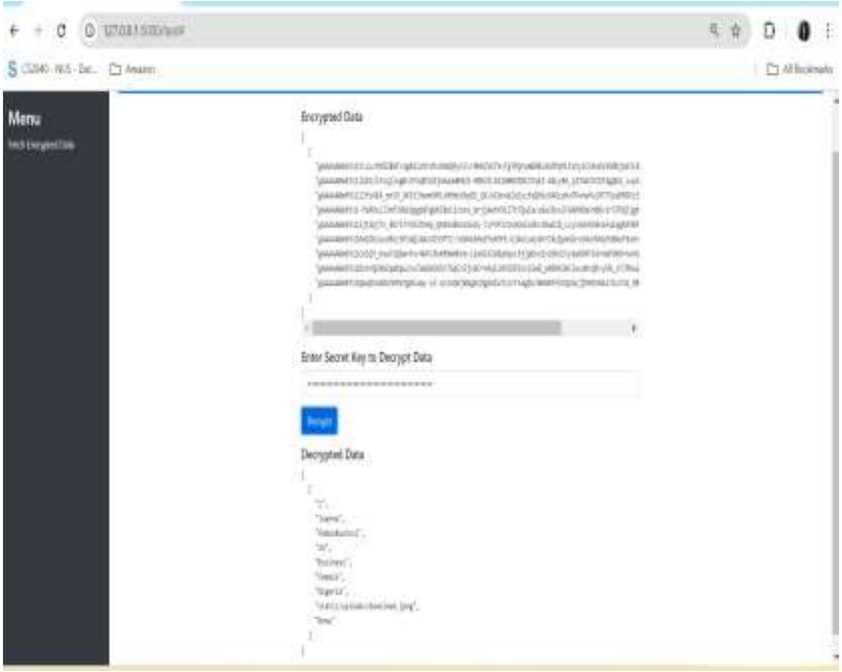


Figure 13: Decrypted data

4.3 Comparison with an Existing System

The result of the system was compared with an existing system by Chougule et al. (2022) in terms of encryption and decryption time. The proposed system uses Advanced encryption Standard (AES) while the existing uses proxy re-encryption technique. The compared results can be found in Table 1 and Figure 14

Table 1 Comparison Table with the existing system

File Size(KB)	Criteria	Existing System (Proxy Re-Encryption)	Proposed System (Advanced Encryption)
File A 50	Encryption Time	0.02s	0.03s
	Decryption Time	0.01s	0.02s
File B 100	Encryption Time	0.05s	0.03s
	Decryption Time	0.04s	0.02s
File C 1000	Encryption Time	0.15s	0.10s
	Decryption Time	0.12s	0.08s

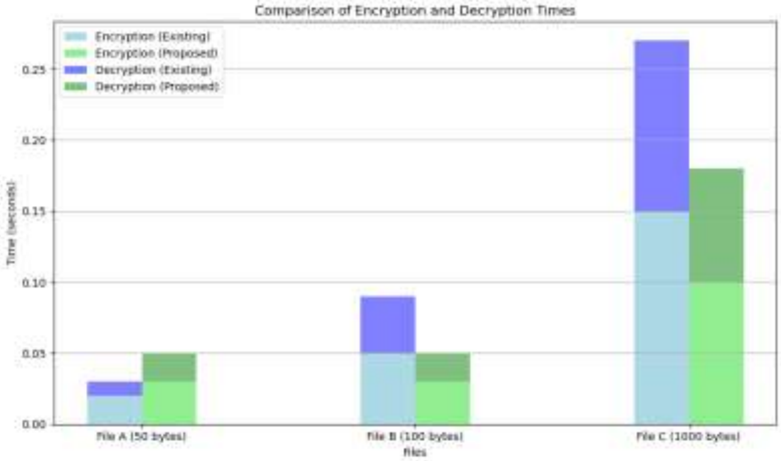


Figure 4.12 Comparison result

5. Conclusion

The dissertation developed a secure crime management system by focusing on access control mechanisms, custom checks, scalability, and confidentiality measures. These measures ensured that only authorized users could access sensitive crime data, maintaining data integrity. Customized checks ensured data validation and verification, minimizing errors and inconsistencies. The system was designed to handle heavy workloads without compromising performance, with a modular architecture for seamless integration. Advanced encryption methods were implemented to protect data confidentiality and privacy. The overall system architecture was designed with a comprehensive approach to security, ensuring data integrity and security throughout. This comprehensive design delivered a robust and scalable crime management system, meeting the project's goals and providing a reliable tool for law enforcement agencies.

References

[1] Janis Wong1, Tristan Henderson1, Kirstie Ball2., (2022) Data protection for the common good:Developing a frame work for a data protection-focused data commons Data & Policy 4: e3 doi:10.1017/dap.2021.40.

[2] Kashmar, N., Adda, M., Atieh, M. (2020). From Access Control Models to Access Control Metamodels:A Survey. In: Arai, K., Bhatia, R. (eds) Advances in Information and Communication. FICC 2019. Lecture Notes in Networks and Systems, vol 70. Springer, Cham. https://doi.org/10.1007/978-3-030-12385-7_61.

[3] Ebert, I., Wildhaber, I., & Adams-Prassl, J. (2021). Big Data in the workplace: Privacy Due Diligence as a human rights-based approach to employee privacy protection. Big Data & Society, 8(1). https://doi.org/10.1177/20539517211013051

[4] Thomas, J., Güez, S., (2018) Protecting Human Subjects in the Digital Age: Issues and Best Practices of Data Protection: https://doi.org/10.29115/SP-2018-0028.

[5] Alyaa H., Aya A, Mariam K, Nour A, Sally K., (2021) Modelling of Crime Record Management System Using Unified Modeling Language https://doi.org/10.18280/isi.260404

[6] Sachoulidou, A. (2023). Going beyond the “common suspects”: to be presumed innocent in the era of algorithms, big data and artificial intelligence. Artificial Intelligence and Law. https://doi.org/10.1007/s10506-023-09347-w

[7] Ajah, B., Chinweze, U., Ajah, I., Onyejegbu, D., Obiwulu, A., Onwuama, E., … & Okpa, J. (2022). Behind bars but not sentenced: the role of computerized central repository in addressing awaiting-trial problems in ebonyi state, nigeria. Sage Open, 12(1). https://doi.org/10.1177/21582440221079822

[8] Biswajit Debnath, Jaafar M. Alghazo, Ghanzafar Latif, Reshma Roychoudhuri, Sadhan Kumar Ghosh (2020).

An Analysis of Data Security and Protection Threat from IT Assets for middle card player, institutions and individual. *Sustainable waste management: policies and case studies: 7th Icon SWM-ISWMAW2017* Volume 1, 403-419.

[9] Cleophas Mutundu Ambira, Henry Nyabuto Kemoni, Patrick Ngulube (2019) A framework for electronic record management in support of e-government in Kenya. *Record Management Journal 29 (3)*, 305-319.

[10] Sarumi Jerry, (2022) A review of encryption methods for secure data communication. 10.22624

[11] Servos, D. and Osborn, S.L., (2017) Current Research and Open Problems in Attribute-.Based Bccess Control, ACM Computing Surveys, 49(4): 1-45

[12] Shakti, D., Rai, P. K., (2021) A Review of Cloud Service Security with various Access Control methods *international journal of computer science and mobile computing*. Vol 10, pg 39 – 45.

[13] Chougule, H., Dhadiwal, S., Lokhande, M., Naikade, R., and Patil, R., (2022) Digital Evidence Management System for Cybercrime Investigation using Proxy Re-encryption and Blockchain *Procedia Computer Science* 215, 71-77.