JOURNAL INNOVATIONS COMPUTER SCIENCE
E-ISSN: 2961-970X, P-ISSN: 2961-9718
Volume 4 Number 1, May 2025, pp 1-17.
DOI: https://doi.org/10.56347/jics.v4i1.217

Web-Based Network Anomaly Detection System for Disaster Recovery Center: A SIEM Implementation at the Indonesian Attorney General Training Agency

Issenoro [a], Herlina Trisnawati [b], Sakius Octavianus Tarigan [c*], Novianti M Faizah [d]
[a*,b,c,d] Information Systems Study Program, Faculty of Engineering, Universitas Tama Jagakarsa, South Jakarta City, Special Capital Region of Jakarta, Indonesia.

ABSTRACT
This research focuses on developing an anomaly detection application for the internet network infrastructure at the Disaster Recovery Center (DRC) building of the Indonesian Attorney General's Training Agency through Security Information and Event Management (SIEM) implementation utilizing Python programming language. The primary objective of this study is to develop a comprehensive application that assists personnel, particularly network administrators at the DRC facility, in monitoring and analyzing internet network communication patterns and traffic flows. The research methodology involves creating a detection system designed to enhance network security capabilities and provide continuous monitoring functionality for network infrastructure protection. The developed application leverages SIEM technology to aggregate and process security-related information extracted from log data across network devices, applications, and hardware components. SIEM technology demonstrates the capability to handle substantial data volumes while correlating and analyzing security events from multiple heterogeneous sources within the network environment. The implementation of this application provides critical visibility into the internal network operations of the DRC facility, enabling proactive threat detection and response capabilities. When security incidents or anomalous activities are identified, the system generates comprehensive reports detailing network conditions and security status, which are subsequently escalated to management for appropriate remedial actions and strategic decision-making.

ARTICLE HISTORY
Received 10 October 2024
Accepted 10 May 2025
Published 30 May 2025

KEYWORDS
Web Application; Internet Network; Python; SIEM; Anomaly Detection; Network Security.

1. Introduction

The rapid advancement of information and communication technology in today's digital era has brought significant impact to various aspects of life, including the management of information technology infrastructure in government institutions. As dependence on digital systems increases, the need for network security and service continuity becomes increasingly crucial, especially for institutions that manage sensitive data and critical systems such as the Indonesian Attorney General's Office. The Disaster Recovery Center (DRC) building of the Indonesian Attorney General's Office, constructed around 2010, serves as a vital infrastructure that functions as a storage center for network equipment and backup servers from various implemented systems. The disaster recovery center concept itself represents an integral part of business continuity planning strategy aimed at ensuring the continuity of information system

CONTACT Issenoro issenoro@gmail.com Information Systems Study Program, Faculty of Engineering, Universitas Tama Jagakarsa, South Jakarta City, Special Capital Region of Jakarta, Indonesia.
© 2025 Information KAWANAD Institute

operations when disruptions or disasters occur in the main system (Mustakim, 2021). The primary function of the DRC building is to house backup servers that can be operated and utilized when the main server experiences disruptions or damage, ensuring uninterrupted service delivery.

The internet network infrastructure at the DRC building currently supports existing system operations, but remains limited from a security perspective. The existing security system relies solely on a firewall installed on the router provided by the internet service provider. This situation reveals a gap in implementing defense-in-depth strategy that should be applied to protect critical infrastructure such as disaster recovery centers. According to Network Security (2022), effective network security requires a layered approach that includes various security components, not just relying on a single point of defense. The limitations of existing security systems present significant risks to data security and service continuity. Increasingly complex and sophisticated cybersecurity threats require proactive and comprehensive security solutions. Mitre (no date) emphasizes the importance of implementing monitoring and detection systems capable of identifying threats in real-time to prevent or minimize the impact of cyber attacks. To address these problems, implementation of additional network security devices and monitoring systems that can secure networks more effectively is required. One solution that can be applied is developing anomaly detection applications utilizing Security Information and Event Management (SIEM) technology. According to Cyber Academy Indonesia (2022), SIEM represents a security solution that combines Security Information Management (SIM) and Security Event Management (SEM) to provide real-time analysis of security alerts generated by applications and network devices.

SIEM technology possesses the capability to collect, analyze, and manage security information originating from various log data sources on networks, applications, and hardware. The main advantage of SIEM lies in its ability to collect large volumes of data and connect and analyze events from various heterogeneous sources. Rijal Kamal and Andri Setiawan (2021) demonstrated that SIEM implementation can significantly improve threat detection capabilities and security incident response. Research conducted by Sijabat and Evo (2023) regarding SIEM design for detecting incidents on websites shows that SIEM implementation can provide comprehensive visibility into network activities and enable early detection of suspicious activities or anomalies that may indicate security threats. This aligns with the needs of the DRC building, which requires a monitoring system that can provide early warnings against potential security threats.

When developing anomaly detection applications, selecting appropriate technology and tools becomes a crucial factor for successful implementation. Python as a programming language has proven effective in developing security applications and data analysis. Rahmat and Nugroho (2021) explain that Python has a rich ecosystem of libraries for data analysis and machine learning, which strongly supports the development of sophisticated anomaly detection systems. For developing responsive and user-friendly interfaces, web frameworks such as Flask can be utilized. Tutorial Flask (no date) explains that Flask is a lightweight and flexible Python web framework, very suitable for developing web applications that require high customization such as security monitoring applications. Combination with Bootstrap 5 as a CSS framework can produce modern and responsive interfaces, as explained by Carnes (2021) in a comprehensive Bootstrap 5 tutorial.

The aspect of storing and managing large log data requires robust and scalable database systems. MySQL as a relational database management system has proven reliable in handling large and complex data volumes. MySQL Tutorial (no date) explains various features and optimizations that can be applied to improve database performance in applications that process log data intensively. Network infrastructure management requires deep understanding of network device configuration and

management such as MikroTik. Dimovan (2019) explains various configuration and monitoring aspects that can be integrated with SIEM systems to provide more comprehensive visibility into network activities. The history and development of the internet explained by Choiri (2021) shows how the complexity of modern internet networks requires sophisticated security approaches. Security threats are no longer limited to conventional attacks but have evolved into advanced persistent threats that require sophisticated and adaptive detection systems.

Developing anomaly detection applications also requires systematic methodological approaches. Prihandoyo (2018) explains the importance of good system modeling in developing complex applications. UML Class Diagram Tutorial (2024) provides comprehensive guidance for designing scalable and maintainable system architecture. Based on needs analysis and existing conditions at the DRC building of the Indonesian Attorney General's Office, developing anomaly detection applications with SIEM implementation becomes an appropriate solution for improving network security posture. The application is expected to provide real-time monitoring capabilities, accurate anomaly detection, and comprehensive reporting to support quick and appropriate decision-making when facing cybersecurity threats.

Implementation of the application will provide significant added value for DRC building operations, not only in security aspects but also in improving IT infrastructure management efficiency. With proactive monitoring systems, potential downtime can be minimized and service continuity can be maintained properly, in accordance with the main function of disaster recovery centers as reliable and secure backup systems. The development approach utilizes Python programming language due to its extensive libraries and frameworks that support security application development. The integration of SIEM technology enables the collection and analysis of security information from various sources, providing network administrators with the tools necessary to identify and respond to potential threats effectively. The application architecture incorporates MySQL database management for efficient log data storage and retrieval, while Flask framework provides the foundation for building an intuitive web-based interface. Network infrastructure at the DRC building currently lacks sophisticated monitoring capabilities, making it vulnerable to undetected security incidents. The proposed anomaly detection application addresses these vulnerabilities by implementing continuous monitoring and automated alert systems. The SIEM implementation enables correlation of events from multiple sources, providing a holistic view of network security status and potential threats.

2. Methodology

This research focuses on the DRC Building of the Indonesian Attorney General's Office as the primary research subject, employing a systematic and comprehensive research methodology. According to Sugiyono (2019), qualitative research methodology requires systematic data collection approaches that combine various techniques to ensure thorough understanding of the research subject. The implemented methodology utilizes three main data collection techniques: direct observation, structured interviews, and in-depth system analysis.

2.1 Data Collection Techniques

The observation technique involves direct examination of network infrastructure within the DRC Building to understand current topology and operational procedures in practice. Creswell (2018) emphasizes that direct observation provides researchers with immediate insights into real-world conditions and operational challenges that may not be apparent through other data collection methods. Through systematic observation, the research reveals that the DRC Building utilizes a hybrid network topology combining

tree, star, and mesh configurations. This hybrid approach, as explained by Tanenbaum and Wetherall (2021), offers enhanced redundancy and fault tolerance but introduces complexity in network management and security monitoring. Structured interviews were conducted with key personnel, particularly the Internet Network Administrator, to gather technical specifications and operational requirements. The interview process focused on understanding current security challenges, system limitations, and operational requirements for anomaly detection capabilities. Kvale and Brinkmann (2020) highlight that expert interviews provide valuable insights into system requirements and operational constraints that influence design decisions. These conversations proved invaluable in identifying the specific needs and pain points that the proposed system would need to address.

2.2 System Requirements Analysis

The system development approach follows the Software Development Life Cycle (SDLC) waterfall model, which provides a sequential and systematic framework for application development. Pressman and Maxim (2020) explain that the waterfall model ensures thorough documentation and systematic progression through each development phase, making it particularly suitable for security-critical applications where requirements must be clearly defined before implementation begins. This methodical approach helps minimize risks and ensures that all stakeholders have a clear understanding of the development process. Software requirements analysis identified the need for a Python-based development environment supported by Windows 11 operating system, XAMPP for local development server capabilities, and Visual Studio Code as the integrated development environment. According to Lutz (2019), Python's extensive library ecosystem and strong support for data analysis and security applications make it an optimal choice for developing sophisticated anomaly detection systems. The selection of MySQL as the database management system aligns with the need for reliable data storage and retrieval capabilities essential for log management and analysis. Hardware requirements specification includes a high-performance development workstation featuring Intel Core i7 processor, 16GB RAM, 8GB VRAM, and 500GB SSD storage to ensure adequate processing power for real-time log analysis and anomaly detection algorithms. Network connectivity requirements include Cat6 RJ45 cables to support high-speed data transmission necessary for continuous monitoring operations. These specifications align with industry standards for security monitoring systems as outlined by the NIST Cybersecurity Framework (2018).

The system design phase incorporates multiple modeling approaches including Use Case Diagrams, Activity Diagrams, Class Diagrams, and Sequence Diagrams to ensure comprehensive system architecture documentation. Booch, Rumbaugh, and Jacobson (2020) emphasize that UML modeling provides standardized visualization techniques that facilitate clear communication between stakeholders and ensure systematic design documentation. This visual approach helps bridge the gap between technical requirements and user expectations.

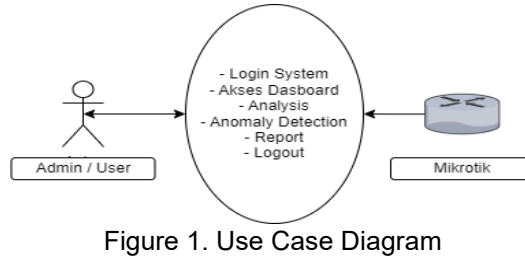

Figure 1. Use Case Diagram

The Use Case Diagram illustrates user interactions with the system, providing a clear overview of how different actors will engage with various system functionalities.

Meanwhile, Activity Diagrams detail the workflow processes for login, dashboard navigation, analysis functions, anomaly detection, reporting, and logout procedures, organized according to their operational sequence: User Login, Dashboard, Analysis, Anomaly Detection, Report, and Logout.

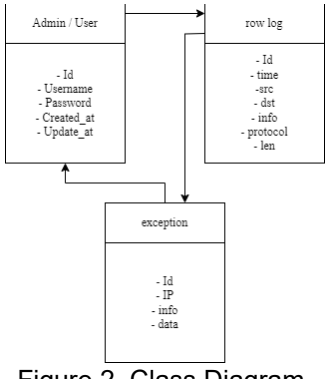

Figure 2. Class Diagram

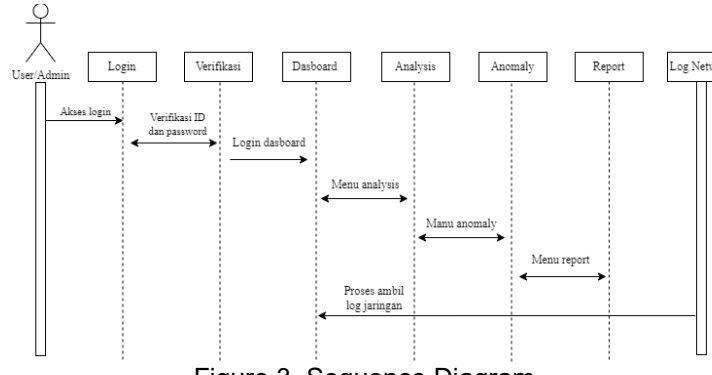

Figure 3. Sequence Diagram

Database design encompasses three primary tables: the users table for authentication management, the row_logs table for network traffic data storage, and the exceptions table for anomaly tracking. The users table includes identification fields, authentication credentials, and timestamp fields following security best practices outlined by OWASP (2021). The row_logs table captures essential network traffic information including timestamps, source and destination addresses, protocol information, and packet length data necessary for comprehensive traffic analysis. The exceptions table stores anomaly detection results with IP address identification, descriptive information, and detailed data for further investigation.

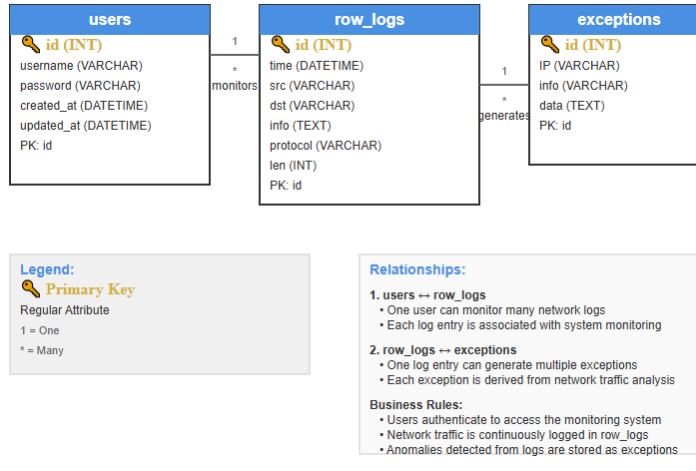

Figure 4. Database ERD for Anomaly Detection System

The coding phase utilizes Python programming language with Flask framework to create a lightweight yet powerful web application architecture. Grinberg (2018) demonstrates that Flask provides excellent flexibility for developing custom security applications while maintaining code simplicity and maintainability. The integration of Bootstrap framework ensures responsive user interface design that adapts to various screen sizes and devices, enhancing usability for network administrators who may access the system from different locations and devices. This flexibility is crucial in

today's dynamic work environment where administrators need to monitor systems remotely. The testing methodology employs black-box testing techniques focusing on functional requirements validation and output verification. Myers, Sandler, and Badgett (2019) explain that black-box testing effectively identifies functional errors and missing features without requiring detailed knowledge of internal code structure. The testing approach evaluates user interface functionality, data processing accuracy, anomaly detection effectiveness, and report generation capabilities to ensure system reliability and performance. This comprehensive testing strategy helps identify potential issues before deployment. System support requirements include both hardware and software components necessary for operational deployment. Hardware support encompasses standard computing equipment including CPU, storage devices, input/output peripherals, and network connectivity components. Software support includes XAMPP server environment, MySQL database management system, Apache web server, Bootstrap CSS framework version 3.3.5, jQuery JavaScript library version 1.11.3, Flask web framework version 0.11.1, Flask-WTF form handling version 0.12, geoip2 geolocation library, Draw.io diagramming tool, and Microsoft Edge browser for testing and operation.

2.3 System Display Prototype Design

The prototype design phase creates visual representations of key system interfaces including login authentication, dashboard overview, analysis visualization, anomaly detection alerts, comprehensive reporting, and secure logout functionality. Nielsen and Budiu (2021) emphasize that prototype design enables early user feedback and iterative improvement before full system implementation. This user-centered approach ensures that the final product meets actual user needs and expectations.

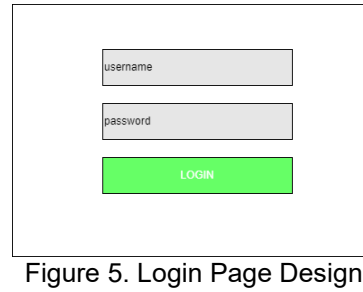

Figure 5. Login Page Design

The login interface incorporates security best practices with encrypted authentication mechanisms, ensuring that only authorized personnel can access the system. The design prioritizes both security and usability, providing a clean and intuitive interface that doesn't compromise on protection. Meanwhile, the dashboard provides intuitive navigation and real-time system status information, serving as the central hub for all system operations. The dashboard pages are designed to display main menu components consisting of Home Menu Page, Analysis Menu Page containing maps information, Anomaly Detection/Warning Page, Report Page, and Logout Page. Each component is carefully designed to provide maximum functionality while maintaining ease of use.

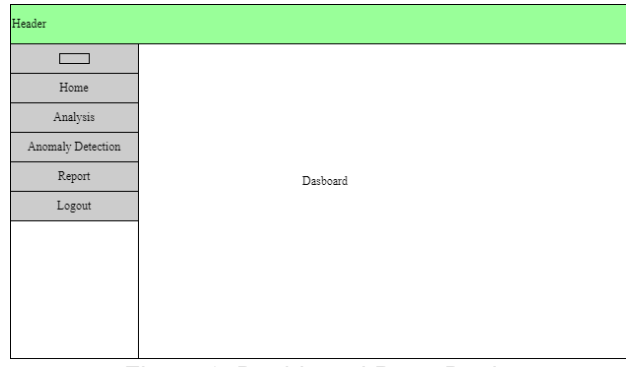

Figure 6. Dashboard Page Design

2.4 System Flowchart

System flowchart documentation illustrates the logical flow of operations from user authentication through anomaly detection and reporting processes. The flowchart serves as a visual guide for understanding system behavior and supports maintenance and future enhancement activities. This systematic approach ensures that all stakeholders understand system functionality and operational procedures, facilitating effective deployment and ongoing management of the anomaly detection application within the DRC Building infrastructure.

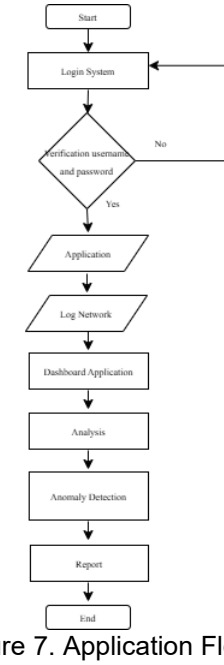

Figure 7. Application Flowchart

The comprehensive methodology outlined above provides a solid foundation for developing a robust and effective anomaly detection system. By combining established research techniques with modern development practices, this approach ensures that the resulting system will meet the specific needs of the DRC Building while maintaining high standards of security and reliability. The systematic documentation and modeling approach also facilitates future maintenance and enhancement activities, ensuring the long-term viability of the system.

3. Results

The implementation of a network anomaly detection system at the DRC Building of the Indonesian Attorney General's Office has successfully produced a web application capable of performing real-time network security monitoring and analysis. The system developed using the Flask framework with Python as the primary programming language has proven to provide an effective solution for network traffic monitoring and anomaly detection. Development results demonstrate that the waterfall methodology approach provides a systematic and organized development structure, enabling the development team to create applications that meet the operational needs of the DRC Building. The developed application system displays an intuitive and responsive user interface, specifically designed to meet the needs of network administrators in conducting security monitoring and analysis. Implementation using the Bootstrap framework ensures that the application can be accessed optimally through various devices and screen sizes, providing flexibility for users to access the system from different locations. The interface design prioritizes ease of navigation and information accessibility, with logical layouts and clear visual hierarchy to help users operate the system efficiently.

Figure 8. Login Page Display

The system login page displays a secure and user-friendly authentication interface, implementing security best practices in the user authentication process. The login page design integrates professional visuals with robust security functionality, including input validation and password encryption (Figure 8). The authentication system uses secure session management to ensure that only authorized users can access network monitoring features. Form validation implementation on the login page helps prevent injection attacks and ensures authentication data integrity. The login page display is also equipped with visual feedback that provides clear information to users regarding login process status, whether successful or failed.

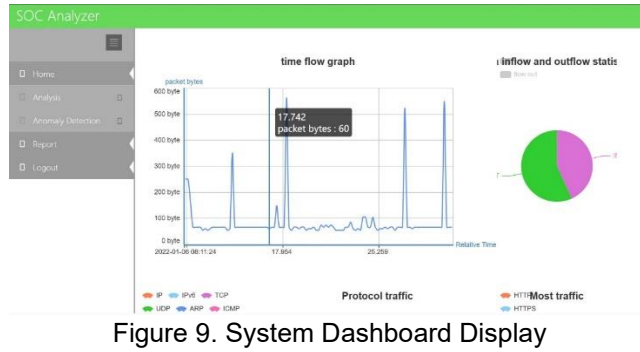

Figure 9. System Dashboard Display

The main system dashboard provides a complete overview of network status and ongoing monitoring activities. The dashboard interface is designed with a user-centered design approach that allows network administrators to quickly understand overall

network security conditions. The dashboard layout uses a responsive grid system, allowing important information to be displayed hierarchically based on priority and urgency levels. The main navigation menu on the dashboard provides easy access to various system modules, including analysis, anomaly detection, and reporting. Real-time updates implementation on the dashboard ensures that displayed information is always current and relevant to network conditions at that time.

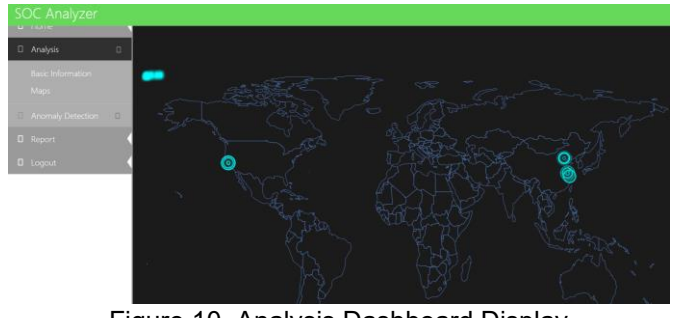

Figure 10. Analysis Dashboard Display

The analysis module in the system consists of two main parts that complement each other to provide a complete picture of network traffic conditions and communication patterns occurring within the DRC Building infrastructure. The Basic Information section displays fundamental statistics about network activity, including traffic volume, protocols used, and communication distribution between hosts. This basic information is displayed in an easily understandable format with informative graph and table visualizations. Statistical data includes temporal analysis showing network usage patterns over time, helping administrators identify normal trends and potential anomalies. Implementation of filtering and sorting on basic information data allows users to perform drill-down analysis according to specific investigation needs. The Maps feature provides geographical visualization of network traffic sources and destinations, utilizing geolocation technology to display geographical distribution of network connections. Interactive maps implementation using modern JavaScript libraries allows users to perform zoom, pan, and filtering based on specific geographical criteria. Maps visualization is very useful for identifying unusual communication patterns or connections from suspicious geographical locations. Integration with geolocation databases ensures location information accuracy and provides valuable context for network security analysis.

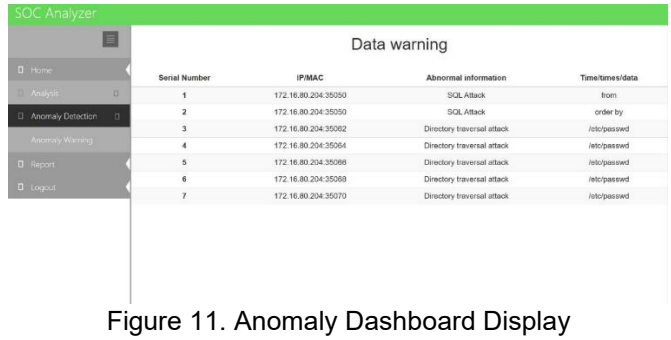

Figure 11. Anomaly Dashboard Display

The Anomaly Detection Dashboard displays real-time analysis results from anomaly detection algorithms that have been implemented in the system. This interface is specifically designed to provide alerts and warnings to network administrators when suspicious or abnormal activities are detected in network traffic. The anomaly

classification system uses color-coding to distinguish severity levels of each detected anomaly, enabling appropriate response prioritization. Filtering and search features on the anomaly dashboard help administrators conduct investigations into specific incidents. Notification system implementation ensures that anomaly alerts can be delivered in real-time to responsible security teams.

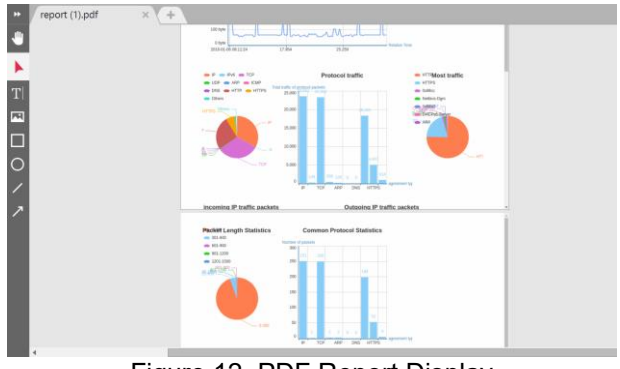

Figure 12. PDF Report Display

The integrated reporting system is capable of generating complete reports in PDF format that can be distributed to related stakeholders. The report generation feature uses customizable templates to meet different reporting needs, ranging from daily operational reports to monthly security analysis reports. Automated report scheduling implementation allows the system to generate reports automatically at predetermined time intervals. Generated PDF reports include graph visualizations, statistical tables, and narrative analysis that provide insights into network security conditions. The system also provides data export features in various formats to support further analysis using external tools.

4. Discussion

The implementation of network anomaly detection systems at the DRC Building of the Indonesian Attorney General's Office has demonstrated substantial improvements in network monitoring and security analysis capabilities. With growing demands for effective security solutions in network environments, machine learning (ML) has become increasingly relevant for modern anomaly detection applications. Recent studies show that algorithms like K-Nearest Neighbors (KNN) and ensemble-based models can be optimized to detect network data anomalies with high accuracy rates (Tama *et al*., 2020; Nassif *et al*., 2021). Additionally, Isolation Forest techniques in web traffic anomaly detection have proven effective in identifying abnormal patterns while maintaining low false positive rates (Chua *et al*., 2024). Integrating ML frameworks into the system effectively enhances the identification and response to potential network security threats.

Selecting Flask as the foundation for web application development proves appropriate due to its high flexibility in building complex systems while maintaining optimal performance. Flask enables lightweight web application development that adapts well to various enterprise requirements (Larasati & Susetyo, 2024). Research demonstrates that Flask framework implementation in information system development facilitates integration with various Python libraries and supports good scalability (Santoso & Saian, 2023; Hattu & Susetyo, 2024). Moreover, choosing Flask as the base framework supports integration with numerous Python libraries for data processing and analysis, as evidenced in various web-based system implementations

(Pilnenskiy & Smetannikov, 2020; Alnaasan *et al*., 2021). Libraries such as Pandas and NumPy, well-established in the scientific community, provide essential functions for efficient data processing in network analysis (Harris *et al*., 2020; Bullejos *et al*., 2022). Meanwhile, scikit-learn offers a user-friendly interface for implementing various ML algorithms, including KNN and other simple models that have proven effective in anomaly detection (Wang *et al*., 2022; Bullejos *et al*., 2022). Using Flask in inventory system development and operational applications demonstrates consistency in delivering robust and maintainable solutions (Izzathohir & Yulianton, 2024; Walingkas & Saian, 2023; Ma'arif & Kurniasih, 2024).

Analysis of the developed user interface shows that the user-centered design (UCD) approach successfully creates an intuitive user experience. The UCD approach prioritizes end-user needs, preferences, and limitations throughout every stage of product design, ensuring the built application meets user expectations optimally. Research indicates that UCD significantly improves user satisfaction and system interaction effectiveness (Ramdani *et al*., 2024; Fazil *et al*., 2024). The system implementation adopts design principles validated in previous research on web-based anomaly detection application development (Issenoro *et al*., 2025). Studies on UI/UX design using user-centered design methods show that proper UCD principle application results in better user acceptance and system usability (Putra *et al*., 2025a; Putra *et al*., 2025b). Evaluation and redesign of UI/UX on web applications further confirms the importance of user-centered approaches in creating effective interfaces (Romadhoni & Dirgahayu, 2024).

The main dashboard successfully provides a clear real-time overview of network conditions, integrating artificial intelligence technology for anomaly detection that enables more accurate decision-making (Octiva *et al*., 2024). Responsive grid systems allow critical information display with appropriate priority, helping network administrators quickly identify areas requiring special attention. Real-time dashboard updates become crucial factors in ensuring decision-making based on accurate and current data, aligning with research on anomaly checking optimization that emphasizes time efficiency in anomaly identification (Aditya & Dewi, 2024). The analysis module, consisting of Basic Information and Maps, provides two different yet complementary perspectives in network analysis. Basic Information successfully delivers statistical overviews that help administrators understand normal network traffic patterns, while the Maps feature provides geographical information valuable for identifying potential threats from unusual locations. Integrating both features creates strong analytical capabilities for security incident investigation, supporting a holistic approach to network security monitoring.

The Anomaly Detection Dashboard demonstrates effective machine learning algorithm implementation for detecting abnormal patterns in network traffic. Recent research shows that artificial intelligence-based anomaly detection technology over encrypted traffic provides robust solutions for modern threat detection (Ji *et al*., 2024). Using color-coding for severity classification enables administrators to perform rapid responses to detected threats. However, anomaly detection accuracy heavily depends on training data quality and algorithm parameter fine-tuning. Implementing real-time notification systems becomes essential for ensuring alerts reach responsible teams promptly. The PDF format reporting feature demonstrates the system's capability in generating professional documentation distributable to various stakeholders. Automated report generation becomes a significant value-added feature by reducing manual workload for administrators in routine report creation. Available template customization provides flexibility in adjusting report formats according to different organizational needs, aligning with reporting module implementations proven effective in various information systems (Ogotan & David, 2024).

From a technical perspective, the system implementation shows scalable and

maintainable architecture. Using modular design approaches enables easier future development and maintenance. However, performance optimization aspects require consideration, especially when processing network data volumes increase significantly. Database optimization and caching mechanisms need special attention to ensure long-term optimal system functionality. User experience evaluation shows the system successfully creates efficient workflows for network administrators. Intuitive navigation and well-structured information architecture enable users to quickly find needed information. Responsive design implementation ensures system accessibility from various devices, providing high operational flexibility.

System security aspects demonstrate adequate security best practices implementation for enterprise environments. Multiple layer authentication, input validation, and secure session management provide sufficient protection against common attack vectors. However, regular security assessments and penetration testing remain necessary to ensure the system can face evolving threat landscapes. Integration with existing network infrastructure at the DRC Building shows good compatibility, though fine-tuning several parameters is required for performance optimization. The system's capability in handling real-time data processing indicates the chosen architecture suits intensive network monitoring needs. From an operational efficiency perspective, the system successfully reduces manual effort required in network monitoring and significantly improves response time to potential security incidents. Automated alerting mechanisms and reporting capabilities provide substantial value for operational teams performing network monitoring and maintenance tasks. Implementation results show the developed network anomaly detection system has met established objectives and is ready for production environment deployment. Continuous monitoring and regular system updates will be key to ensuring the system continues providing optimal protection against evolving network security threats. The research contributes to developing practical network security solutions implementable in enterprise environments, aligning with trends in artificial intelligence use in big data systems for decision-making (Octiva *et al*., 2024).

5. Conclusion and Recommendations

Based on the comprehensive analysis presented in this study, the implementation of the network anomaly detection system at the Disaster Recovery Center (DRC) building of the Indonesian Attorney General's Office has successfully achieved its established objectives. The developed system demonstrates remarkable capability in providing real-time information regarding internet network traffic and anomaly detection, enabling network administrators to promptly identify and respond to potentially harmful network anomalies that could compromise system security. The system has proven highly effective in assisting network administrators with monitoring and surveillance of network security devices through comprehensive information provision regarding logs and events in internet network traffic. The integrated dashboard approach utilizing Security Information and Event Management (SIEM) principles has significantly enhanced operational efficiency by enabling centralized monitoring of various logs and events from multiple devices within a single platform. This consolidation has streamlined the monitoring process and reduced the complexity traditionally associated with managing multiple security tools simultaneously. The implementation of machine learning algorithms within the anomaly detection system has yielded satisfactory results in identifying abnormal network patterns with commendable accuracy levels. The strategic choice of Flask framework as the foundation for web application development has proven appropriate, offering high flexibility and seamless integration capabilities with various Python libraries that support data analysis and machine learning algorithm implementation.

The user-centered design approach has resulted in an intuitive interface that enhances user experience and operational effectiveness, while the modular architecture ensures system scalability and maintainability for future enhancements.

The successful implementation of this network anomaly detection system opens several avenues for future improvements and expansions. First and foremost, the Disaster Recovery Center of the Indonesian Attorney General's Office should consider integrating Security Orchestration, Automation and Response (SOAR) capabilities into the existing network security infrastructure. SOAR systems possess advanced automation capabilities for preventing and responding to network anomalies deemed dangerous, significantly enhancing the speed and effectiveness of security incident response. This integration would enable the system to not only detect threats but also automatically execute mitigation actions according to predefined playbooks, reducing response time and human error in critical situations. The development of remote access capabilities represents another crucial enhancement opportunity. The implemented system should be expanded to allow secure remote access from various locations and devices, including mobile platforms. This enhancement would substantially improve the effectiveness of network administrator performance by enabling continuous monitoring and incident response capabilities regardless of physical location. Implementing secure remote access with multi-factor authentication would ensure operational flexibility without compromising security aspects, particularly important in today's increasingly distributed work environments. Continuous improvement of machine learning algorithms used in the anomaly detection system should be prioritized through regular evaluation and enhancement initiatives. The implementation of ensemble learning techniques and deep learning approaches could significantly improve detection accuracy while reducing false positive rates. Additionally, regular updates to training datasets are essential to ensure the system can recognize emerging threat patterns and adapt to evolving security landscapes. This continuous learning approach will maintain the system's effectiveness against sophisticated and constantly evolving cyber threats. The implementation of robust backup and disaster recovery systems becomes critical given the vital importance of continuous security monitoring operations. This includes establishing regular data backup procedures, system replication to alternative locations, and thoroughly tested recovery procedures to ensure minimal downtime during system failures. Such measures are particularly crucial for a disaster recovery center, where system availability directly impacts the organization's ability to maintain operations during critical situations. Establishing comprehensive training and human resource development programs for network administrators and IT security teams regarding the implemented system's utilization is essential. These training programs should cover anomaly detection result interpretation, incident response procedures, and system maintenance protocols to ensure optimal utilization of technological investment. Regular training updates should be conducted to keep pace with system enhancements and emerging security threats, ensuring that human resources remain capable of effectively leveraging the advanced capabilities provided by the implemented system.

References

Aditya, M. R., & Dewi, C. (2024). Optimisasi pengecekan anomali pada proses job: Analisis waktu dan data untuk identifikasi anomali yang efisien. *Jurnal Indonesia: Manajemen Informatika Dan Komunikasi*, *5*(2), 1819-1832. https://doi.org/10.35870/jimik.v5i2.737

Alnaasan, N., Jain, A., Shafi, A., Subramoni, H., & Panda, D. (2021). *OMB-Py: Python micro-benchmarks for evaluating performance of MPI libraries on HPC systems*. https://doi.org/10.48550/arxiv.2110.10659

AMT-IT. (2022, March 15). *Network security: Tingkatkan keamanan jaringan perusahaan*. https://amt-it.com/blog/network-security-adalah/

Bootstrap. (2024). *Build fast, responsive sites with Bootstrap*. https://getbootstrap.com/

Bullejos, M., Cabezas, D., Martín-Martín, M., & Alcalá, F. (2022). A k-nearest neighbors algorithm in Python for visualizing the 3D stratigraphic architecture of the Llobregat River Delta in NE Spain. *Journal of Marine Science and Engineering*, *10*(7), 986. https://doi.org/10.3390/jmse10070986

Carnes, B. (2021, May 17). *Full Bootstrap 5 tutorial for beginners*. freeCodeCamp. https://www.freecodecamp.org/news/full-bootstrap-5-tutorial-for-beginners/

Choiri, E. O. (2021, August 12). *Sejarah singkat internet & perkembanganya sampai saat ini*. Qwords. https://qwords.com/blog/sejarah-singkat-internet/

Chua, W., Pajas, A., Castro, C., Panganiban, S., Pasuquin, A., Purganan, M., Salonga, K., Velasco, L., & Velasco, L. (2024). Web traffic anomaly detection using isolation forest. *Informatics*, *11*(4), 83. https://doi.org/10.3390/informatics11040083

Cyber Academy Indonesia. (2022, June 8). *Perbedaan SIEM dan SOAR*. https://www.cyberacademy.id/blog/perbedaan-siem-dan-soar

Dimovan, M. (2019). *Dasar-dasar Mikrotik*. http://103.44.149.34/elib/assets/buku/Dasar_Mikrotik.pdf

Fazil, A. W., Hamidi, S. A., & Habibi, H. (2024). Evaluating the impact of emerging technologies on mobile user experience: The role of user-centered design in overcoming development challenges. *International Journal Software Engineering and Computer Science (IJSECS)*, *4*(3), 1244–1252. https://doi.org/10.35870/ijsecs.v4i3.3167

Harris, C., Millman, K., Walt, S., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N., Kern, R., Picus, M., Hoyer, S., Kerkwijk, M., Brett, M., Haldane, A., Río, J., Wiebe, M., Peterson, P., ... Oliphant, T. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357-362. https://doi.org/10.1038/s41586-020-2649-2

Hattu, A., & Susetyo, Y. (2024). Development of operational application system at PT. XYZ with Flask overriding. *International Journal Software Engineering and Computer Science (IJSECS)*, *4*(1), 312–320. https://doi.org/10.35870/ijsecs.v4i1.2318

Issenoro, Trisnawati, H., Tarigan, S. O., Faizah, N. M., & Veranita. (2025). Perancangan dan pengembangan aplikasi deteksi anomali pada jaringan internet gedung disaster recovery center Badan Diklat Kejaksaan RI dengan implementasi sistem manajemen informasi dan keamanan (SIEM) berbasis web. *Jurnal Ilmu Komputer Dan Teknologi Informasi*, *2*(1), 12-21. https://doi.org/10.35870/jikti.v2i1.1341

Izzathohir, K. M., & Yulianton, H. (2024). Sistem aplikasi penjualan gula aren berbasis web menggunakan framework Flask. *Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi)*, *8*(1), 163–169. https://doi.org/10.35870/jtik.v8i1.1332

Ji, I., Lee, J., Kang, M., Park, W., Jeon, S., & Seo, J. (2024). Artificial intelligence-based anomaly detection technology over encrypted traffic: A systematic literature review. *Sensors*, *24*(3), 898. https://doi.org/10.3390/s24030898

Kamal, M. R., & Setiawan, M. A. (2021). Deteksi anomali dengan Security Information and Event Management (SIEM) Splunk pada jaringan UII. *Automata*, *2*(2), 1–6. https://journal.uii.ac.id/AUTOMATA/article/view/19522

Larasati, S., & Susetyo, Y. A. (2024). Development of a web-based trading term application using Flask framework at PT. XYZ. *International Journal Software Engineering and Computer Science (IJSECS)*, *4*(1), 367–376. https://doi.org/10.35870/ijsecs.v4i1.2339

Ma'arif, O. M., & Kurniasih, T. (2024). Perancangan sistem inventory berbasis web menggunakan framework Flask: PT. Gagas Mitra Jaya (Area Salatiga). *Jurnal Indonesia: Manajemen Informatika Dan Komunikasi*, *5*(2), 1947-1959. https://doi.org/10.35870/jimik.v5i2.822

MITRE Corporation. (2024). *Cybersecurity*. https://www.mitre.org/focus-areas/cybersecurity

Mustakim, A. (2021, March 16). *Solusi keamanan siber untuk perusahaan*. ACS Group. https://acsgroup.co.id/id/2021/03/16/indonesia-solusi-keamanan-siber-untuk-perusahaan/

MySQL Tutorial. (2024). *MySQL tutorial*. https://www.mysqltutorial.org/

Nassif, A., Talib, M., Nasir, Q., & Dakalbab, F. (2021). Machine learning for anomaly detection: A systematic review. *IEEE Access*, *9*, 78658-78700. https://doi.org/10.1109/access.2021.3083060

NgodingData. (2024). *Tutorial Flask – Web framework Python*. https://ngodingdata.com/tutorial-flask-web-framework-python/

Octiva, C. S., Suryadi, D., Judijanto, L., Laia, M., & Irwan, D. (2024). The application of artificial intelligence for anomaly detection in big data systems for decision-making. *International Journal Software Engineering and Computer Science (IJSECS)*, *4*(3), 983–989. https://doi.org/10.35870/ijsecs.v4i3.3358

Ogotan, T. E., & David, F. (2024). Pengembangan modul stock fulfillment online transaction pada aplikasi distribution center system menggunakan framework Flask di PT.XYZ. *Jurnal Indonesia: Manajemen Informatika Dan Komunikasi*, *5*(2), 1485-1494. https://doi.org/10.35870/jimik.v5i2.696

Pilnenskiy, N., & Smetannikov, I. (2020). Feature selection algorithms as one of the Python data analytical tools. *Future Internet*, *12*(3), 54. https://doi.org/10.3390/fi12030054

Prihandoyo, M. T. (2018). Unified Modeling Language (UML) model untuk pengembangan sistem informasi akademik berbasis web. *Jurnal Informatika: Jurnal Pengembangan IT*, *3*(1), 126-129. https://doi.org/10.30591/jpit.v3i1.765

Putra, A. A. A. W., Wanditya, I. M. D., & Fhadillah, M. L. H. (2025). Pengembangan antarmuka website kebugaran Fitme untuk mendukung pola hidup sehat dengan penerapan metode human-centered design. *Jurnal Indonesia: Manajemen Informatika Dan Komunikasi*, *6*(1), 161-173. https://doi.org/10.35870/jimik.v6i1.1136

Putra, A. A. A. W., Wirdianthi, N. L. R. P., & Azzaky, R. K. (2025). Perancangan UI/UX aplikasi stunting your buddy dengan metode user-centered design. *Jurnal Indonesia: Manajemen Informatika Dan Komunikasi*, *6*(1), 115-127. https://doi.org/10.35870/jimik.v6i1.1132

Python Software Foundation. (2024). *Python*. https://www.python.org/

Rahmat, B. (2021). *Pemrograman deep learning dengan Python*. CV. Indomedia Pustaka.

Ramdani, R. M., Yuniarti, R., & Komarudin, A. (2024). Interaction design on basic hand movement training game in taekwondo using user-centered design. *International Journal Software Engineering and Computer Science (IJSECS)*, *4*(1), 339–349. https://doi.org/10.35870/ijsecs.v4i1.2314

Romadhoni, M. N., & Dirgahayu, T. (2024). Evaluasi dan redesain UI/UX pada aplikasi web Young on Top. *Jurnal Indonesia: Manajemen Informatika Dan Komunikasi*, *5*(3), 2390-2401. https://doi.org/10.35870/jimik.v5i3.909.

Santoso, B. B., & Saian, P. O. N. (2023). Implementasi Flask framework pada development modul reporting aplikasi sistem informasi helpdesk di PT.XYZ. *Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi)*, *7*(2), 217–226. https://doi.org/10.35870/jtik.v7i2.718

Sijabat, D. R., & Evo, S. (2023). Perancangan Security Information and Event Management (SIEM) untuk mendeteksi insiden pada situs web. *J-Intech*, *11*(1), 10–17. https://doi.org/10.32664/j-intech.v11i1.860

Tama, B., Nkenyereye, L., Islam, S., & Kwak, K. (2020). An enhanced anomaly detection in web traffic using a stack of classifier ensemble. *IEEE Access*, *8*, 24120-24134. https://doi.org/10.1109/access.2020.2969428

Tutorial Mikrotik. (2024). *Tutorial Mikrotik*. https://tutorialmikrotik.com/

Visual Paradigm. (2024). *UML class diagram tutorial*. https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/

W3Schools. (2024). *MySQL tutorial*. https://www.w3schools.com/mysql/

Walingkas, H. L., & Saian, P. O. N. (2023). Penerapan framework Flask pada pembangunan sistem informasi pemasok barang. *Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi)*, *7*(2), 227–234. https://doi.org/10.35870/jtik.v7i2.729

Wang, P., Wang, Z., Chi, L., Ren, X., Wu, W., & Cheng, W. (2022). Research and application of the network security monitoring capability evaluation model of power control system based on AHP and fuzzy comprehensive evaluation. *Journal of Physics: Conference Series*, *2246*(1), 012046. https://doi.org/10.1088/1742-6596/2246/1/012046

Yuniar, Suganda, U. S., Alkahfi, M., & Suryadi, D. (2021). *Koleksi program database Python*.