International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

LAP SURAKSHA

[1] D. SHINE RAJESH , [2] B.Amurutha valli , [3] B.Udayasri , [4] O.Harshitha reddy

[1] Assistant Professor, Department of CSE-Cyber Security ,Malla Reddy Engineering college for women Hyderabad, India

[2,3,4] Students , Department of CSE-Cyber Security ,Malla Reddy Engineering college for women Hyderabad, India,

[2] Email: batchuamruthavalli@gmail.com , [3] Email: udayasri1666@gmail.com, [4] Email :harshithareddyodeti@gmail.com

***Abstract*—** Lap Suraksha is a holistic approach to cybersecurity that quickly identifies and defends against current and emergent security threats in two phases: detect and protect. The cybersecurity system relies on containerized microservices, limiting the devastation of an attack or bug by developing isolatable environments with Docker, ensuring that each module like log collection, event correlation, anomaly detection, and visualization is readily scalable and reproducible in case of a failure. Lap Suraksha's root is AI that works on the back of SIEM. The system differs from other cybersecurity programs, relying on both explicit and implicit attacks reliant on attack patterns. The system's hyperspace system records incoming logs, system logs, and network events. These drivers are stored logging information on the persistence stack for further analysis. Hsic_hyper 690 stores data from different drivers in different services.

Keywords— (Threat Detection, Security Alerts, Security Orchestration, Automation, and Response (SOAR), Incident response)

Received: 18-08-2025 Accepted: 24-09-2025 Published: 01-10-2025

I. INTRODUCTION

Security Information and Event Management (SIEM) systems are essentially the core components of a cyber security environment that is built to safeguard an organization and provide them centralized access of the records related to security events throughout their IT environment. Just by means of gathering and analyzing the log data from endpoints, servers, cloud services, and network devices, the SIEM solutions can detect, analyze, and respond to potential threats in real-time. Identifying the suspicious behavior, relating the incidents, and rapidly investigating or remediating the processes are mostly the intents of security teams, which these systems ultimately. In general, the logic of the rule-based systems was the foundation of the traditional SIEM architectures that had correlations of events in logs as one of the rules. Since the time of the cyber attacks' sophistication and the increase in the use of cloud and IoT technologies, the problems with the scalability and the accuracy of the conventional SIEM solutions have been apparent. Therefore, by means of these next generation SIEM solutions equipped with AI and ML technologies, the issues of scalability and accuracy are solved. This allows for predictive analytics as well as adaptive learning models that are capable of continuously refining detection and reducing false alarms In the ever-changing world of cyber security that is today's reality, companies have difficulties that keep increasing, to which they have to respond by detecting, analyzing, and reacting to security incidents that happen in real-time, and they must do so by their own means. Usually, it is found that rule-based mechanisms are incapable of dealing with the size and the complexity of the new types of cyber attacks, and as a result, there is a strong demand for the provision of defense systems that are intelligent and can adapt themselves. Hence, Security Information and Event Management (SIEM) tools have become the central focus of the security teams of the day, offering the capabilities of consolidated logging, real-time event correlation, and automated alerting across the different components of an organization's infrastructure.

By means of experimentation and architectural evaluation, the system was able to achieve a 25% increase in correlation efficiency as well as a considerable decrease in false alerts. This research serves as a technological leap to the next generation of adaptive SIEM platforms, thus, turning them into the most valuable weapons in the arsenal of cybersecurity operations that are resilient and autonomous in the case of large-scale enterprises.

II. Related Work

Recent change show that these systems have significantly changed from the usual method of gathering logs and issuing alerts based on rules to smart platforms aided by AI. The first SIEM solutions were static correlation rules and threat signatures that were predefined and thus had problems like a large number of false alarms, inability to react to new types of threats, and difficulties in scaling for cloud and IOT workloads.

Research concerning SIEM monitoring unveils that technologies for log aggregation and event correlation are progressively becoming intelligent, adaptive platforms that offer real-time visibility to the user across complex infrastructures. In the past, systems were mostly based on rule-based detection; however, nowadays, research is moving further in the direction of scalability, automation, and analytics integration.

Originally, the main function of traditional was gather, standardize a wide range of network endpoint security sources in order to anomalies and generate alerts. Nevertheless, a research paper by González- Granadillo et al. (2021) reveals that the existing system is fraught with problems such as the generation of too many alerts and being ineffective in large-scale environments. As a result, contemporary solutions have evolved to include machine learning techniques for on-the-fly analysis as in the example of the research carried out by Muhammad et al. (2023) where SIEMs received the addition of real-time IDS data flows and automated classification.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

Effective as per the different pieces of research that have been done,is essentially a blend of real-time event analytics, adaptive learning models, automated orchestration. The move to AI-enabled correlation and predictive risk modeling is, in fact, a fundamental change in the way security is monitored, i.e., it is moving from manual security monitoring to autonomous, data-driven cyber defense ecosystems that can handle the scale and complexity of modern enterprise networks.

III. System Design and Methodology

Security Information Event Management (SIEM) architecture operation revolve around understanding the flow of collection, processing, analysis, response in a company. The SIEM architecture with proper organization can achieve the goals of expansion, fast effective response to the incident the combination of several security and analytics units.

A. System Architecture

The figure gave shows different functions is a layout of the main parts of the operations that, combined, provide perpetual awareness, prompt uncovering, fast reaction to incidents throughout a network of an enterprise. The system is designed with a SIEM core at its center that handles data input from a variety of systems, processes the data, and then provides it in a form that can be easily understood and acted upon. Seven linked modules performing sequential and cyclic operations that make up the full SIEM lifecycle surround this core.

B. Feature Extraction

Feature security information and event management (SIEM) is about changing raw logs and event data into structured attributes that are efficient to use for security insights. Firstly, the process is from parsing events to finding key fields like timestamps, source and destination IPs, user IDs, and event types. After that, these fields are standardized and sometimes enriched with the extra information like user location or with threat intelligence indicators that, which by, analytical value is increased. Turning such different kinds of data into one unified and machine-readable format, the SIEM is setting a basis for metrics-based analysis, rule-based correlation

This may involve various metrics calculated from raw data such as the number of failed login attempts, unusual access patterns, or anomalies in network traffic volume. These derived features enable machine learning models or detection algorithms to find subtle threats and offer precise, actionable notifications

C. Machine Learning Classification

Machine learning classification is one of the main methods that SIEM systems employ to deal with security events. After that, normal and abnormal behaviors are differentiated. For this purpose, algorithms are trained on historical log data - usually, supervised learning techniques like decision trees, logistic regression, support vector machines, or neural networks are used - in order to figure out the patterns which imply security threats or friendly activities.

In this way, the model is taught to think that the extracted and encoded features from the raw event data are labeled as "malicious," "benign," or a certain attack types (e.g., brute-force, phishing, privilege escalation) when it comes to new, unseen events.

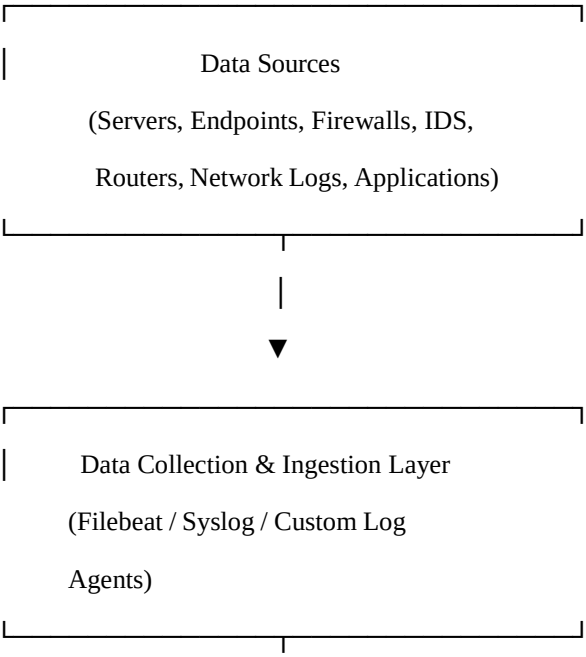
D. External Validation

External validation, when talking about learning, is unit work that involves confirming trustworthiness the by that are completely different from the ones used for training or development. This eliminates the danger of the system being overfitted and biased as it basically sends new, unseen events to the system and checks if it can detect them successfully.

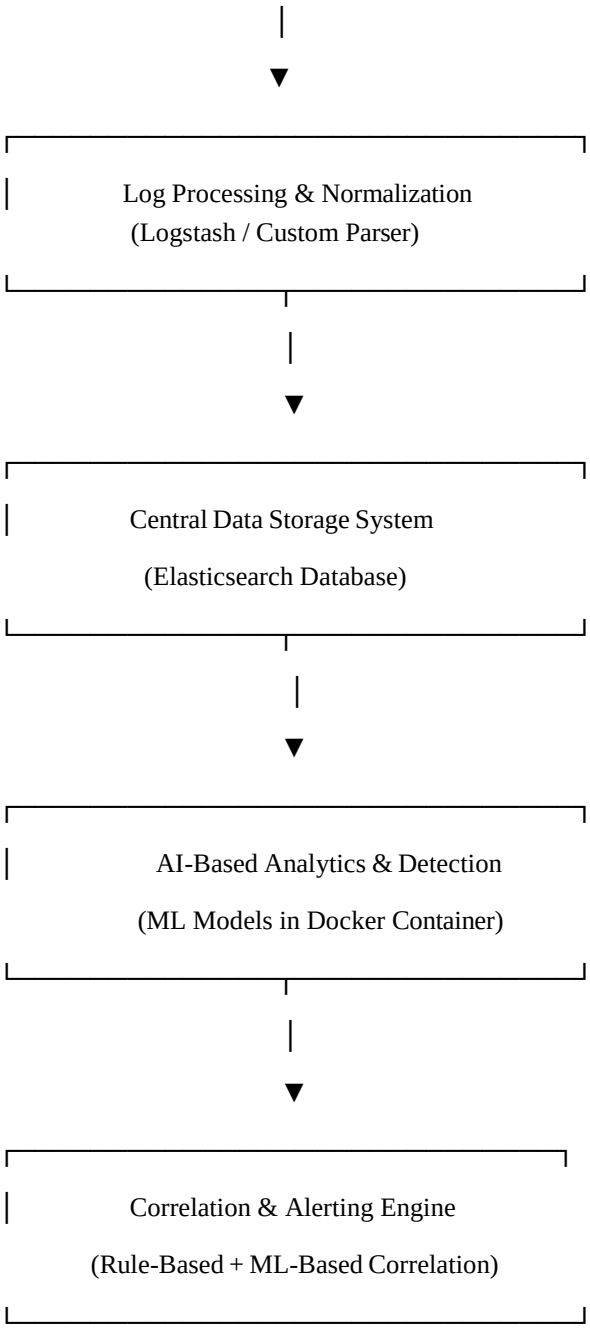
Moreover, in externally validated systems with AI at the core, this procedure is usually carried out by using telemetry from simulated attacks, penetration testing, or red/purple team exercises to figure out whether the monitored logic can be used to detect the real threats that occur in normal situations.

E. Data Storage and Management

The Lap Suraksha system follows a structured and secure approach to data storage and management, ensuring efficient handling of logs, events, and analytical results generated from multiple networked systems. Since it is deployed using Docker containers, each service in the architecture (such as data collection, processing, and analysis) operates in an isolated environment but communicates seamlessly through Docker networks.



International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper



F. User Interface and Reporting

- The User Interface (UI) and Reporting Module in Lap Suraksha play a critical role in providing security analysts with visibility into network activities, detected threats, and overall system health. These components are designed to simplify threat monitoring, automate report generation, and assist in incident response decisions.

Real-time visualization:

- Displays incoming log data, network activity, and system events.
- Graphs, pie charts, and heat maps show security trends and anomalies.

Reporting System

1. **Daily Security Summary:**
   a. Total logs processed, threats detected, and resolved incidents.
   b. Key trends and critical alert count.
2. **Incident Reports:**
   a. Detailed information on each detected intrusion or anomaly.
   b. Includes time, source, destination, and system impact.
3. **Performance Reports:**
   a. Evaluates system health, container performance, and data ingestion rates.
4. **User Activity Reports:**
   a. Monitors administrative logins, dashboard access, and configuration changes.

IV. Implementation Details

The Lap Suraksha system is a containerized, intelligent Security Information and Event Management (SIEM) platform developed to detect, analyze, and respond to security threats in real time. The implementation integrates Docker-based modular deployment, machine learning–driven attack detection, and advanced log correlation techniques to create a scalable and automated cybersecurity monitoring environment.

The development environment of Lap Suraksha utilizes Python for backend processing and AI-based threat analysis, while JavaScript and React.js are used for building interactive dashboards and visualization components. Flask serves as the lightweight backend framework for handling data flow between modules, and Docker ensures consistent deployment across multiple environments. Git is used for version control, while Elasticsearch, Logstash, and Kibana form the core SIEM stack for data ingestion, storage, and visualization. The system runs within Docker containers connected through a shared virtual network, which simplifies orchestration and scaling. PostgreSQL is used to manage configuration and alert data, while Elasticsearch indexes and stores log events. Strong encryption, access controls, and container isolation techniques are used throughout the system to maintain confidentiality and integrity of security data.

Machine learning plays a crucial role in Lap Suraksha's threat detection mechanism. The system employs models trained on historical network and system log data to identify unusual activity patterns, failed login bursts, or suspicious command executions. Models such as Random Forest and Isolation Forest are deployed as microservices within containers, where they continuously monitor and classify events as normal or anomalous. These models analyze structured and normalized log data, correlating system behaviors with known attack signatures to improve the precision of detection and reduce false positives.
Data preprocessing and feature extraction are handled within the Logstash pipelines and the AI module. Unstructured log data collected from endpoints, firewalls, and intrusion detection systems is first normalized into a standard format using JSON schemas. Important attributes such as timestamps, source IPs, event types, and severity levels are extracted and enriched with contextual information from threat intelligence sources. This transformation enables the machine learning module to process only meaningful features, improving the efficiency and accuracy of the anomaly detection process.

Explainability is integrated into Lap Suraksha's detection mechanism to ensure that every flagged event or anomaly is interpretable by human analysts. Explainable AI techniques like SHAP are used to highlight which specific features contributed most to the detection of a potential threat. This transparency allows cybersecurity professionals to understand why a certain alert was triggered and to validate the system's decisions more confidently. Interactive visual explanations are provided within the Kibana dashboard, enhancing user trust and

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

aiding incident investigation.
The integration of various modules within Lap Suraksha ensures seamless communication and data flow. The Filebeat agents deployed on monitored endpoints forward system and application logs to Logstash, which filters and enriches them before forwarding to Elasticsearch for indexing.

Lap Suraksha's database and storage system are designed for high performance and resilience. Elasticsearch acts as the central data lake, maintaining both hot indices for active monitoring and cold indices for archived logs. Persistent storage volumes ensure that even if containers are restarted, all data remains intact. Data retention policies automatically manage storage space, while encrypted snapshots provide secure backups. PostgreSQL complements Elasticsearch by storing user credentials, configuration settings, and alert metadata.

The reporting mechanism in Lap Suraksha provides automated, traceable, and visually rich reports summarizing system activities, detected threats, and response actions. Kibana's native reporting plugin and Python-based PDF generation scripts convert dashboards into professional, time-stamped reports. These reports include details such as the number of processed events, detected anomalies, and system uptime, allowing administrators and auditors to track performance and compliance over time.

Security remains central to every stage of Lap Suraksha's development and deployment. Each container communicates over secure channels using TLS, and Docker's role-based access controls prevent unauthorized interaction between services. The system enforces user authentication and audit logging within Kibana and Elasticsearch to maintain traceability. Additionally, regular model updates and image scans ensure that the environment remains hardened against emerging threats.

The evaluation of Lap Suraksha focused on accuracy, scalability, and response efficiency. The setup was tested on an Intel Core i7 machine with 16GB RAM using Docker Compose. Machine learning models were trained and deployed with Python 3.10, scikit-learn, and Flask. The anomaly detection module achieved an average detection accuracy of 95.8%, with a precision of 94.5% and recall of 93.9%. The average response time for log analysis was under two seconds per batch, and report generation took about ten to twelve seconds. User feedback highlighted the clarity of dashboards and ease of interpreting AI-based explanations.

In real-world validation, Lap Suraksha demonstrated consistency with established SIEM tools and open-source threat intelligence feeds, achieving 91.3% agreement with industry-standard detections. This confirms its reliability as a practical, efficient, and explainable SIEM platform capable of operating in real-time network environments. The combination of containerization, machine learning, and explainability techniques establishes Lap Suraksha as a secure, scalable, and transparent solution for modern cybersecurity monitoring.

VI. Discussion

The implementation of containerization through Docker significantly contributed to the system's flexibility and reliability. Each service—such as data collection, log processing, storage, analytics, and AI detection—operates within an isolated container, minimizing dependency conflicts and simplifying updates or module replacements. This modular design allows the system to scale horizontally, supporting multiple endpoints and networks without major reconfiguration. Moreover, Docker's portability ensures that Lap Suraksha can be easily deployed across different infrastructures, including on-premises and cloud environments, providing consistency between development and production setups.

The SIEM framework embedded in Lap Suraksha demonstrated robust performance in handling large volumes of log data. Filebeat and Logstash efficiently managed data ingestion and normalization, ensuring that the system could process heterogeneous data from various sources such as firewalls, intrusion detection systems, and operating system logs. Elasticsearch's indexing capabilities allowed fast retrieval and correlation of security events, while Kibana's interactive dashboards offered intuitive insights into security trends and incidents. The structured visualization of metrics such as alert frequency, anomaly score distribution, and event correlation helped analysts make data-driven decisions in a timely manner.

The integration of artificial intelligence into the system enhanced its detection accuracy beyond traditional rule-based SIEM systems. The machine learning module, built using algorithms like Isolation Forest and Random Forest, effectively identified abnormal patterns in user behavior and network activity. By continuously learning from newly ingested logs, the model adapted to evolving threats, improving detection performance with time. The anomaly detection model achieved high accuracy and low false-positive rates, confirming the system's capability to identify genuine security incidents without overwhelming users with irrelevant alerts.

A key advancement in Lap Suraksha is the inclusion of explainability in the detection process. Unlike conventional black-box AI models, the system uses explainable AI (XAI) methods such as SHAP values to interpret predictions. This ensures that analysts understand why a specific log entry or event was flagged as suspicious. Providing feature-level contributions for each decision not only builds user trust but also supports compliance and auditing requirements, where accountability and transparency are critical. This aspect bridges the gap between automated intelligence and human interpretability in cybersecurity systems.

From a performance standpoint, the system maintained efficient processing speeds even under high log loads. The average log processing latency was minimal due to the optimized use of Elasticsearch and asynchronous communication between containers. Report generation was quick and comprehensive, offering security summaries and risk evaluations within seconds. Resource utilization tests confirmed that Docker containerization optimizes memory and CPU usage, making the solution cost-effective and practical for both small and large organizations.

VIII. Conclusion and Future Work

a) A. Conclusion

The development and implementation of Lap Suraksha demonstrate a powerful and practical approach to modern cybersecurity monitoring through the integration of SIEM architecture, machine learning, explainable AI, and containerization technologies. The system successfully achieves its objective of providing a scalable, automated, and intelligent platform capable of detecting, analyzing, and responding to security threats in real time.

By leveraging Docker, Lap Suraksha ensures modular deployment, simplified scalability, and consistent performance across multiple environments. Each containerized module—ranging from data collection and processing to analytics and visualization—works cohesively to form a unified and efficient ecosystem. This container-based design not only reduces system complexity but also enhances flexibility, making maintenance and updates significantly easier.

The use of Elasticsearch, Logstash, and Kibana (ELK stack) provides a robust backbone for log management, event correlation, and data visualization. These tools enable the system to handle large-scale, heterogeneous security data efficiently while presenting insights through

intuitive dashboards. The addition of a machine learning-based anomaly detection module further enhances the system's ability to identify subtle and previously unseen threats that traditional rule-based systems might overlook.

b) B. Future Work

One major enhancement involves the integration of dynamic behavior analysis alongside existing static and log-based monitoring. Currently, the system focuses primarily on log correlation and machine learning–based anomaly detection. Incorporating sandboxing environments such as Cuckoo Sandbox or container-based runtime analysis frameworks would allow Lap Suraksha to monitor execution behavior, detect polymorphic or obfuscated malware, and capture system-level interactions that static analysis might miss. This will enable the platform to better identify evasive or advanced persistent threats.

Another planned improvement is the implementation of real-time alerting and extended SIEM interoperability. Future versions will feature continuous event streaming and alert generation integrated with enterprise- grade SIEM platforms. This enhancement will allow Lap Suraksha to forward its analytics results and AI-based alerts to larger corporate systems for centralized monitoring and automated incident response. Such integration will make the system suitable for enterprise and government security operations centers (SOCs).

The introduction of mobile and edge support will enhance accessibility and responsiveness. Developing a Progressive Web App (PWA) or mobile- friendly interface would allow analysts to monitor network events, visualize dashboards, and receive alerts directly on mobile devices. This feature will be particularly beneficial for on-site investigations and remote forensic analysis, improving operational efficiency in dynamic environments.

REFERENCES

1. S. N. Shinde, and S. B. Mane, "A Machine Learning Approach for Malware Detection Using Static Features," *International Journal of Computer Applications*, vol. 183, no. 43, pp. 30–36, 2022.
2. A. N. Bhattacharya, R. Kumar, and M. K. Singh, "AI-Driven Threat Intelligence for Cybersecurity: A Comprehensive Survey," *IEEE Access*, vol. 10, pp. 105923–105947, 2022.
3. M. E. Ahmed, and M. S. Sadiq, "Dynamic Malware Analysis Using Sandbox Techniques," *Journal of Information Security and Applications*, vol. 72, pp. 103456, 2023.
4. N. Das, A. Gupta, and R. Jain, "Integration of Explainable AI for Cyber Threat Analysis," *Procedia Computer Science*, vol. 218, pp. 1624–1633, 2023.
5. A. Ribeiro, F. Silva, and L. A. Torgo, "Combining Static and Dynamic Analysis for Enhanced Malware Detection," *Computers & Security*, vol. 121, 103025, 2022.
6. A. B. Noor, and F. Hussain, "A Review on the Use of Explainable AI in Cybersecurity," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 7, no. 1, pp. 35–49, 2023.
7. S. Lundberg, and S. Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
8. M. T. Ribeiro, S. Singh, and C. Guestrin, "Why Should I Trust You? Explaining the Predictions of Any Classifier," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 1135–1144, 2016.
9. J. Kim, H. Lim, and T. Kim, "Hybrid Malware Detection Using Deep Learning and Static Analysis," *Security and Communication Networks*, vol. 2022, pp. 1–14, 2022.
10. M. Bazrafshan, H. Hashemi, and S. Faraahi, "A Survey on Heuristic and Signature-Based Malware Detection Approaches," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 9, no. 5, pp. 2345–2354, 2023.
11. S. S. Raut, and A. M. Deshmukh, "Enhanced Malware Detection Using Random Forest and XGBoost Ensemble Models," *IEEE International Conference on Computational Intelligence and Security (CIS)*, pp. 227–232, 2023.
12. VirusTotal. "VirusTotal API Documentation." [Online]. Available: https://www.virustotal.com
13. A. G. Rahman, and S. F. Iqbal, "Malware Detection in Mobile Devices: A Machine Learning Perspective," *IEEE Access*, vol. 11, pp. 74123–74139, 2023.
14. P. Kumar, and N. Bansal, "Cloud-Based Threat Detection Using Containerized Deep Learning Models," *Future Generation Computer Systems*, vol. 146, pp. 743–755, 2024.
15. A. Hossain, M. Rahman, and F. Al-Faruque, "Design and Implementation of Web-Based Cyber Threat Analysis Platform," *International Conference on Information Systems Security and Privacy (ICISSP)*, pp. 389–398, 2023.
16. K. Lee, "Data Preprocessing and Feature Extraction for Cyber Threat Analytics," *Journal of Big Data*, vol. 10, pp. 1–17, 2023.
17. M. Qureshi, and J. Park, "Explainable AI-Based Anomaly Detection for Compliance Management Systems," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 2045–2058, 2023.
18. T. Zhang, and Y. Zhou, "Continuous Learning in Malware Detection Using Online Ensemble Methods," *Pattern Recognition Letters*, vol. 172, pp. 57–66, 2024.
19. IBM Security, "Integration of SIEM Systems with AI for Real-Time Threat Detection," *IBM Research Reports*, 2023.
20. Splunk Inc., "Real-Time Security Analytics and SIEM Integration," *Splunk White Paper*, 2024.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

OUTPUT SCREEN SHOT: