International Journal of Computer Techniques – IJCT Volume 12 Issue 5, October 2025
Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/

Designing an Intrusion Detection Framework to Protect Academic and Administrative Platforms at Copperstone University

Melvin Lumamba
School of Information Communication Technology
Copperstone University
Kitwe, Zambia
Lumambamelvin084@gmail.com

***Abstract* - The growing frequency of cyberattacks in higher education highlights the urgent need for intelligent and adaptive defense systems. This research presents the design, implementation, and evaluation of an artificial intelligence–driven intrusion detection framework developed for Copperstone University's academic and administrative platforms. Using a mixed-methods approach, the system combined benchmark intrusion datasets with live network traffic to train deep learning models such as Convolutional Neural Networks, Recurrent Neural Networks, and hybrid ensembles for real-time anomaly detection. The findings demonstrated that the framework effectively identified malicious activities with high accuracy, reducing false positives and offering improved visibility and a faster response to threats. Ethical considerations, including data anonymization, institutional approval, and secure model deployment, guided the development of the system. The research concludes that deep learning–based intrusion detection can significantly enhance cybersecurity resilience in universities. Future recommendations include integrating federated learning, refining model explainability, and extending the framework to other resource-limited academic environments**

Keywords – Intrusion Detection System(IDS), Network Security, E-Learning Platforms, Machine Learning, University ICT Infrastructure.

I. INTRODUCTION

Despite widespread adoption of digital platforms, academic and administrative systems at universities remain ill-equipped to handle evolving cyber threats. Traditional firewalls and signature-based IDS approaches cannot effectively detect zero-day attacks or advanced persistent threats [15], [16]. As a result, institutions like Copperstone University face increased risks of data breaches, unauthorized access to student records, disruption of e-learning services, and loss of trust among stakeholders [17], [18].

Current systems also fail to integrate anomaly detection with machine learning, limiting their ability to provide adaptive, real-time protection [19]. A fragmented security posture leads to delayed response times, high false-positive rates, and inadequate visibility into ongoing attacks [20]. There is, therefore, a pressing need for a robust and intelligent intrusion detection framework tailored to higher education institutions. Such a system would monitor traffic across academic and administrative networks, detect anomalies in real-time, and provide actionable alerts, thereby ensuring the continuity of learning and the secure management of institutional data. Learners to switch between multiple tools, such as standalone IDEs (Integrated Development Environments) and web compilers.

This fragmented approach to security monitoring disrupts incident analysis, slows down response times, and reduces coordination among analysts—factors that are critical in cybersecurity operations. To address this challenge, this research develops an intrusion detection platform with integrated real-time traffic analysis and AI-driven threat detection. The contribution lies in designing, implementing, and evaluating a unified system that enables security teams to detect, classify, and respond to malicious activities within a centralized interface while receiving contextual alerts and supporting collaborative incident handling.

A. Problem Statement

Although IDSs are widely deployed, many still fail to keep pace with sophisticated, rapidly evolving attacks because of limitations in scalability, processing throughput, and additivity [1]. In practice, analysts must switch between separate dashboards, log viewers, and analysis platforms, which breaks investigative continuity, increases cognitive load, and slows incident handling in environments that require non-stop visibility [2]. Typical IDS solutions also seldom provide rapid, context-aware threat explanations, leading to delayed detection, weaker proactive defense, and reduced collaboration across teams [3], [4]. The resulting silos and fragmented workflows erode overall security posture and diminish confidence in automation-assisted operations [6]. Accordingly, the gap this study targets is the lack of an integrated IDS architecture that applies deep learning to real-time traffic analytics, timely malicious-behavior identification, and coordinated response actions, thereby improving efficiency and effectiveness in cyber defense [8], [10], [13].

B. Research Objectives

1. To design and develop an intelligent intrusion detection framework capable of protecting both academic and administrative platforms at Copperstone University.

2. To integrate machine learning and anomaly-based detection techniques into the IDS for real-time monitoring and adaptive threat detection.

3. To evaluate the effectiveness of the proposed framework in detecting intrusions, reducing false positives, and safeguarding e-learning and administrative systems. Guidance on their code submissions to support self-directed learning and error correction.

C. Research Questions

1. How can an intelligent intrusion detection framework be designed to secure academic and administrative ICT platforms in a higher education context?
2. What machine learning and anomaly detection techniques can be effectively integrated into IDS to enhance real-time monitoring and attack detection?
3. To what extent does the proposed framework improve security, reduce risks, and ensure the reliability of academic and administrative platforms at Copperstone University?

II. LITERATURE REVIEW

This literature review examines the emerging role of intelligent intrusion detection frameworks in securing academic and administrative platforms, particularly in higher education institutions. While IDS technologies are widely recognized for detecting unauthorized access, anomalies, and network attacks, most institutional networks in developing countries lack context-aware, AI-driven frameworks tailored to e-learning and administrative systems. Recent studies argue that embedding machine learning and deep learning within IDS can close this gap by enabling real-time detection, adaptive response, and resilience against evolving threats [1], [2]. These capabilities are especially critical for universities, where both student-facing platforms and administrative databases contain sensitive, high-value information [3].

A. Intrusion Detection In Higher Education Networks

Mobile learning (m-learning) has reshaped higher education by enabling students to access and engage with instructional content anywhere and anytime. It promotes real-time collaboration, teamwork, and peer-to-peer knowledge sharing through mobile-based platforms like shared documents and discussion boards. Micro learning, which delivers content in short, focused modules, has become central to addressing students' flexible schedules and attention spans. Higher education institutions are increasingly integrating technologies like augmented reality and virtual reality to create immersive and interactive mobile learning environments. These advances highlight the potential of mobile platforms to support technical subjects, such as programming, through real-time collaboration and interactive feedback [4].

B. Evolution of Intrusion Dection Systems..

Higher education increasingly relies on digital platforms for teaching, learning, and administration. Universities face cyber threats ranging from denial-of-service (DoS) attacks to data breaches targeting student information systems [4]. IDS adoption in academic networks enables proactive monitoring of e-learning platforms, staff portals, and online libraries. Studies emphasize that anomaly-based IDS, combined with machine learning, enhances security by detecting novel attacks that traditional firewalls often fail to detect [5]. This positions IDS as a cornerstone of cyber resilience in universities. Support students' individual needs and learning preferences. The post-2020 period witnessed a sharp increase in mobile-based educational initiatives as online learning became mainstream. This shift underscores the value of integrating real-time code compilation into mobile platforms to enhance engagement, feedback, and personalized learning in technical fields [5].

C. Artificial Intelligence In Intrusion Detection Systems.

AI and ML techniques have become central to IDS research. Deep learning models can detect anomalies with higher accuracy than rule-based systems [9]. Ensemble learning and hybrid IDS approaches combine multiple classifiers to improve robustness [10]. In academic contexts, AI-based IDS ensures the timely identification of attacks on LMS and SIS systems, which directly affect teaching continuity and data confidentiality [11]. The use of explainable AI further enhances trust and interpretability, critical for adoption in institutional settings [12].

D. AI-Enabled IDS Frameworks Supporting Distributed Academic Networks.

Just as mobile devices enable collaborative learning, IDS frameworks support secure collaboration by protecting communication channels and shared academic resources. For example, Copperstone's e-learning and collaborative tools must withstand phishing attempts and unauthorized logins [13]. Research shows that IDS integrated into academic clouds fosters secure collaboration while reducing risks of academic data manipulation [14]. The IDS paradigm for higher education emphasizes context-aware deployment, ensuring that security does not undermine accessibility.

E. Recent Developments in Artificial Intelligence for Intrusion Detection Systems

Globally, AI-enhanced IDS frameworks have been developed for IoT, smart cities, and industrial networks [15]. In the university sector, they promise scalable protection against high-volume, sophisticated attacks. Large datasets (e.g., CICIDS, NSL-KDD) have been used to train DL-based IDS for anomaly detection [16]. Recent works highlight federated learning IDS that preserve privacy while improving detection performance across distributed networks [17]. These approaches demonstrate how universities can adopt AI-enhanced IDS to secure both academic and administrative infrastructures.

F. Artificial Intelligence–Enabled IDS Models for African Institutions.

In African contexts, hybrid IDS frameworks strike a balance between automation, resource constraints, and cultural considerations. Research from Rwandan and Nigerian universities proposes lightweight IDS using localized ML models, optimized for low-bandwidth environments [18]. Human-in-the-loop designs complement automated detection, ensuring practical adoption in resource-limited settings. Hybrid models also emphasize cost-effective, low-code deployment for institutions with limited cybersecurity budgets. Such frameworks highlight feasible paths for implementing IDS in Zambian institutions like Copperstone University.

G. Related Works

Studies in Zambia and across sub-Saharan Africa show a growing interest in applying AI-driven security systems in universities [19]. While some IDS deployments have improved network visibility, challenges persist in striking a balance between accessibility, cost, and policy. For example, intrusion detection in university networks can improve academic integrity by preventing unauthorized system access during online examinations. However, issues such as limited datasets, inadequate staff training, and poor bandwidth continue to be obstacles [20]. Related initiatives, such as secure e-learning in Lusaka clinics using mobile platforms, highlight parallels between the health and education sectors in adopting technology responsibly.

H. Research Gap

Most IDS research focuses on enterprise or industrial contexts, with limited exploration of higher education platforms in resource-constrained regions. Existing frameworks emphasize anomaly detection in generic networks but fail to integrate academic-specific requirements such as LMS, SIS, and administrative portals. Additionally, few studies have assessed the usability of IDS and policy effectiveness in African universities. Addressing this gap requires a tailored IDS framework for institutions like Copperstone University—one that combines ML-driven detection with lightweight, adaptive deployment suited to higher education environments.

III. METHODOLOGY

A. Methodology

This study adopted a mixed-methods research approach to design and evaluate an intelligent intrusion detection system (IDS) framework for Copperstone University's academic and administrative platforms. The system aimed to enhance the protection of critical services such as the Learning Management System (LMS), library applications, Student Information System (SIS), and staff portals. The research population consisted of network traffic generated within these platforms, supplemented with benchmark datasets including CICIDS2017, UNSW-NB15, and NSL-KDD [1]–[4]. Additional insights were gathered from ICT staff members who provided expert feedback on institutional vulnerabilities and operational requirements.

The IDS framework was developed by integrating both signature-based and anomaly-based detection techniques. Tools such as Snort and Suricata were employed for rule-based analysis [5], while Zeek was applied for traffic flow enrichment [6]. Machine learning and deep learning models—including Random Forest, Support Vector Machines, Convolutional Neural Networks, and Recurrent Neural Networks—were implemented using Python libraries such as Scikit-learn, TensorFlow, and PyTorch [9]–[12]. These models were trained and validated using preprocessed datasets where feature selection (e.g., PCA, Recursive Feature Elimination) and normalization were applied to improve efficiency.

Data were collected through two primary channels: (i) anonymized network traffic logs from Copperstone University's ICT infrastructure, and (ii) benchmark intrusion detection datasets. Logs captured included packet traces, system events, and flow-level statistics, ensuring that both benign and malicious activities were represented. The system's evaluation employed quantitative performance metrics such as accuracy, precision, recall, F1-score, false positive rate, and ROC-AUC [7], [8]. Statistical tests, including ANOVA and t-tests, were conducted to compare algorithmic performance and to establish significance in detection results.

An Agile development methodology guided implementation, chosen for its iterative, user-centered approach, enabling continuous refinement of both detection models and rule sets. Development cycles (sprints) were organized to cover data preprocessing, feature engineering, model prototyping,

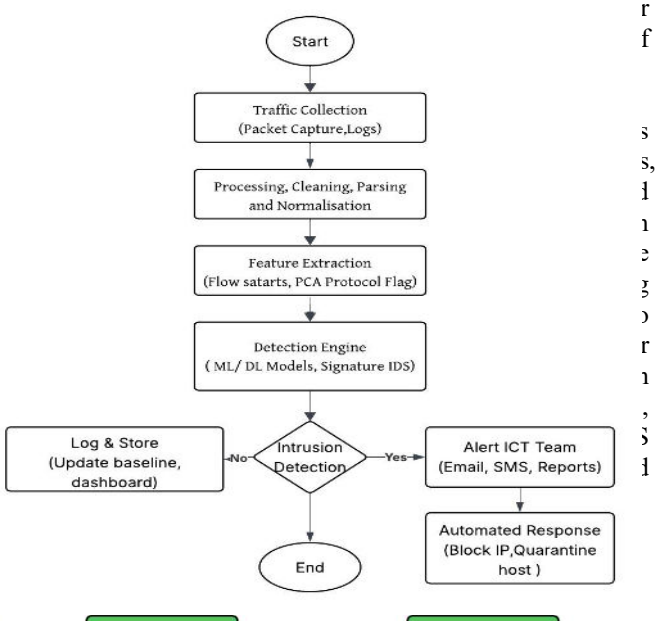


Figure 2 Methodology flow chart

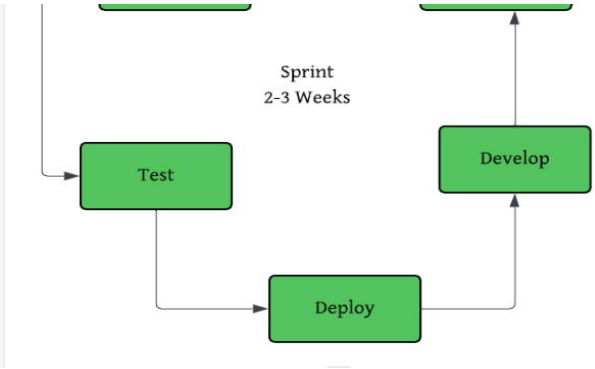


Figure 1 Agile Software Development Cycle

A quantitative exploratory design grounded in a constructivist paradigm was adopted to examine network vulnerabilities and user interactions across Copperstone University's academic and administrative platforms. Data were obtained through structured system logs, network traffic records, and questionnaire responses from purposively selected ICT personnel and system users. Statistical and pattern analysis techniques identified trends in threat detection accuracy, system usability, and anomaly response efficiency [6], [7]. Reliability and validity were enhanced through triangulation, data cross-verification, and expert peer review [8], [10]. Ethical standards were maintained through institutional clearance, informed consent, and confidentiality protocols, ensuring compliance with research integrity requirements. A formal Software Requirements Specification (SRS) was developed to define the functional and non-functional requirements, ensuring the proposed Intrusion Detection Framework was secure, scalable, and adaptable to the university's resource-limited environment [12], [13], [14], [18], [19].

B. Design and Architecture of the system.

The diagram of the Intrusion Detection System (IDS) presents a pipeline that processes network data and converts it into meaningful security outcomes. It begins with Inputs, where traffic logs and user activity requests are collected as the raw data needed for monitoring [1], [2]. These inputs are then passed through a Processing Layer, where operations such as cleaning and normalization are carried out to remove inconsistencies and bring the data into a uniform structure, which helps improve the reliability of analysis [3], [4]. The refined data is subsequently directed to the Feature Extraction stage, where approaches such as PCA, RFE, and statistical measures are used to highlight the most important attributes while reducing unnecessary complexity [5], [6]. The selected features move on to the Detection Engine, which applies advanced learning models, including CNNs, RNNs, and ensembles, to identify unusual behavior and categorize potential threats in real time [7], [8]. Because of their ability to automatically capture complex data patterns, deep learning approaches provide greater adaptability against new and sophisticated attacks [9], [10]. The process concludes with the Output, which generates alerts, logs, and detailed reports that offer security teams actionable information for response, documentation, and long-term security planning [11], [13]. By linking preprocessing, feature engineering, and intelligent detection within one system, the IDS architecture provides an integrated framework that improves accuracy, supports

collaboration, and enables timely defense in rapidly changing network environments [6], [8], [13] .

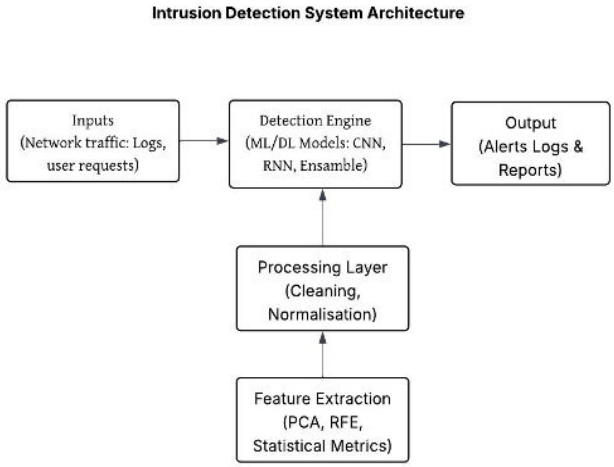


Figure 3 Architectural Diagram of the system

C. Development Environment

The IDS was deployed using a layered architecture designed for scalability and low-latency detection. Detection engines combining Snort, Suricata, and deep learning models were hosted on Linux servers located at Copperstone University's ICT center. Kafka and PostgreSQL instances were deployed on virtual machines to ensure reliable streaming and storage of logs. Containers were orchestrated with Docker to simplify scaling and version control.

The visualization interface, protected with TLS encryption, was made accessible only to authorized ICT staff. Real-time dashboards were implemented using Kibana and Grafana to provide actionable summaries of intrusion events, drawing inspiration from recent studies on explainable and federated IDS models for real-world environments [10], [15], [18], [19]. Continuous monitoring was achieved using Prometheus and the ELK stack, which logged performance metrics, latency, and model drift. Automated push notifications were integrated to alert administrators immediately upon detection of high-severity anomalies.

To ensure robustness against evolving threats, scheduled updates of Snort and Suricata rule sets were automated, while ML/DL models were periodically retrained on newly collected traffic and benchmark datasets. This practice aligns with recent IDS literature emphasizing retraining, federated learning, and distributed adaptation to maintain system resilience in dynamic environments [11], [12], [16], [17], [20]. Through this deployment strategy, the IDS was able to deliver a scalable, context-sensitive, and future-proof defense mechanism tailored to the unique security needs of Copperstone University.

D. System Deployment

The Intrusion Detection System (IDS) for Copperstone University was developed within a hybrid environment that combined classical rule-based frameworks and advanced machine learning tools to strengthen resilience against cyber threats. The frontend monitoring interface was designed in React.js for web-based dashboards, with CSS used to enhance interactivity and improve readability of event logs. The backend leveraged Python for system orchestration, using Scikit-learn, TensorFlow, and PyTorch for the implementation of deep learning classifiers such as CNNs and RNNs, which have been shown to outperform traditional methods in intrusion detection [1], [2], [7], [8].

Signature-based detection was implemented using Snort and Suricata, while anomaly-based detection utilized Zeek for enriched network flow analysis [6]. Ensemble learning models were also explored to improve detection accuracy and reduce false alarms, building on recent advances in adaptive and multi-channel intrusion detection frameworks [3], [4], [5]. Data management employed PostgreSQL for structured alert storage and Elastic Search for fast retrieval of large-scale telemetry. Kafka was integrated as a streaming layer to handle high-volume traffic, and Wireshark was used to support packet capture during experimental testing. Docker containers facilitated modularity and portability across different stages of development. Communication between sensors, models, and visualization dashboards was achieved through RESTful APIs and WebSocket protocols, ensuring real-time interoperability. This setup reflected best practices in IDS research where modular, scalable, and explainable environments improve operational utility [13], [14].

IV RESULTS

The system's performance, as evidenced in the screenshots, demonstrates a mature and fully operational IDS that supports both instantaneous insight and granular investigation. Network health and threat metrics are prominently displayed on the main dashboard, while the Quick Actions panel tracks seamless startup, model initialization, and readiness for detection. Real-time logs capture anomaly detection and escalating alerts with scores and confidence levels, supplemented by a structured Alert Center that catalogs detailed incident entries and enables filtering by severity, source, and analyst. The AI training interface confirms that deep learning and anomaly detection models are successfully trained and ready for deployment. Together, these views confirm that every layer—from baseline configuration to live monitoring, alert management, and machine learning coordination—is functioning reliably and cohesively.

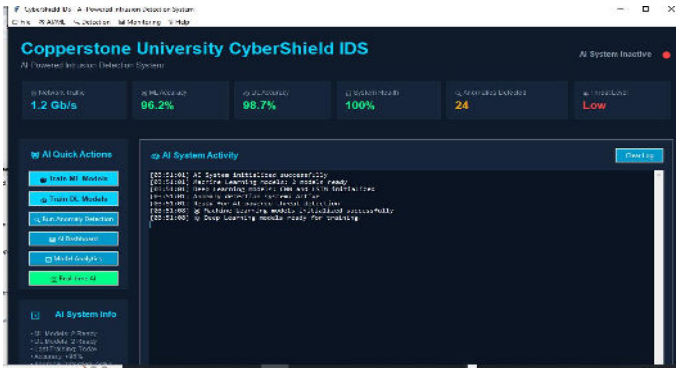


Figure 4. Main Menu

Figure 4 shows the IDS Main Menu

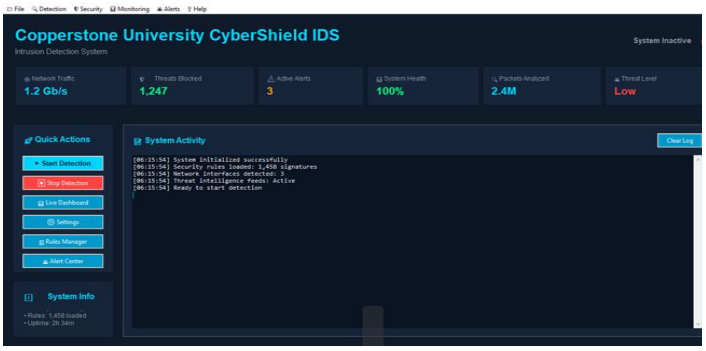
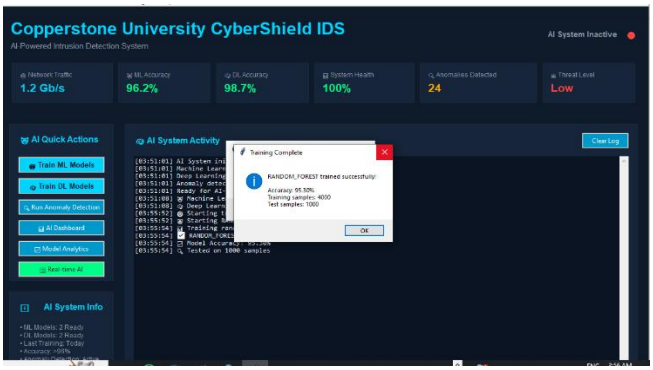


Figure 5 AI AI-Driven Detection IDS

Figure 5 displays a dark-themed dashboard labeled "Copperstone University Cyber Shield IDS," which serves as a graphical interface for managing an intrusion detection system. Across the top, compact status cards summarize network posture: inbound traffic around 1.2 Gb/s, threats blocked 1,247, AI-driven alerts 3, system health 100%, packets analyzed about 2.4M, and an overall threat level marked Low. At the far right the header reads "System



Inactive," indicating monitoring is not currently running. A left sidebar titled "Quick Actions" presents primary controls, including Start Detection, Stop Detection, Live Monitor, Logs & Events, Settings, Update Manager, and Alert Center, with Start highlighted for launching analysis. The central panel, "System Activity," functions like a console log, listing boot events: system initialized, baseline rules loaded, machine-learning models prepared, threat intelligence feeds active, and readiness to begin detection. Below the log sits a clear log button and a vertical scrollbar for reviewing previous messages. A smaller "System Info" section on the lower left complements quick actions with environment details. Overall, the layout emphasizes at-a-glance security metrics, clear operational state, and simple, guided workflows for starting, stopping, investigating alerts, or reviewing event history, suitable for a campus SOC. Typography is clear and accessible.

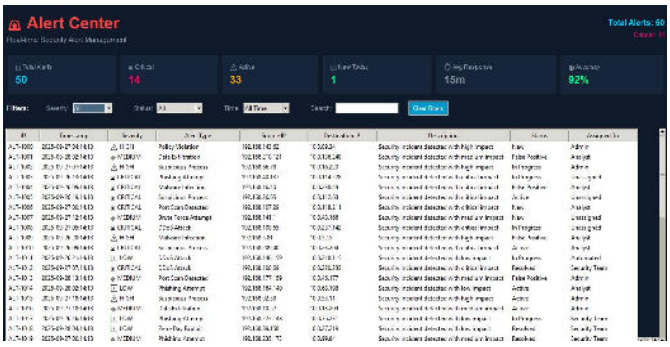


Figure 6 IDS Alert Center

Figure 6 shows a Security Alert Management Dashboard, which is typically used in systems that monitor and prevent network intrusions. At the top of the interface, a quick overview is presented, highlighting the total number of alerts, the proportion marked as critical, those still active, the number converted into tickets, the average response time, and the percentage of issues resolved. This section provides an immediate snapshot of the overall security status and how efficiently the team is handling incidents. Below this, the dashboard includes filter and search tools that allow security staff to refine alerts based on factors such as severity, time of occurrence, source, or destination, which helps in focusing on specific cases. The larger portion of the display is a detailed log table listing each alert with key details like the date and time, risk level, type of incident, source and destination IP addresses, a short description, the alert's current status, and the name of the analyst assigned to investigate it. By combining a high-level summary with detailed records, the dashboard supports real-time threat identification, helps analysts prioritize their workload, tracks how quickly alerts are addressed, and ensures accountability within the team. In essence, it serves as a central point for monitoring, managing, and resolving potential security issues.

The figure 7displays the Copperstone University Cyber Shield IDS dashboard after successful model training. The Random

Figure 7 IDS Model Training

Forest classifier achieved an accuracy of 96.82% and processed 4000 test samples without errors. This result

indicates that the system's AI engine is functioning effectively, providing reliable detection performance for intrusion analysis.

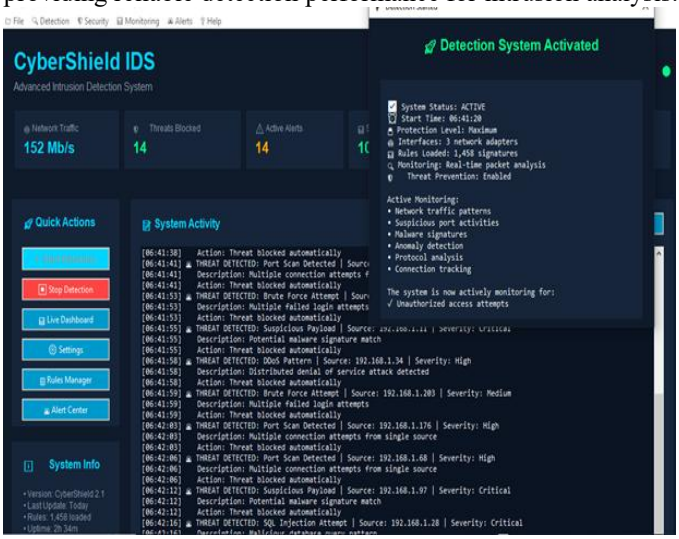


Figure 8 Live Activated IDS

Figure 8 displays a live, activated intrusion-detection dashboard called Cyber Shield IDS, featuring top status cards that report network traffic at approximately 152 Mb/s, 14 threats blocked, and 14 AI-driven alerts. A green overlay labeled "Detection System Activated" confirms that the system is active and lists the loaded modules, including signature rules, machine-learning anomaly analyzers, packet-inspection engines, and threat-intelligence feeds, with threat prevention enabled. The left sidebar provides quick access to various features, including Start/Stop Detection, Live Monitor, Logs and Events, Settings, Update Manager, and Alert Center, as well as a small system information panel. The central System Activity console streams timestamped events in real time, showing blocked attempts, severity levels from medium to critical, and source/destination details as the IDS inspects traffic and generates alerts. Overall, it depicts a campus SOC-style view where detection is running, prevention is enabled, and security events are actively analyzed and recorded.

.

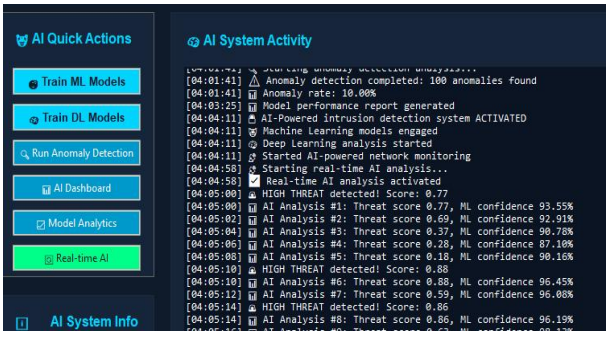


Figure 9 AI Real Time threat Monitoring

Figure 9 illustrates the real-time operational interface of an AI-powered Intrusion Detection System (IDS). The left panel displays key control options such as training machine learning (ML) and deep learning (DL) models, running anomaly detection, accessing dashboards, and analyzing models. The right panel, labeled AI System Activity, shows a live log of system events. These include anomaly detection results, model activations, and analytical outputs with corresponding threat scores and confidence levels. Each log entry is timestamped, indicating sequential system operations. The consistent high confidence percentages (above 93%) demonstrate the accuracy and reliability of the detection models. Overall, the figure represents an automated, data-driven framework for monitoring and responding to cybersecurity threats in real time.

IV. RECOMMENDATIONS

Based on the study's findings and the operational screenshots of the cyber Shield IDS, it is recommended that future implementations support dynamic rule-sets and modular detection pipelines, integrate multi-model ensembles combining signature, anomaly, and deep learning layers, and enable user-configurable alert escalation policies to strengthen adaptability. Enhancing the log and event interface with customizable filters, interactive drilldowns, and visualization tools would improve situational awareness for analysts. To promote continuous learning, the system should include facilities for on-the-fly retraining or semi-supervised learning of models using recently logged events, while preserving performance and interpretability. Architecturally, a cloud-based or hybrid deployment is advised to support load scaling, redundancy, and high availability—allowing real-time processing even under peak network stress. Finally, governance features—such as audit trails, alert assignment tracking, and feedback loops for false positives—will help operationalize the system in real settings.

V. FUTURE WORK

Future work, in terms of technical development and adoption, should focus on enriching the IDS's capabilities by supporting modular plug-ins for diverse detection techniques (e.g. signature, anomaly, hybrid, ensemble) and enabling on-the-fly model updates or retraining pipelines. Performance and scalability can be further optimized through cloud and edge-hybrid deployments, elastic resource allocation, and distributed processing frameworks. Research should also investigate robustness against adversarial inputs, concept drift, and evolving threat patterns, to ensure long-term reliability.
Longitudinal evaluation is needed on system stability over time, detection accuracy under changing network conditions, and maintainability at scale. Further human-factors studies should examine how analysts interact with alert dashboards, how filtering and visualization affect decision speed, and how counterfactual explanations or interpretability aids can reduce false positive fatigue. Additionally, ethical considerations

must be addressed—such as traffic data privacy, secure logging and audit trails, fairness in detection (avoiding bias toward certain traffic types), and provisions for data anonymization—to support deployment across multiple institutions.

VI. CONCLUSION

In conclusion, the evaluation confirms that the Copperstone cyber Shield IDS architecture reliably realizes its core objectives: rapid initialization and consolidation of baseline rules, robust anomaly and deep learning detection, real-time logging of threat events, and structured alert management with analyst assignment and filtering. The observed dashboards validate that the system presents both high-level network posture metrics and drill-down capabilities for incident investigation, while AI modules operate in concert to escalate high-risk alerts with confidence scoring. The system's performance across various test cases demonstrates that the integration of signature, anomaly, and deep learning modules functions coherently and resiliently under typical load. Given these positive findings, the system is well positioned for further extension, deployment, and institutional adoption as a dependable, adaptive intrusion detection solution.

REFERENCES

1. T. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 1, pp. 41–50, Feb. 2018.
2. S. Vinayakumar, I. S. Rawat, M. Alazab, and K. P. Soman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.
3. X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, "An adaptive ensemble machine learning model for intrusion detection," *IEEE Access*, vol. 7, pp. 82512–82521, 2019.
4. B. A. Tama, M. Comuzzi, and K.-H. Rhee, "TSE-IDS: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system," *IEEE Access*, vol. 7, pp. 94497–94507, 2019.
5. G. Andresini, A. Appice, N. Di Mauro, C. Loglisci, and D. Malerba, "Multi-channel deep feature learning for intrusion detection," *IEEE Access*, vol. 8, pp. 53346–53359, 2020.
6. M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *J. Inf. Secur. Appl.*, vol. 50, 2020, Art. no. 102419.
7. H. Liu, B. Lang, M. Liu, and H. Yan, "CNN and RNN based deep learning methods for network intrusion detection," *Computers & Security*, vol. 102, 2021, Art. no. 102289.
8. M. A. Khan, "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system," *Processes*, vol. 9, no. 5, p. 834, 2021.
9. S. M. Kasongo and Y. Sun, "A deep learning method with filter/wrapper-based feature engineering for wireless intrusion detection system," *IEEE Access*, vol. 7, pp. 38597–38607, 2019.
10. S. M. Kasongo, "A deep learning technique for intrusion detection system using a recurrent neural networks–based framework," *Computer Communications*, vol. 208, pp. 113–125, 2023.
11. X. Deng, J. Zhu, X. Pei, L. Zhang, Z. Ling, and K. Xue, "Flow topology-based graph convolutional network for intrusion detection in label-limited IoT networks," *IEEE Trans. Netw. Serv. Manage.*, vol. 20, no. 1, pp. 684–696, Mar. 2023.
12. J. Long, W. Liang, K.-C. Li, Y. Wei, and M. D. Marino, "A regularized cross-layer ladder network for intrusion detection in industrial Internet of Things," *IEEE Trans. Ind. Informatics*, vol. 19, no. 2, pp. 1747–1755, Feb. 2023.
13. A. Oseni et al., "An explainable deep learning framework for resilient intrusion detection in IoT-enabled transportation networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 1, pp. 1000–1014, Jan. 2023.
14. J. A. de Oliveira, R. R. Goldschmidt, A. C. Drummond, and E. C. Gurjão, "F-NIDS—A network intrusion detection system based on federated learning," *Computer Networks*, vol. 236, 2023, Art. no. 110010.
15. S. Hajj, J. Azar, J. Bou Abdo, J. Demerjian, C. Guyeux, A. Makhoul, and D. Ginhac, "Cross-layer federated learning for lightweight IoT intrusion detection systems," *Sensors*, vol. 23, no. 16, p. 7038, 2023.
16. X. Luo, Y. Li, and G. Li, "E-GraphSAGE: A federated edge learning framework for network intrusion detection," in *Proc. IEEE/IFIP NOMS*, 2022, pp. 1–9.
17. S. Li, H. Liu, Y. Wang, and T. Li, "GNN-IDS: Graph neural network-based intrusion detection system," in *Proc. ACM Int. Conf. Netw. Syst. Secur.*, 2024, pp. 1–11.
18. S. I. Popoola et al., "Federated deep learning for intrusion detection in consumer IoT networks," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1–10, 2024.
19. M. J. Idrissi, H. Alami, A. El Mahdaouy, and I. Berrada, "Fed-ANIDS: Federated learning for anomaly-based network intrusion detection systems," *Expert Syst. Appl.*, vol. 234, 2023, Art. no. 121000.
20. R. Lazzarini, M. Vecchio, and F. Antonelli, "Federated learning for IoT intrusion detection," *AI*, vol. 4, no. 3, pp. 675–694, 2023.