IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 1, January 2026

DOI: 10.17148/IJARCCE.2026.15142

Intrusion Prevention using Machine Learning with Advanced Data Protection and Real Time Threat Analysis

Sivakrishna P A[1], Abhinav A[2], Saindav C Das[3] and Prof. Marina Glastin[4]

Undergraduate Research Paper, Department of Computer Science,

College of Engineering Kottarakkara, Kollam, Kerala, India[1,2,3]

Assistant Professor, Department of Computer Science, College of Engineering Kottarakkara, Kerala, India[4]

Abstract: In today's increasingly interconnected digital environment, protecting information systems from evolving cyber threats has become a critical necessity. Traditional intrusion detection and prevention systems often rely on static, rule-based approaches, which are insufficient to identify sophisticated and previously unseen attacks. This paper presents an Intelligent Intrusion Prevention System (IPS) that leverages machine learning techniques for advanced data protection and real-time threat analysis. The proposed system is deployed within a self-hosted Linux-based private cloud environment, created by converting a standard laptop into a secure server infrastructure, ensuring data sovereignty and administrative control. Machine learning models such as Random Forest, Support Vector Machine (SVM), and Deep Neural Networks (DNN) are employed to analyse network traffic and system logs, enabling accurate classification and prediction of intrusion activities. To enhance security, the system integrates encryption mechanisms, access control policies, and automated firewall actions, providing a multi-layered defence framework. A real-time monitoring dashboard visualizes intrusion attempts, system performance, and threat metrics, allowing prompt response and mitigation. The results demonstrate that the proposed system delivers an adaptive, scalable, and cost-effective cybersecurity solution capable of autonomous threat detection and prevention while preserving data privacy. This approach offers a practical and efficient framework suitable for academic, research, and small organizational environments.

Keywords: IPS, DNN, Real-Time Threat Analysis, Cybersecurity, Anomaly Detection.

I. INTRODUCTION

The rapid growth of digital connectivity, cloud computing, and data-driven applications has significantly increased the vulnerability of information systems to cyber threats. Organizations today face a wide range of attacks such as malware intrusions, distributed denial-of-service (DDoS), phishing, and unauthorized access. Traditional Intrusion Detection and Prevention Systems (IDPS), which primarily rely on static signatures and predefined rules, are often ineffective against novel and evolving attack patterns. This limitation highlights the need for intelligent, adaptive, and data-driven security mechanisms capable of real-time threat detection and prevention.

Machine Learning (ML) has proven to be an effective solution for addressing these challenges by enabling systems to learn complex patterns from historical and real-time network data. In this work, an Intelligent Intrusion Prevention System (IPS) is proposed that integrates ML-based intrusion detection with advanced data protection techniques within a self-hosted Linux private cloud environment. The system utilizes supervised learning algorithms, namely Random Forest (RF), Support Vector Machine (SVM), and Deep Neural Networks (DNN), to classify network traffic as normal or malicious based on extracted features such as IP addresses, ports, protocols, packet size, and flow behaviour.

The proposed system was implemented on a Linux-based private cloud server created by converting a standard laptop into a secure server environment. Experimental evaluation was conducted using benchmark intrusion datasets along with real-time network traffic captured from a local network. The results indicate that the ML-based IPS achieves high detection accuracy and reliable real-time performance. Specifically, the Random Forest model achieved an accuracy of approximately 96–97%, with a low false-positive rate, making it suitable for fast real-time detection. The SVM model demonstrated stable classification performance with an accuracy of around 94–95%, while the Deep Neural Network achieved the highest detection capability for complex intrusion patterns, reaching an accuracy of approximately 97–98%.

Real-time testing further showed that the system could detect and respond to intrusion attempts with an average response latency of less than one second, enabling prompt prevention actions such as IP blocking and firewall rule enforcement.

The integration of AES-based encryption, secure communication protocols, and role-based access control ensured data confidentiality and integrity throughout the monitoring process. Additionally, the real-time dashboard successfully visualized attack statistics, system status, and threat trends, assisting administrators in timely decision-making.

These experimental results demonstrate that the proposed ML-driven IPS provides an effective, scalable, and cost-efficient cybersecurity solution. By combining intelligent threat detection with a self-hosted Linux cloud infrastructure, the system offers enhanced data privacy, adaptive learning capabilities, and real-time protection against modern cyber threats, making it suitable for academic, research, and small organizational deployments.

II. LITERATURE SURVEY

Intrusion Detection and Prevention Systems (IDPS) have undergone substantial advancements with the integration of machine learning and artificial intelligence techniques to counter modern cyber threats. Recent studies have explored data-driven approaches to enhance detection accuracy, reduce false alarms, and improve real-time response capabilities. Researchers have addressed key challenges such as evolving attack patterns, scalability in high-traffic networks, encrypted data analysis, and efficient deployment in resource-constrained environments. This section presents a review of significant contributions that form the foundation for intelligent, adaptive intrusion prevention frameworks

Celestin and Vanitha [1] analysed security and privacy risks in IoT systems, highlighting excessive data collection, weak authentication, outdated firmware, and insufficient encryption. Their findings show that many devices favour functionality over security, enabling unauthorized access and covert surveillance.

Sittig and Singh [2] presented a socio-technical framework for mitigating ransomware attacks in healthcare systems. The study emphasizes system configuration, user training, continuous monitoring, and rapid recovery, highlighting shared responsibility between technology, users, and organizational policies.

Koller et al. [3] proposed Root sense, a real-time host-based intrusion prevention system that integrates exploit-based and anomaly-based detection. By correlating events across multiple system subsystems, the approach improves detection accuracy while allowing tuneable performance overhead.

Margale et al. [4] proposed a machine learning–based intrusion detection system using SVM, Random Forest, XGBoost, and Decision Tree classifiers. Their dual-panel architecture achieved 92–96% accuracy, improving detection reliability and reducing false positives in dynamic network environments.

Jadhav and Nichat [5] proposed a hybrid ML-based intrusion detection and prevention system combining real-time signature analysis in Golang with anomaly detection in Python. By integrating Random Forest, SVM, and GAN-generated data, the system improves detection accuracy and reduces false alarms in networks IT.

Scaife et al. [6] introduced CryptoDrop, a data-centric ransomware detection system that monitors user file transformations instead of program behaviour. By analysing entropy changes, file similarity, and type modifications, the system achieved early detection with minimal data loss and low false positives.

Nathan et al. [7] proposed a manifold-based framework for unsupervised anomaly detection in high-dimensional data. By distinguishing on-manifold and off-manifold anomalies after dimensionality reduction, the approach combines complementary detection methods to significantly improve recall while maintaining precision, achieving up to 16% recall improvement on benchmark datasets such as MNIST.

Qadeer et al. [8] presented a Linux-based packet sniffer using libpcap for network traffic analysis and intrusion detection. By operating NICs in promiscuous mode and decoding protocol headers, the system enables real-time monitoring and detection of suspicious network activities.

Lazim and Ali [9] examined security challenges in IIoT-based smart metering networks and proposed an ML-driven intrusion detection and prevention system. Their study shows that integrating anomaly-based ML techniques improves detection accuracy while maintaining efficiency on resource-constrained edge devices.

III. PROPOSED METHEDOLOGY

A. System Setup and Private Cloud Configuration:
A standard laptop is converted into a secure Linux-based private cloud server. The Linux operating system is configured with essential services and security tools, including Python, machine learning libraries (Scikit-learn, TensorFlow), web

DOI: 10.17148/IJARCCE.2026.15142

frameworks (Flask/Django), database systems, and firewall utilities. Secure network access, SSH authentication, and encrypted communication channels are established.

B. Data Acquisition:
Network traffic data is collected from both benchmark intrusion datasets (such as NSL-KDD and CICIDS2017) and real-time packet captures from the local network. Packet sniffing tools operate in real time to gather network flows, system logs, and access events. All collected data is securely stored within the private cloud infrastructure.

C. Data Preprocessing and Feature Extraction:
The raw network data is cleaned and normalized to remove noise and inconsistencies. Feature extraction is performed to obtain relevant attributes such as source and destination IP addresses, port numbers, protocol types, packet size, and flow duration. Categorical features are encoded, and the dataset is divided into training and testing subsets for supervised learning.

D. Machine Learning Model Training:
Supervised machine learning algorithms—Random Forest, Support Vector Machine (SVM), and Deep Neural Networks (DNN)—are trained using labelled intrusion datasets. Random Forest provides fast and accurate baseline classification, SVM ensures effective binary and multiclass detection, and DNN captures complex attack patterns. Model performance is evaluated using accuracy, precision, recall, and false-positive rate metrics.

E. Real-Time Threat Detection and Analysis:
The best-performing trained model is deployed on the Linux cloud server. Incoming network traffic and system logs are continuously monitored and analysed in real time. The ML engine classifies activities as normal or malicious and identifies intrusion attempts instantly.

F. Intrusion Prevention and Data Protection:
Upon detecting a threat, the system automatically initiates preventive actions such as blocking malicious IP addresses, enforcing firewall rules, and isolating affected components. Advanced data protection mechanisms, including AES encryption, SSL/TLS secure communication, and role-based access control, are applied to protect sensitive data and logs.

G. Visualization, Alerting, and Optimization:
A real-time web-based dashboard visualizes intrusion alerts, attack statistics, system health, and prevention actions. Detected events are logged securely for auditing and future analysis. Continuous testing and optimization are performed using simulated and real-world attack scenarios to reduce false positives and improve detection speed.

H. Testing and Optimization:
The proposed system is validated through controlled penetration testing and simulated cyberattack scenarios to evaluate its effectiveness under realistic operating conditions. Various attack types are introduced to assess the system's ability to detect intrusions accurately while maintaining continuous operation. This testing phase ensures that the intrusion prevention framework performs reliably in dynamic and unpredictable network environments.

Key performance metrics, including detection accuracy, response time, system reliability, and false-positive rate, are systematically analyzed. Based on the evaluation results, model parameters, classification thresholds, and decision rules are optimized to reduce false alerts and improve detection speed. This optimization enhances the overall efficiency and reliability of the intrusion prevention system.

I. Hardware Requirements:
- Processor: Intel Core i5 / i7 or equivalent
- RAM: Minimum 8 GB (16 GB recommended for optimal performance)
- Storage: 500 GB HDD or SSD
- Network Adapter: Gigabit Ethernet
- Display: 1920 × 1080 resolution
- Peripheral Devices**:** Keyboard, mouse, and optional Wi-Fi module

J. Software Requirements
- Operating System: Ubuntu / Debian (Linux Server)
- Programming Language: Python 3.12
- Frameworks & Libraries: TensorFlow, Scikit-learn, Flask / Django

- Database: MySQL / PostgreSQL
- Web Server: Apache / Nginx
- Data Capture Tools: Zeek, Tshark, Tcpdump
- IDE / Editor: Visual Studio Code / PyCharm
- Cloud Interface: Custom private cloud setup on Linux server
- Security Tools**:** OpenSSL, iptables, fail2ban

Intrusion prevention system follows a structured workflow that integrates real-time traffic analysis, machine learning–based detection, and advanced data protection mechanisms. Figure 3 illustrates the overall methodology and data flow of the proposed system.

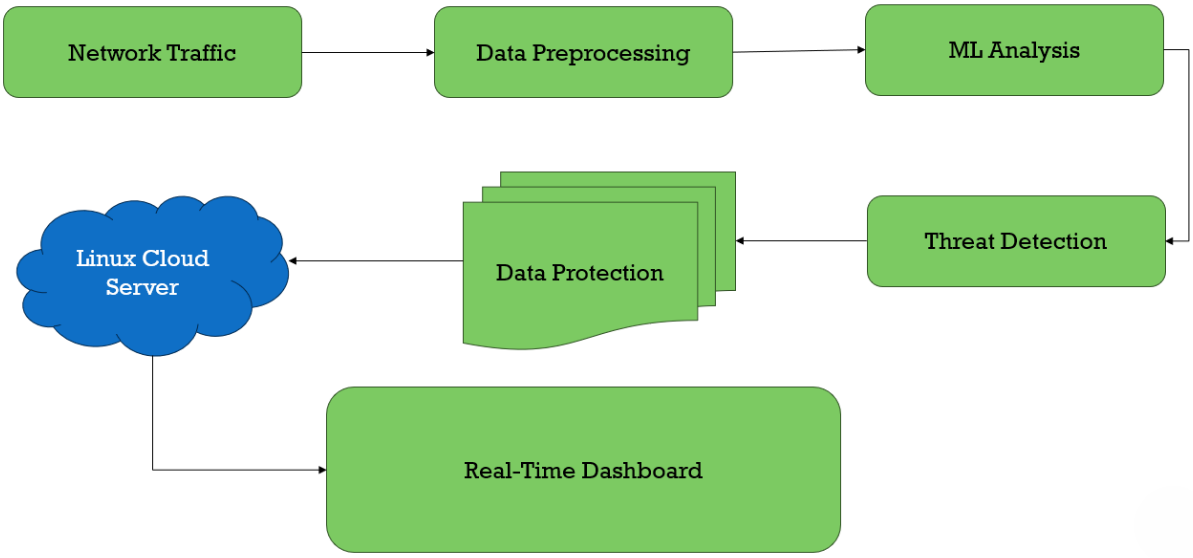


Fig 3. Proposed Intrusion Prevention Architecture

Initially, real-time network traffic is captured from connected devices and forwarded to the data preprocessing module, where noise removal, normalization, and feature extraction are performed. The processed data is then analysed by the machine learning analysis module, which employs trained models to identify anomalous or malicious behaviour. Upon detection of a threat, the threat detection module triggers preventive actions and forwards relevant information to the data protection layer, where encryption and access control mechanisms are applied. All components are hosted within a Linux-based private cloud server, which ensures secure storage, centralized processing, and administrative control. Finally, detected threats, system status, and security metrics are visualized through a real-time dashboard for continuous monitoring and decision support.

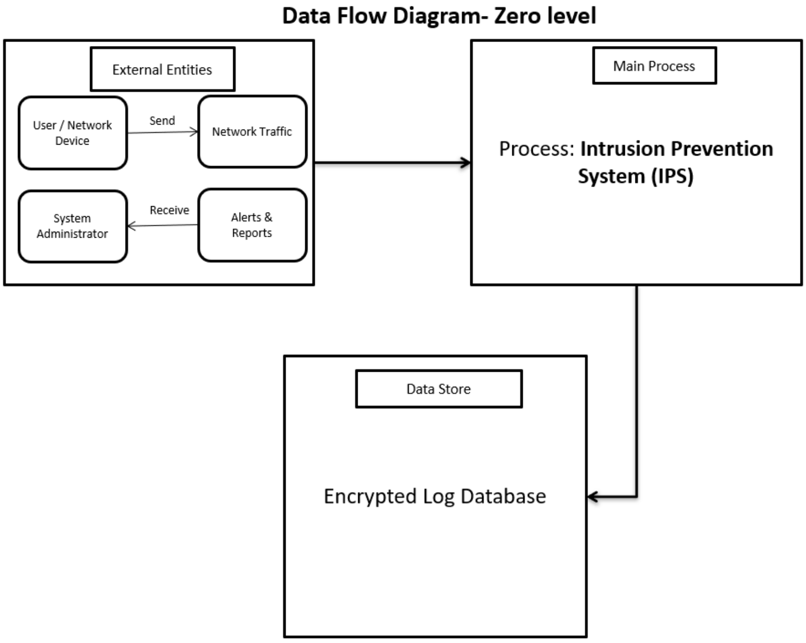
IV. DATA FLOW DIAGRAM

The data flow diagram illustrates the movement of data within the proposed intrusion prevention system, highlighting interactions between external entities, internal processes, and data stores at different levels of abstraction.

A. Data Flow Diagram (Level 0):

Figure 4.1 represents the Level 0 Data Flow Diagram of the proposed Intrusion Prevention System (IPS). At this level, the system is viewed as a single process interacting with external entities. Network users and devices generate traffic that is sent to the IPS, while the system administrator receives alerts and reports. All detected events and logs are securely stored in an encrypted log database for auditing and analysis.

DOI: 10.17148/IJARCCE.2026.15142



. Fig 4.1. Level 0 Data Flow Diagram of the proposed IPS

B. Data Flow Diagram (Level 1)

Figure 4.2 illustrates the Level 1 Data Flow Diagram, which decomposes the IPS into major functional modules. Network data is collected and pre-processed before being analysed by the machine learning module. Based on the analysis, threat decisions are made, alerts are generated, and system status is displayed on the dashboard. The data store maintains raw traffic, extracted features, encrypted logs, and trained model information.

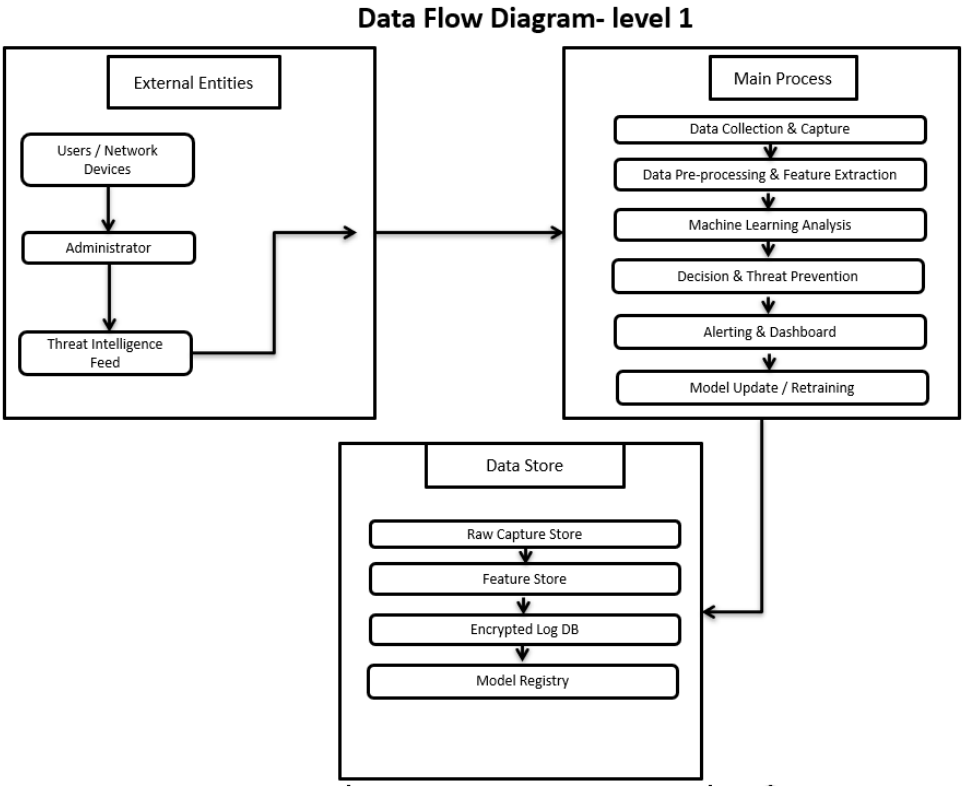


Fig 4.2 Level 1 Data Flow Diagram showing the internal processes of the proposed IPS.

C. Data Flow Diagram (Level 2)

Figure 4.3 presents the Level 2 Data Flow Diagram, providing a detailed view of internal operations. It includes packet sniffing, log acquisition, traffic filtering, data cleaning, normalization, feature engineering, and labeling. The machine learning engine performs model loading, classification, and threat scoring. Based on severity evaluation, prevention actions such as rule matching and firewall control are executed. Alerts, logs, feedback, and model retraining are managed to ensure continuous system improvement.

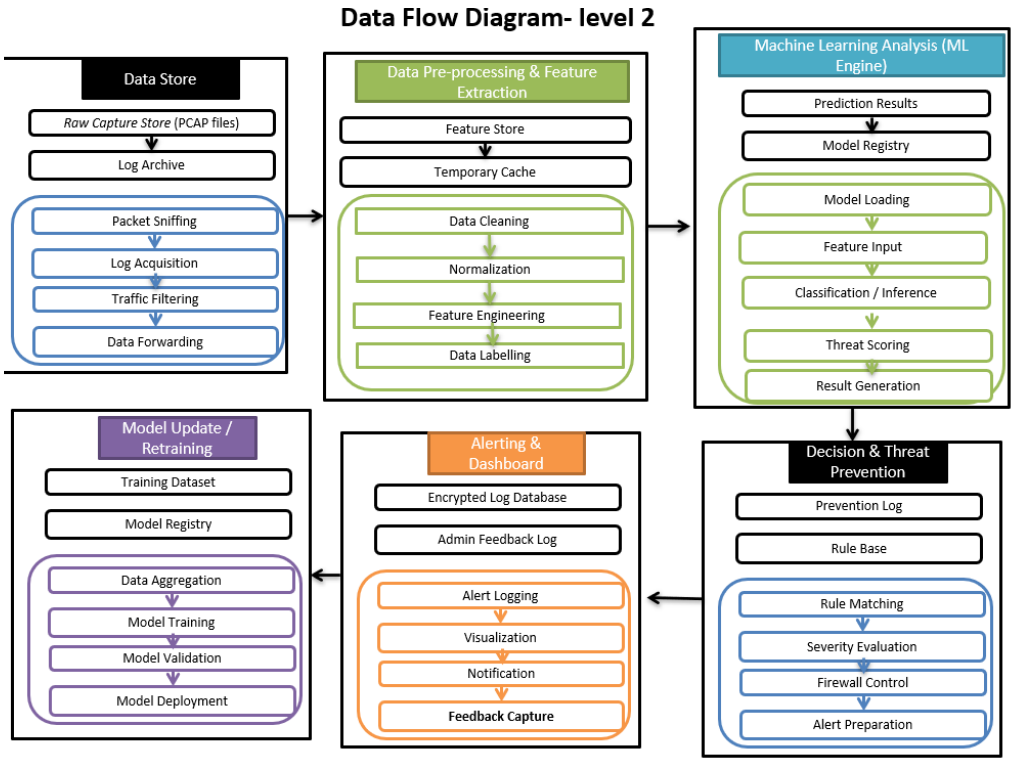


Fig 4.3 Level 2 Data Flow Diagram detailing data processing, machine learning analysis, and threat prevention.

V. RESULT AND DISCUSSION

Figure 5.1 illustrates the detection frequency of various attack types and the performance comparison of the machine learning models employed in the proposed intrusion prevention system. Random Forest achieved the highest overall accuracy with fast classification, SVM provided stable detection with moderate precision, and the DNN model demonstrated superior recall by effectively identifying complex intrusion patterns. The results show effective detection of DoS, port scanning, brute force, and intrusion attacks, while benign traffic constitutes the majority of observed network activity. This confirms the system's capability to accurately distinguish malicious behaviour from normal network traffic.

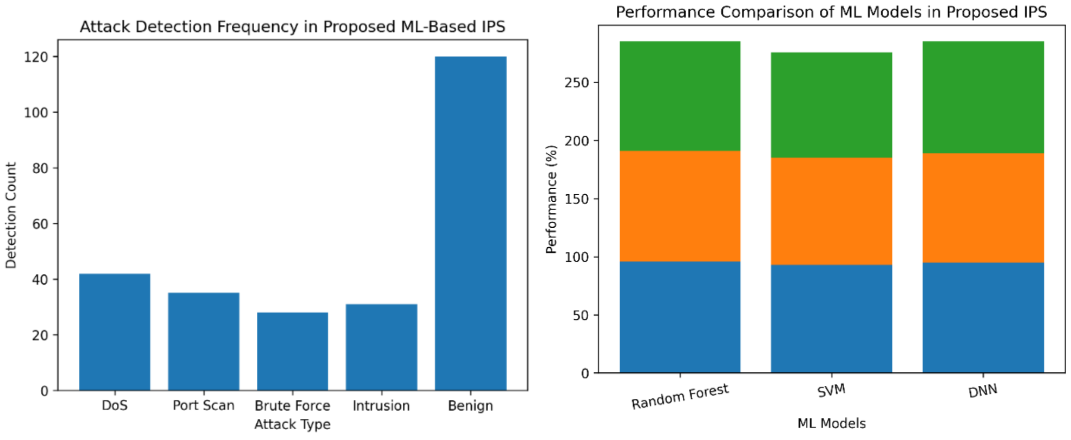


Fig 5.1 Attack Detection Frequency & Performance comparison of ML models

DOI: 10.17148/IJARCCE.2026.15131

comparative analysis of the performance of machine learning models used in the proposed intrusion prevention system. The stacked bars represent accuracy, precision, and recall values for Random Forest, SVM, and DNN models. Random Forest demonstrates the highest overall accuracy and precision, indicating reliable and fast classification of network traffic. The SVM model shows comparatively lower performance, particularly in recall, reflecting reduced sensitivity to certain attack patterns. In contrast, the DNN model achieves the highest recall, highlighting its effectiveness in detecting complex and subtle intrusion behaviours. Overall, the figure indicates that while Random Forest is suitable for real-time deployment due to its efficiency, DNN provides enhanced detection capability for sophisticated attacks, thereby improving the robustness of the intrusion prevention system.

VI. CONCLUSION

This paper presented a machine learning–based intrusion prevention system capable of detecting and preventing cyberattacks in real time while ensuring advanced data protection. The system integrated Random Forest, Support Vector Machine, and Deep Neural Network models within a Linux-based private cloud environment. Experimental evaluation showed that the Random Forest model achieved the highest accuracy of 96% with a precision of 95% and recall of 94%, making it well suited for real-time intrusion prevention. The SVM model attained an accuracy of 93%, precision of 92%, and recall of 91%, providing stable but comparatively lower detection performance. The DNN model demonstrated strong capability in identifying complex attack patterns, achieving a recall of 96%, with an accuracy of 95% and precision of 94%.

The results confirm that the proposed system effectively distinguishes malicious traffic from normal network behaviour while minimizing false positives. The combination of automated prevention actions, encrypted data storage, and real-time visualization enhances system reliability and security. Overall, the study demonstrates that machine learning significantly improves intrusion prevention effectiveness compared to traditional approaches. Future work will focus on adaptive model retraining, larger datasets, and deployment in large-scale network environments.

REFERENCES

[1]. Cyber security in the age of Iot: Are your Devices Spying on You?Mbonigaba Celestin N.Vanitha International Journal of Multidis ciplinary Research and Modern Education (IJMRME)2015.

[2]. A Socio-technical Approach to Preventing, Mitigating, and Recovering from Ransomware Attacks Dean F. Sittig and Hardeep Singh Applied Clinical Informatics 2016.

[3]. Anatomy of a Real-time Intrusion Prevention System Ricardo Koller, Raju Rangaswami, Joseph Marrero, Igor Hernandez, Geoffrey Smith, Mandy Barsilai, Silviu Necula, S. Masoud Sadjadi, Tao Li,and Krista Merrill Florida International University Technical Report 2007.

[4]. Intrusion Detection and Prevention System Using Machine Learning and Deep Learning Techniques Prathamesh Margale, Shreya Kadam, Atharva Kakade, Prasad Papade, Prasad Papade, Prof. Naved Raza Q. Ali, and Prof. Ganesh D. Jadhav (IJARCCE) 2024.

[5]. Network Intrusion Detection and Prevention System Using ML Algo rithm Bhushan Ajit Jadhav, Prof. Amit Nichat International Journal of Creative Research Thoughts (IJCRT)2024.

[6]. CryptoLock (and Drop It): Stopping Ransomware Attacks on User Data Nolen Scaife, Henry Carter, Patrick Traynor, and Kevin R.B. Butler IEEE 36th International Conference on Distributed Computing Systems 2016.

[7]. Finding Pegasus: Enhancing Unsupervised Anomaly Detection in High Dimensional Data using a Manifold-Based Approach R.P. Nathan, Niko laos Nikolaou, Ofer Lahav arXiv, 2025.

[8]. Network Traffic Analysis and Intrusion Detection using Packet Sniffer Mohammed Abdul Qadeer Arshad Iqbal Second International Confer ence on Communication Software and Networks 2010.

[9]. Machine Learning-Based Intrusion Detection and Prevention System for IIoT Smart Metering Networks: Challenges and Solutions Sahar Lazim, Qutaiba I. Ali IEEE 36th International Conference on Distributed Com puting Systems 2016.