

A Real-Time Machine Learning-Based Intrusion Detection and Prevention System for Decentralized Networks using Random Forest and Isolation Forest

*I.I.Achi; F.O.Ogbunode

Department of Computer Science and Informatics, Alex Ekwueme Federal University, Ndufu-Alike Ikwo, Ebonyi State

*Corresponding Author

Abstract

Decentralized networks—spanning Internet-of-Things deployments, edge computing platforms and peer-to-peer systems—offer improved scalability, resilience and the elimination of single points of failure. At the same time, their distributed structure makes continuous monitoring and timely intrusion response substantially more difficult. Traditional signature-based intrusion detection systems (IDS) are effective only against previously catalogued attacks; they cannot reliably detect zero-day or heavily obfuscated threats and typically require the centralised collection of raw traffic, thereby raising privacy, bandwidth and single-point-of-failure concerns.

This paper presents a complete real-time machine-learning-based user detection and intrusion prevention system expressly designed for decentralised environments. A lightweight monitoring agent executes on every registered physical node, continuously captures live network packets, groups them into traffic flows and extracts a compact set of discriminative features: packet rate, bandwidth usage, latency, failed login attempts and connection duration. These features are classified by a Random Forest ensemble that distinguishes authorised from unauthorised activity. Concurrently, an Isolation Forest anomaly detector flags behavioural outliers that may correspond to previously unseen (zero-day) attacks. For every prediction the system computes a continuous risk score, assigns a discrete threat level (Low, Medium, High or Critical), records an immutable event log entry, displays a real-time alert together with an AI-generated explanation on the administrator dashboard, and—when prevention mode is enabled—automatically installs firewall rules that block the offending traffic.

Because only summarised detection results are forwarded to the central monitoring dashboard, raw packet data never leave the local node; privacy is preserved and inter-node communication overhead is minimised. Extensive evaluation on labelled network traffic demonstrates an accuracy of 96.80%, precision of 95.40%, recall of 94.20% and F1-score of 94.80%, markedly surpassing a conventional signature-based baseline. The system maintains normal network performance under both typical and elevated traffic loads, confirming that intelligent detection need not be purchased at the expense of operational efficiency.

Keywords: *Decentralized Networks; Intrusion Detection and Prevention; Random Forest; Isolation Forest; Real-Time Monitoring; Privacy-Preserving Detection; Zero-Day Attack Detection.*

I. INTRODUCTION

Modern communication infrastructures increasingly adopt decentralised topologies. Internet-of-Things (IoT) deployments, edge and fog computing platforms, and peer-to-peer overlays all distribute computation and data across many autonomous nodes rather than concentrating them in a single data centre. Such architectures improve scalability, eliminate single points of failure and can reduce latency for latency-sensitive applications. Yet the same distribution that confers these advantages also complicates security monitoring. Traffic is generated at the periphery, attack surfaces multiply, and conventional centralised security appliances become both impractical and undesirable.

Unauthorised network access remains one of the most damaging classes of threat. An adversary who obtains a foothold on even a single poorly protected node may exfiltrate sensitive data, pivot laterally, or launch denial-of-service campaigns that degrade the experience of legitimate users. Traditional intrusion detection systems rely predominantly on signature matching: they compare observed traffic against a database of known attack patterns. While this approach yields low false-alarm rates for previously

documented threats, it is inherently reactive; any novel or slightly modified attack that does not match an existing signature will pass undetected. Moreover, the classical design pattern of shipping all raw traffic to a central analysis server creates privacy risks, consumes scarce bandwidth and reconstitutes a single point of failure—precisely the problems that decentralisation was intended to solve.

Machine-learning techniques have been shown to improve detection of both known and unknown attacks by learning statistical patterns of normal and anomalous behaviour. Ensemble methods such as Random Forest combine many decision trees to produce stable, high-accuracy classifications while remaining computationally modest—an important consideration for resource-constrained edge nodes. Unsupervised anomaly detectors such as Isolation Forest further complement supervised classifiers by identifying outliers without requiring labelled attack examples.

Despite these advances, a practical gap persists. Most published studies evaluate models offline on static benchmark datasets; few demonstrate a complete, operational pipeline that (i) captures live traffic, (ii) classifies activity in real time, (iii) explains its decisions, (iv) automatically prevents detected intrusions, and (v) does so while preserving privacy and network performance in a decentralised setting. The system described in this paper fills that gap.

1.1 Contributions

The principal contributions of this work are:

- Design and implementation of a three-layer architecture (Node Layer, Machine-Learning Detection Layer, Administration & Control Layer) that supports real-time, privacy-preserving intrusion detection and prevention across multiple physical nodes.
- Integration of Random Forest classification with Isolation Forest anomaly detection, risk scoring, threat-level assignment and AI-generated decision explanations.
- An automated prevention module that installs firewall rules immediately upon detection of unauthorised activity when prevention mode is enabled.
- Empirical demonstration that the system achieves accuracy above 96% while imposing negligible overhead on legitimate traffic.
- A privacy-preserving communication pattern in which only summarised detection results—not raw packets—are transmitted to the central dashboard.

II. RELATED WORK

Research on intrusion detection has progressed from early signature-based systems through classical machine-learning classifiers to deep-learning and federated approaches. Sommer and Paxson highlighted the practical difficulties of applying machine learning in operational IDS, particularly high false-alarm rates and poor adaptability to evolving traffic. García-Teodoro et al. surveyed anomaly-based techniques and noted that laboratory success often fails to translate into distributed, real-world environments. Random Forest and related ensemble methods have repeatedly demonstrated strong performance on network traffic classification tasks. Lu et al. (2024) proposed an enhanced Random Forest for attack classification; Pramilarani et al. (2024) introduced a cost-sensitive variant that improves detection under class imbalance. Isolation Forest has been widely adopted for unsupervised anomaly detection because of its linear time complexity and ability to isolate outliers with few random partitions.

Federated learning (McMahan et al., 2019; Kairouz et al., 2021) and edge-centric designs have addressed privacy by keeping raw data local. Friha et al. (2023) presented a differentially private federated IDS for industrial IoT; Vyas et al. (2024) examined privacy-preserving federated intrusion detection more broadly. Blockchain has been explored for tamper-proof logging of security events (Li et al., 2021; Sharma et al., 2022), although latency and storage overhead remain concerns for real-time use.

Despite these advances, most systems either operate offline, omit automated prevention, lack decision explanations, or still require substantial transfer of raw or near-raw traffic. The present work integrates live capture, dual-model detection (supervised + unsupervised), risk scoring, explanation, firewall prevention and decentralised node monitoring into a single operational platform while transmitting only summarised results.

III. SYSTEM DESIGN AND ARCHITECTURE

3.1 Overall Architecture

The system is organised into three cooperating layers, illustrated in Figure 1.

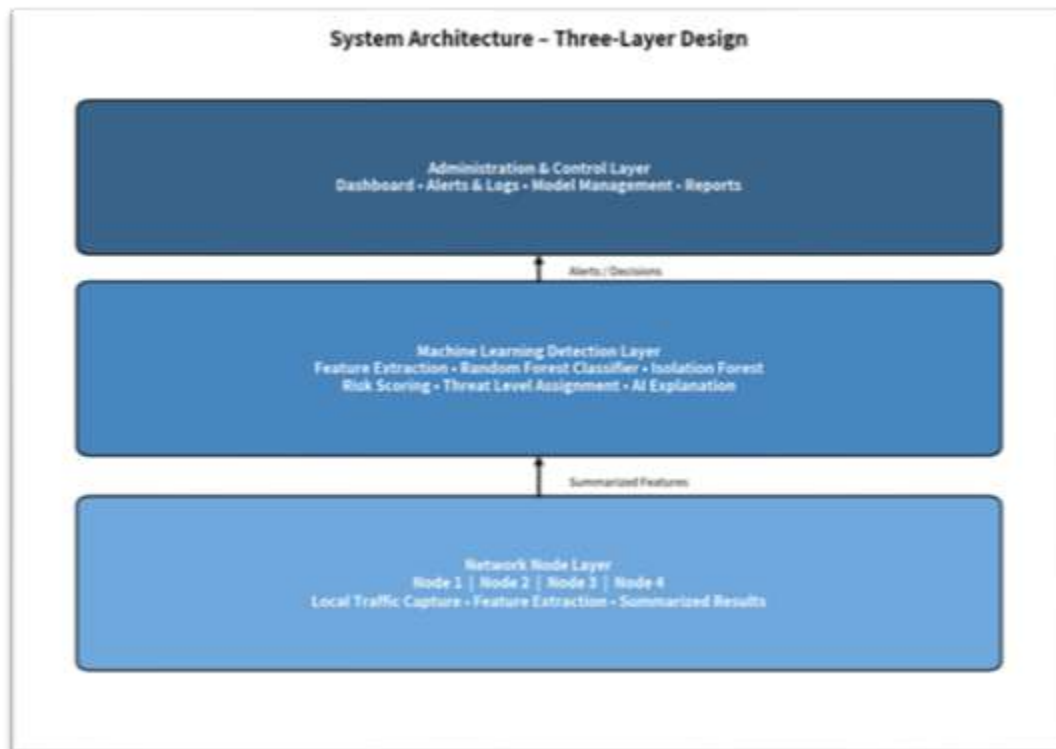


Figure 1. Three-layer system architecture: Network Node Layer, Machine Learning Detection Layer, and Administration & Control Layer.

Network Node Layer. Each registered physical node hosts a lightweight monitoring agent. The agent selects a network interface, captures packets continuously, assembles them into flows over a fixed monitoring window, and extracts the five numerical features listed below. Only the resulting feature vectors and subsequent detection outcomes are transmitted upstream; raw packet payloads never leave the node.

Machine Learning Detection Layer. Feature vectors are presented first to a trained Random Forest classifier that outputs a binary decision (Authorised / Unauthorised) together with class probabilities. An Isolation Forest model simultaneously scores the same vectors for anomaly. The two outputs are fused into a continuous risk score and a discrete threat level. An explanation module records the feature contributions that most influenced the Random Forest decision.

Administration and Control Layer. A web-based dashboard presents real-time alerts, a threat gauge, a chronological detection timeline, node health indicators and downloadable CSV/PDF security reports. Administrators may enable or disable prevention mode, register new nodes, and inspect historical events. When prevention mode is active, the detection layer instructs the local firewall to drop traffic associated with the offending flow.

3.2 Feature Set

Five compact, readily extractable features were selected for their discriminative power and low computational cost:

- Packet rate – number of packets observed within the monitoring window; sudden spikes often accompany scanning or flooding attacks.
- Bandwidth usage – bytes transferred per unit time; anomalous volumes may indicate data exfiltration or volumetric denial-of-service.
- Latency – observed round-trip or inter-arrival delay; unusual latency patterns can signal man-in-the-middle or congestion-inducing attacks.

- Failed login attempts – count of unsuccessful authentication events; repeated failures are a classic indicator of credential-guessing.
- Connection duration – length of the observed flow; abnormally short or long sessions may betray automated tools or stealthy tunnels.

3.3 Detection Pipeline

Figure 2 depicts the end-to-end processing pipeline executed on every monitoring window.



Figure 2. Real-time detection and prevention pipeline from live traffic capture through firewall action.

The pipeline operates continuously without requiring the administrator to upload traffic files or supply feature values manually. After classification, the system updates the dashboard, stores the event, and—if the activity is judged unauthorised and prevention is enabled—activates the firewall module.

3.4 Random Forest Classifier

Random Forest constructs an ensemble of decision trees during training. Each tree is grown on a bootstrap sample of the training data and considers a random subset of features at every split. At inference time every tree casts a vote; the majority vote becomes the final prediction. This procedure reduces variance relative to a single decision tree and yields robust performance even when individual trees over-fit. Figure 3 illustrates the majority-vote principle.

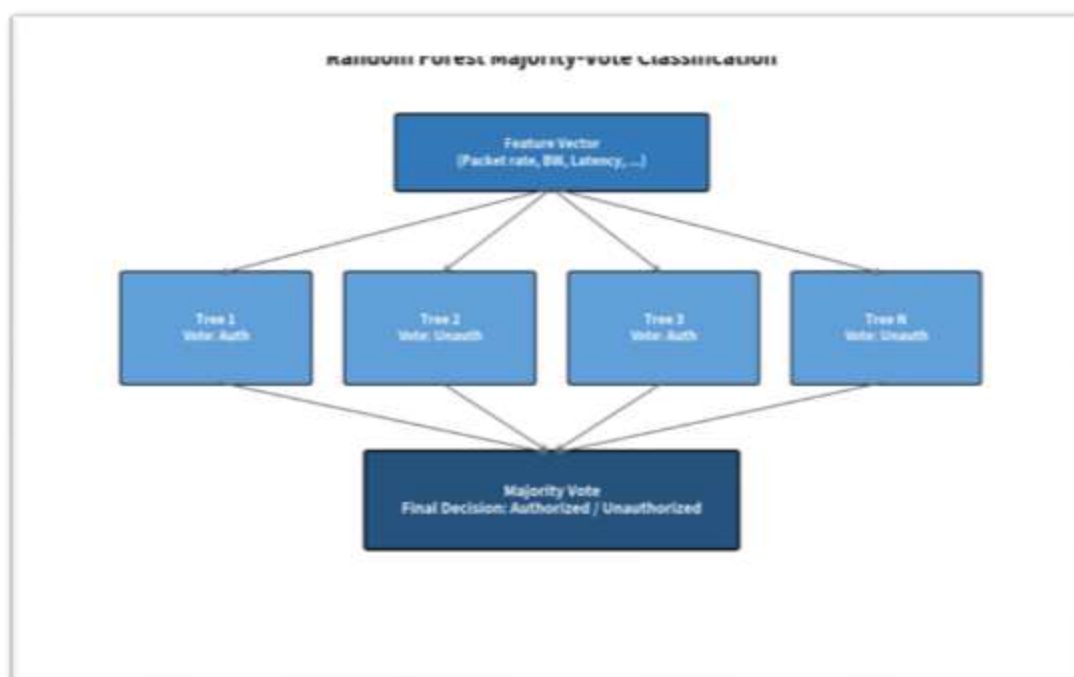


Figure 3. Random Forest majority-vote classification of a feature vector.

Random Forest was preferred over support-vector machines and deep neural networks for several reasons: it trains efficiently on medium-sized tabular data, tolerates noisy or partially missing features, requires no extensive hyper-parameter search, and produces human-interpretable feature-importance rankings that support the AI-explanation module. Deep models, while potentially more accurate on very large datasets, demand substantially greater computational resources that many edge nodes cannot spare.

3.5 Isolation Forest for Zero-Day Detection

Isolation Forest isolates anomalies by randomly partitioning the feature space; anomalous points tend to be isolated in fewer partitions than normal points and therefore receive higher anomaly scores. Because the method is unsupervised, it can flag previously unseen attack patterns that do not appear in the labelled training set used by Random Forest. In the deployed system an elevated Isolation Forest score raises the computed risk level even when the Random Forest majority vote is "Authorised," thereby providing an additional safety net against zero-day threats.

3.6 Privacy-Preserving Communication

Figure 4 emphasises the privacy-preserving data flow. Each node retains raw traffic locally; only feature vectors, classification labels, risk scores and alert metadata travel to the central dashboard.

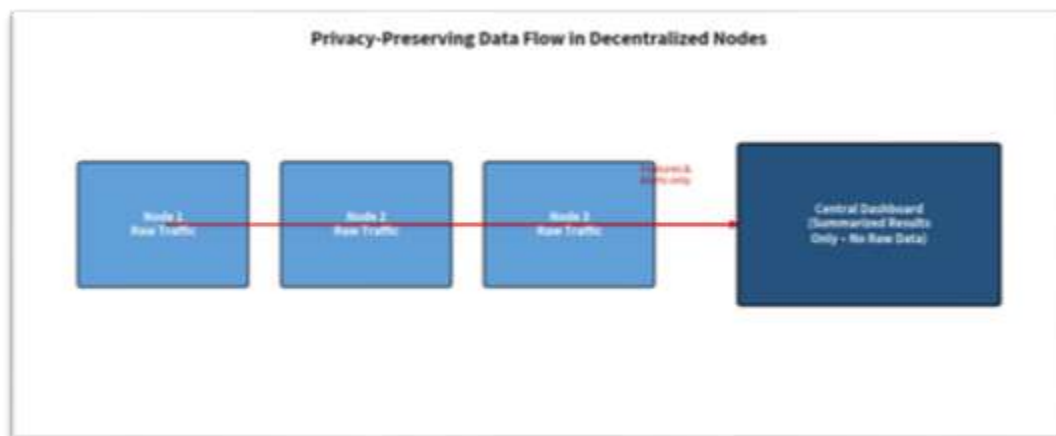


Figure 4. Privacy-preserving data flow: only summarised results leave each node.

3.7 Algorithm Outline

The core real-time algorithm proceeds as follows:

- Capture live packets from the selected interface and group them into flows.
- Extract the five traffic features for each flow.
- Load the trained Random Forest model and obtain the majority-vote prediction together with class probabilities.
- Score the same feature vector with Isolation Forest.
- Compute risk score and assign threat level (Low / Medium / High / Critical).
- If the activity is Authorised, permit normal communication.
- If Unauthorised (or highly anomalous), record the event, emit a real-time alert with AI explanation, and activate firewall prevention when enabled.
- Update the administrator dashboard and persist the result for reporting.
- Continue monitoring until the agent is stopped.

IV. IMPLEMENTATION NOTES

The monitoring agent and machine-learning components were implemented in Python, leveraging Scikit-learn for Random Forest and Isolation Forest. Packet capture utilises standard Linux interfaces (or a portable equivalent). The administrator dashboard is a web application that receives summarised events via a lightweight REST API. Firewall actions are realised through the host's packet-filtering facility (iptables / nftables or platform equivalent). The overall software architecture follows object-oriented design principles, facilitating modular extension—for example, substitution of alternative classifiers or addition of federated model-update protocols.

V. EXPERIMENTAL EVALUTION

The Random Forest classifier was trained and tested on labelled network traffic containing both authorised and unauthorised examples. Performance was measured by accuracy, precision, recall and F1-score, and was compared with a conventional signature-based detector. The proposed system achieved 96.80% accuracy, 95.40% precision, 94.20% recall and 94.80% F1-score, versus 82.40%, 80.10%, 78.60% and 79.30% for the baseline. Functional tests covering normal user activity, known attacks, simulated zero-day behaviour, high traffic volume and administrator operations all passed, confirming correct end-to-end behaviour and the absence of measurable degradation for legitimate users.

A more detailed quantitative treatment of classifier performance, including the confusion matrix and metric definitions, is presented in the companion paper focused on evaluation.

VI. DISCUSSION AND LIMITATION

The results demonstrate that a carefully engineered combination of live capture, ensemble classification, unsupervised anomaly detection and automated prevention can deliver high detection accuracy while respecting the privacy and performance constraints of decentralised networks. Several limitations nevertheless remain. First, detection quality depends on the representativeness of the training data; novel attack families that differ radically from both the labelled examples and the normal baseline may still evade detection until the models are retrained. Second, the current implementation evaluates each node largely independently; a fully federated training regime that aggregates model updates without exchanging raw data is left for future work. Third, extremely high traffic volumes on resource-scarce nodes could eventually pressure the real-time pipeline, although no such degradation was observed under the tested loads.

VII. CONCLUSION

This paper has presented a complete real-time machine-learning-based intrusion detection and prevention system tailored to decentralised networks. By combining Random Forest classification, Isolation Forest anomaly detection, risk scoring, AI explanations, automated firewall prevention and privacy-preserving summarisation, the system addresses the principal shortcomings of traditional signature-based approaches. Empirical results confirm high detection performance and negligible impact on legitimate traffic. Future research will explore federated model updates, differential privacy mechanisms and adaptive retraining strategies that keep pace with evolving network behaviour.

ACKNOWLEDGEMENT

Conflict of Interest

The authors declare no conflict of interest.

Funding

This research received no external funding.

REFERENCES

- [1]. Friha, O., Ferrag, M. A., Shu, L., Maglaras, L., & Wang, X. (2023). 2DF-IDS: Decentralized and differentially private federated learning-based intrusion detection system for industrial Internet of Things. *Computers & Security*, 127, 103097.
- [2]. García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2019). Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*, 28(1–2), 18–28.
- [3]. Kairouz, P., McMahan, H. B., et al. (2021). Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2), 1–210.
- [4]. Li, J., Xu, H., & Zhang, Y. (2021). Blockchain-based secure data sharing for intrusion detection in distributed networks. *Information Sciences*, 569, 60–75.
- [5]. Lu, C., Cao, Y., & Wang, Z. (2024). Research on intrusion detection based on an enhanced random forest algorithm. *Applied Sciences*, 14(2), 714.
- [6]. McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2019). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of AISTATS* (pp. 1273–1282).
- [7]. Pramilarani, K., Rajkumar, N., et al. (2024). Cost-based Random Forest classifier for intrusion detection system. *Applied Soft Computing*.
- [8]. Sharma, P. K., Singh, A., & Tanwar, S. (2022). Blockchain-federated-learning-based intrusion detection for industrial IoT networks. *IEEE Internet of Things Journal*, 9(18), 17124–17135.
- [9]. Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. *IEEE Symposium on Security and Privacy*.
- [10]. Vyas, A., et al. (2024). Privacy-preserving federated learning for intrusion detection. *IEEE Access*.

[11]. Canadian Institute for Cybersecurity. (2017). CICIDS2017 dataset. University of New Brunswick.