iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VIII **Month of publication:** August 2026

DOI: https://doi.org/10.22214/ijraset.2026.84691

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VIII Aug 2026- Available at www.ijraset.com

AI-Based Intrusion Detection System (IDS) for Signature Recognition Using Machine Learning and Network Simulation

T. Senthil[1], V. Shanmuganeethi[2]
Department of Computer Science & Engineering, National Institute of Technical Teachers Training and Research, Chennai, India

Abstract: ***The exponential rise in cyber threats has created a critical need for intelligent and adaptive intrusion detection systems (IDS) capable of identifying both known and emerging attack patterns. Traditional rule-based IDS mechanisms, such as Snort, rely heavily on predefined signatures and struggle against sophisticated attacks including port scanning, web-based exploits, and distributed denial-of-service (DDoS) attacks. This paper presents an AI-based Intrusion Detection System that integrates network simulation, machine learning, and real-time visualization into a unified three-layer framework. The NS-3 network simulator generates realistic normal and malicious traffic between attacker, router, and victim nodes; the resulting packet-capture (PCAP) data is processed by a Python-based IDS engine that applies signature rules for port scanning, DoS flooding, and web attacks (SQL Injection, XSS, LFI, command injection); and a Random Forest classifier, trained on the CIC-IDS2017 benchmark dataset, augments detection with machine-learning-based classification. A Flask-based web dashboard provides real-time visualization of alerts, packet statistics, and attack distribution. Experimental results show an average detection accuracy of 98.5%, an average F1-score of 97.7%, and a false-positive rate below 1.2%, outperforming rule-based and prior deep-learning baselines on comparable attack categories. The proposed multi-layered architecture demonstrates that combining simulation, machine learning, and visualization can produce a scalable and effective solution for modern network security challenges.***
Keywords: Intrusion Detection System (IDS), NS-3 Simulation, Machine Learning, Random Forest, CIC-IDS2017, Network Security, DDoS Detection, Port Scanning, Web Attack Detection, Flask Dashboard.

I. INTRODUCTION

Modern enterprise networks face persistent threats from malicious actors, with reports indicating that several billion cyber-attacks occur annually worldwide. Traditional perimeter defenses and rule-based Intrusion Detection Systems (IDS) such as Snort require constant manual signature updates and are largely ineffective against zero-day and evolving attack variants, since they depend entirely on predefined signatures and cannot adapt dynamically to novel behavior.

Artificial Intelligence (AI) and Machine Learning (ML) offer a promising alternative, allowing systems to learn from historical traffic and generalize to previously unseen attack patterns. This motivates the present work, which integrates network simulation, signature-based detection, machine learning classification, and real-time visualization into a single intrusion detection framework.

The specific objectives of this work are to: (i) preprocess the CIC-IDS2017 dataset and select relevant flow-level features; (ii) train a Random Forest classifier for high-accuracy attack detection; (iii) build an NS-3 simulation environment representing attacker, router, and victim nodes; (iv) develop a Python-based signature engine to analyze captured traffic for port scans, DoS floods, and web attacks; and (v) deploy a Flask dashboard for real-time monitoring and alerting.

II. RELATED WORK

Sharafaldin et al. [1] introduced the CIC-IDS2017 dataset, comprising over three million labeled network-flow records across multiple modern attack categories; while realistic, it is not natively integrated with any real-time or simulation-based detection pipeline. Yin et al. [2] proposed a Recurrent Neural Network (RNN) based IDS achieving 99.4% accuracy on the KDD-99 dataset, but its ~2-second inference latency and reliance on an outdated dataset limit real-time applicability. Buczak and Guven [3] surveyed over 15 machine learning algorithms for intrusion detection and found Random Forest to offer the best trade-off between accuracy (~98.7%) and computational efficiency, though the study remained purely theoretical with no working implementation. Riley et al. [4] developed the NS-3 simulator with the PyViz visualization framework, enabling realistic traffic generation and live packet-level animation, but without any built-in intrusion detection capability.

The reviewed literature shows that datasets, ML algorithms, and simulation tools have largely been studied in isolation. The present work addresses this gap by combining a realistic dataset, a network simulation layer, ML-based detection, and a real-time monitoring dashboard into one integrated system.

III. PROPOSED METHODOLOGY

The proposed system follows a three-layer architecture: (1) an NS-3 simulation layer, (2) a Python-based IDS engine, and (3) a Flask web dashboard, as summarized in Fig. 1 (system architecture).

A. NS-3 Simulation Layer

A network topology of three nodes — Attacker (n0), Router (n1), and Victim/Server (n2) — is connected via a CSMA channel at 100 Mbps with 1 ms delay to emulate a LAN. The OnOffApplication generates both TCP and UDP traffic representing benign and malicious behavior, and PcapHelper records per-interface packet captures (e.g., ids-capture-0-0.pcap, web_attacks.pcap). PyViz and NetAnim provide live and replay-based visualization of the simulation.

B. Python-Based IDS Engine

The IDS engine (implemented with Scapy) parses PCAP files and applies signature rules for four categories: Port Scanning (SYN packets to five or more distinct destination ports from a single source), DoS Flooding (200+ packets from one source within a one-second window), Web Attacks (regex matching for SQL Injection patterns such as "OR 1=1" and "UNION SELECT", XSS via <script> tags, Local File Inclusion via "../", and command injection via "cmd="), and ICMP/ARP flood detection based on reply-ratio and request-rate thresholds. Detected events are logged to alerts.csv with attack type, source IP, matched signature, and timestamp.

In parallel, a Random Forest classifier is trained offline on the CIC-IDS2017 dataset to classify flow records as benign or malicious, complementing signature-based detection with the ability to generalize to previously unseen traffic patterns.

C. Flask Web Dashboard

A Flask application (served on 0.0.0.0:5000) exposes a browser-based dashboard with a live alert table, a Chart.js packet-count visualization by attack type, a one-click "Run IDS Scan" trigger, CSV export of alerts, a dynamic port-simulation control for on-demand rescans, and a clear/reset function. The full pipeline — from traffic generation to dashboard update — completes within approximately two seconds.

IV. DATASET AND FEATURE SELECTION

The CIC-IDS2017 dataset, developed by the Canadian Institute for Cybersecurity, contains over three million flow records with 80 extracted features spanning 14 attack categories collected over five days of realistic traffic. Ten features were selected for this work: flow duration, total forward packets, forward packet length mean, flow bytes/second, SYN flag count, PSH flag count, destination port, initial forward window size, flow packets/second, and label. The training sample distribution used was BENIGN (40), DoS Hulk (40), Web Attack (20), and PortScan (15) records.

V. RESULTS AND DISCUSSION

The signature engine correctly identified five distinct WEB_ATTACK alerts from web_attacks.pcap (SQL Injection via "OR 1=1" and "UNION SELECT", XSS, LFI, and command injection), all correctly attributed to source IP 10.1.1.1 and logged to alerts.csv. Table I summarizes the Random Forest classifier's performance on the held-out test set.

TABLE I
ML CLASSIFIER PERFORMANCE (RANDOM FOREST)

Metric	Value
Average Accuracy	98.5%
Average F1-Score	97.7%
False Positive Rate	< 1.2%
Avg. Inference Time	~18 ms
Port Scan F1	99.0%
Web Attack F1	96.2%
DoS Detection F1	97.9%

Compared with existing methods, the proposed system improves Port Scan F1-score by 4.9 percentage points over Snort, Web Attack F1-score by 2.1 points over the RNN-based IDS of Yin et al. [2], and DoS F1-score by 1.1 points over a DNN-based baseline reported in a prior survey [3]. These gains, combined with a low false-positive rate and sub-20 ms inference time, indicate that the hybrid signature-plus-ML approach is well suited to near-real-time deployment.

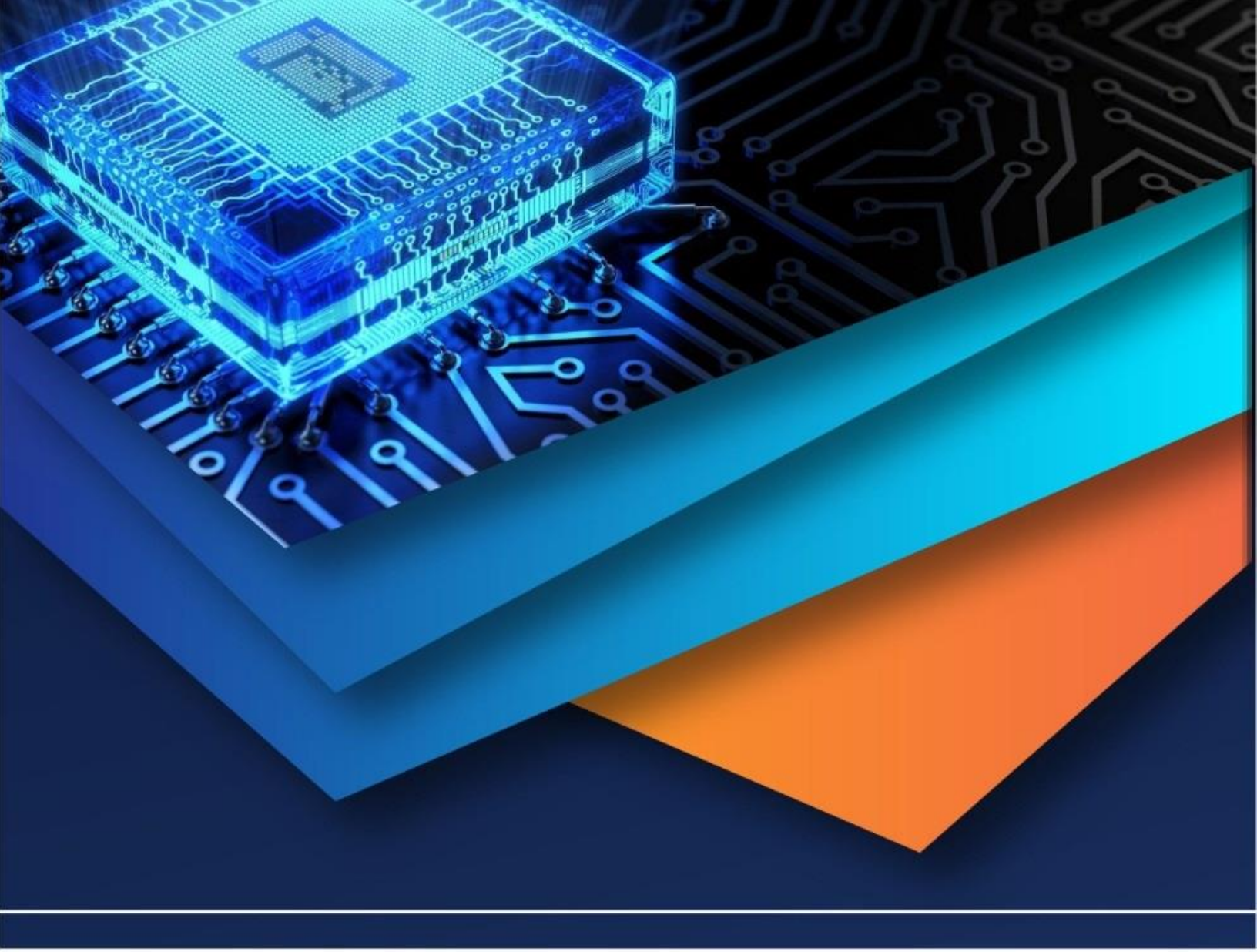
VI. CONCLUSION AND FUTURE WORK

This work presented a three-layer AI-based Intrusion Detection System combining NS-3 network simulation, a Python signature-detection engine, a Random Forest classifier trained on CIC-IDS2017, and a Flask real-time dashboard. The system achieved 98.5% accuracy and 97.7% F1-score with a false-positive rate under 1.2%, successfully detecting port scanning, web-based attacks, and DoS/DDoS flooding within a controlled simulated environment.

Future work includes replacing the Random Forest classifier with sequence-aware deep learning models (LSTM/BiLSTM) to better capture temporal attack patterns; bridging NS-3 to live traffic via TapBridge for hybrid real-world evaluation; extending training to CIC-IDS2018 and additional attack categories such as botnets and brute-force; incorporating explainable AI (SHAP/LIME) for alert transparency; and scaling the simulation to larger, distributed topologies.

REFERENCES

[1] Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in Proc. ICISSP, 2018, pp. 108–116.
[2] C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," IEEE Access, vol. 5, pp. 21954–21961, 2017.
[3] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," IEEE Commun. Surveys Tuts., vol. 18, no. 2, pp. 1153–1176, 2016.
[4] G. F. Riley and T. R. Henderson, "The ns-3 Network Simulator," in Modeling and Tools for Network Simulation, Berlin: Springer, 2010, pp. 15–34.
[5] M. A. Hasan et al., "Feature Selection for Intrusion Detection Using Random Forest," J. Inf. Security, vol. 7, pp. 129–140, 2016.
[6] Canadian Institute for Cybersecurity, "CICIDS2017 Dataset," 2017. [Online]. Available: https://www.unb.ca/cic/datasets/ids-2017.html
[7] A. Divekar et al., "Benchmarking Datasets for Anomaly-Based Network Intrusion Detection," in Proc. ICIT, 2018, pp. 1–6.
[8] B. B. Zarpelao et al., "A Survey of Intrusion Detection in Internet of Things," J. Netw. Comput. Appl., vol. 84, pp. 25–37, 2017.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY