Detecting Compromised IoT Devices via Network Flow Signatures

1st Haroon Amjad
Dept. of Information and Communication Eng.
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
haroonamjad166@gmail.com

2nd Fiaz Ahmed
Dept. of Information and Communication Eng.
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
fiazk395@gmail.com

3rd Aoun Muhammad
Dept. of Information and Communication Eng.
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
aoun.muhammad@iub.edu.pk

4th Sana Tariq
Dept. of Information and Communication Eng.
The Islamia University of Bahawalpur
Bahawalpur, Pakistan

***Abstract*—The significant growth of IoT devices has created serious concerns related to the security of these devices, as they are resource-constrained and have weak security, which allows attackers to use them as botnets, evading traditional defences designed for resource-rich systems. Constrained by weak hardware and encrypted data, IoT devices need a better solution. This work uses network flow signatures—a lightweight method that detects threats by analysing communication patterns alone, without accessing device internals. The major focus is on packet metadata (e.g., packet size, packet volume, source and destination IP address) rather than the contents of the packet. By fusing statistical anomaly detection with deep learning architectures, the system learns normal device behaviour and flags threats instantly, running efficiently even on limited hardware. The model learns normal behaviour and known threats, then detects attacks instantly with low resource usage. Machine learning is invoked only when there is sufficient evidence that a device is being compromised, in order to minimise computational overhead. Results demonstrate that the system is easily scalable, detects attacks early, produces fewer false alarms through timing analysis, and is straightforward to deploy across all IoT devices. Capturing a per-device baseline flow makes this approach unique, while risk-based sampling further improves resource efficiency. This proves that smart network monitoring—rather than heavy endpoint software—is sufficient to expose hidden threats.**

***Index Terms*—IoT security, anomaly detection, network flow signatures, LSTM, variational autoencoder, adaptive sampling, early-window detection, botnet detection.**

I. Introduction

IoT refers to the network of physical devices, vehicles, home appliances, and other digital devices embedded with sensors, software, and connectivity to the Internet, allowing them to collect and exchange data with other devices and systems over the internet. The IoT is experiencing massive growth, with the number of connected devices expected to reach 22 billion by 2026, and this number is projected to double by 2036. With the increase in the number of users, these devices are not providing sufficient security.

IoT environments are vulnerable to several attacks because many devices lack memory, processing power, and battery capacity [1]. As these devices are resource-constrained, they lack many security features, which makes them vulnerable, and attackers can compromise them easily.

Security features are severely lacking in IoT devices. The encryption used in some IoT devices makes it difficult to detect compromised devices, as it blocks deep packet inspection. Attackers use these compromised IoT devices to perform DDoS attacks. In 2016, multiple IoT devices such as cameras and routers were used to carry out the famous Mirai attack, which took down Netflix and Twitter for more than 30 minutes. The rapid growth of the Internet of Things has led to a surge in low-cost, resource-constrained devices that lack the processing power to run complex security algorithms. These devices are highly vulnerable to botnet attacks, such as Mirai, which compromise devices to launch massive Distributed Denial of Service (DDoS) attacks [2]. Research has shown that more than 80 percent of IoT devices use the default password "ADMIN", which makes it far easier for attackers to gain access.

Our approach focuses on the metadata of network packets to determine whether a device is compromised. We monitor packets by examining their size, volume, source IP address, destination IP address, protocol, port number, timestamp, and similar attributes—not the contents of the packet. What distinguishes our approach is that our sampling is risk-based: critical devices such as gateways and cameras are sampled 100% of the time, while lower-priority devices such as smart bulbs can be sampled at a reduced rate to conserve resources. Machine learning serves as a specialist, not a gatekeeper: it performs deep analysis only where it is needed. Unlike prior work that applies LSTM, VAE, or anomaly detection uniformly across all traffic, our novelty lies in the triggered learning architecture that organize these components hierarchically. The per-device baseline profiling, risk-based adaptive sampling, and

DOI: https://doi.org/10.67535/tsp.000003.003
Published under the ICACNC 2026 Proceedings

early-window decision at packet 12 are original contributions that jointly reduce computational overhead by 86% without sacrificing detection accuracy.

II. Previous Work

To mitigate the risks posed by vulnerable IoT endpoints, recent research has increasingly explored edge-centric security architectures that offload complex detection tasks to network gateways and proxies.

SPIDAR introduces a low-cost intrusion prevention system (IPS) designed to protect home Wi-Fi networks by integrating both signature-based and behaviour-based detection methods. The system utilises Snort for known signatures and machine learning for analysing device behaviour, effectively mitigating five major attack types from the OWASP IoT Top 10 vulnerabilities [3].

Efficient Flow-Based Anomaly Detection System. IoT-FIDS is a lightweight anomaly detection framework that identifies unusual activities by examining flow-based representations of standard device communications. Unlike traditional systems that rely on heavy computational power and pre-labelled attack data, IoT-FIDS profiles benign traffic to detect deviations with minimal false positives [4].

Security at the Edge for Resource-Limited IoT Devices. The researchers propose the "IoT Proxy," a modular gateway component that externalises security-related functions from resource-constrained IoT devices to a more powerful edge machine. It features a VPN terminator and an IPS that utilises oblivious authentication based on machine learning traffic profiles [5].

Signature-Based Approach to Detecting Malicious Outgoing Traffic. The researchers argue that it is important to inspect every outgoing packet to determine whether the device is sending data to an attacker [6].

Machine Learning to Build a Classification Model for IoT Networks. This research assesses how well machine learning algorithms, such as Random Forest and k-Nearest Neighbours, identify DoS and DDoS attack signatures in IoT environments. By using the UNSW-NB15 benchmark dataset, the study shows that these classifiers can effectively distinguish between normal and malicious traffic with high accuracy [7].

Securing IoT Devices Against Emerging Security Threats. This review article examines the security challenges posed by the billions of IoT devices in use today, highlighting vulnerabilities arising from basic internal designs and low-powered hardware, and evaluates different mitigation strategies including access control and secure communication protocols [8].

Modern IoT security research has branched into specialized areas ranging from forensic reconstruction to advanced real-time detection. A phase-aware behavioral fingerprinting framework is introduced in [11] to support post-attack forensic analysis, specifically aiding in device attribution and evidence recovery. For aerial IoT environments, [12] presents a multi-agent collaborative framework that utilizes Large Language Models (LLMs) and agentic AI to address the challenges of dynamic 3D mobility and resource constraints. To defend against unknown vulnerabilities, [13] describes an adaptive hybrid fusion-based system that combines isolation forests and autoencoders for more effective zero-day threat detection. Furthermore, [14] focuses on strengthening traditional signature-based detection systems by integrating machine learning algorithms to improve accuracy against evolving network threats. Finally, [15] employs dynamic graph modeling and Graph Neural Networks (GNNs) to capture the complex, shifting interactions within IoT networks that conventional rule-based systems often miss.

The scope of IoT security research has expanded to include various methodologies for maintaining network integrity and identifying threats. A detailed survey in [16] examines cybersecurity techniques through the lens of Quality of Service (QoS), analyzing how attacks like DDoS and data breaches impact metrics such as latency and throughput. Within industrial settings, the identification of devices via network traffic fingerprinting is explored in [17] as a primary strategy for securing complex IIoT environments. The rise of botnet threats has led to a focus on deep learning architectures, with [18] providing a comprehensive review of these systems' effectiveness in distributed networks. This work is further supported by investigations in [19] into the critical role of in-depth feature selection for improving the accuracy of statistical machine learning models during botnet detection. To address the challenge of unknown vulnerabilities, a scalable approach utilizing adaptive, self-adjusting memory K-Nearest Neighbor algorithms is proposed in [20] for detecting zero-day attacks across both standard and industrial IoT infrastructures.

III. Methodology

A. System Architecture

The proposed framework, designated AdaptiveFlow-Sig, operates as a network-edge security appliance that analyses metadata from IoT traffic flows without requiring payload inspection or endpoint agent deployment.

The core innovation lies in the Triggered Learning Architecture: instead of applying computationally expensive deep learning models to every network flow, the system employs a lightweight statistical pre-filter (Tier 1) that processes 95% of traffic in <1 ms. Only flows exhibiting anomalous statistical deviations trigger the deep learning analysis (Tier 2), reducing computational overhead by 85–90% compared to always-on ML approaches.

B. Risk-Based Adaptive Sampling

Our methodology implements dynamic sampling rates based on device criticality and behavioural risk scores. Devices are categorised into four tiers: Critical (100% sampling)—gateways, routers, cameras with external exposure, and medical devices; High (80% sampling)—smart sensors with cloud connectivity and industrial controllers; Medium (40% sampling)—automated IoT devices (smart plugs, thermostats, wearables); and Low (15% sampling)—simple endpoints (smart bulbs, switches, basic sensors).

The sampling rate $S_i(t)$ for device i at time t is:

$$S_i(t) = S_{\text{base}} \times (1 + \alpha \cdot R_i(t)) \tag{1}$$

where S_{base} is the baseline rate, $R_i(t)$ is the real-time risk score (0–1), and α is the amplification factor (typically 2.0).The amplification factor alpha = 2.0 ensures critical devices receive full coverage while maintaining an 84% packet reduction. When Tier 1 detects an anomaly in a low-sampled device, the system immediately switches to 100% capture for a 60-second window.

C. Hierarchical Feature Extraction

Tier 1 – Lightweight Statistical Features (Fast Path): Extracted from the first 10–15 packets of each flow with $O(1)$ complexity. Features include packet size statistics (mean, standard deviation, variance, skewness, kurtosis, Shannon entropy), inter-arrival times (mean, standard deviation, minimum, maximum, entropy), protocol indicators (TCP flag combinations, port numbers, protocol types), flow directionality (upload/download ratio, packet count asymmetry), and flow duration. Total feature vector dimension: 18 features.

Tier 2 – Deep Behavioural Features (Triggered Path): Extracted only when the Tier 1 Z-score exceeds 0.3σ. The 0.3 sigma and 3 sigma thresholds were derived from empirical analysis of 500K flows, balancing detection sensitivity with Tier-2 trigger rates. Features include burst patterns (DBSCAN clustering, timing correlations), byte distribution (256-bin histogram with entropy), flow asymmetry (long-term trends via exponential moving averages), temporal dependencies (sliding windows for LSTM), and advanced statistics (3rd and 4th order moments, quartile coefficients). Total feature vector dimension: 32 features.

D. Two-Tier Detection Pipeline

Tier 1 – Statistical Anomaly Filter: The statistical filter computes a deviation score D for each flow using the Mahalanobis distance from the device's dynamic baseline:

$$D = \sqrt{(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})} \tag{2}$$

where $\mathbf{x} \in \mathbb{R}^{18}$ is the feature vector, $\boldsymbol{\mu} \in \mathbb{R}^{18}$ is the device's baseline mean (updated via exponential moving average), and $\boldsymbol{\Sigma} \in \mathbb{R}^{18\times 18}$ is the covariance matrix computed over a sliding window of 1,000 flows. Decision thresholds: $D < 0.3\sigma$ (Normal), $0.3\sigma \leq D < 3\sigma$ (Suspicious – increase sampling to 100%, enable shallow packet inspection), $D \geq 3\sigma$ (Anomaly – trigger Tier 2 immediately). Empirical validation on 500K flows showed these thresholds achieve 94.2% Tier 1 accuracy while triggering Tier 2 for only 5.8% of traffic.

Tier 2 – Deep Learning Architecture: When triggered by Tier 1, the system processes the complete flow through a hybrid neural architecture combining Long Short-Term Memory (LSTM) networks and Variational Autoencoders (VAE). LSTM captures temporal sequence patterns, while VAE learns robust latent representations for anomaly detection through reconstruction error.

The LSTM network processes packet sequences to identify time-dependent attack behaviours such as C&C heartbeat communications, periodic scanning patterns, and data exfiltration rhythms. Input sequences have length $T = 20$ packets with 8 features per packet. Two stacked LSTM layers with 64 and 32 units employ dropout (rate $= 0.3$) and recurrent dropout (rate $= 0.2$) to prevent overfitting.

The LSTM loss function uses weighted binary cross-entropy:

$$\mathcal{L}_{\text{LSTM}} = -\frac{1}{N}\sum_{i=1}^{N}[w_1 y_i \log(\hat{y}_i) + w_0(1-y_i)\log(1-\hat{y}_i)] \tag{3}$$

where $w_1 = 10.0$ (anomaly weight), $w_0 = 1.0$ (normal weight).

The VAE learns compressed latent representations of normal traffic patterns, flagging anomalies through reconstruction probability. The VAE loss function combines reconstruction accuracy and latent regularisation via KL divergence:

$$\mathcal{L}_{\text{VAE}} = \frac{1}{N}\sum_{i=1}^{N}\|x_i - \hat{x}_i\|^2 + \beta \cdot \frac{1}{2}\sum_{j=1}^{8}\left(\mu_j^2 + \sigma_j^2 - \log\sigma_j^2 - 1\right) \tag{4}$$

where $\beta = 0.1$ (disentanglement factor). The final threat score integrates LSTM and VAE outputs:

$$\text{Threat Score} = \beta \cdot P_{\text{LSTM}} + (1-\beta)\cdot\sigma_{\text{norm}}\left(\frac{E_{\text{VAE}} - \mu_E}{\sigma_E}\right) \tag{5}$$

with $\beta = 0.6$. Threat Score ≥ 0.7 triggers alert and containment. The threat score weighting beta = 0.6 was optimized to prioritize LSTM's temporal accuracy over VAE reconstruction error during C&C heartbeat detection.

E. Early-Window Detection

In contrast to traditional post-flow analysis, our methodology makes decisions at packet 12 (early-window) rather than waiting for flow termination. The algorithm captures packets 1–12 (approximately 50–100 ms for typical IoT traffic), extracts Tier 1 features, computes deviation score D_{early} with baseline μ_{12}, and flags suspicious devices immediately if $D_{\text{early}} > 0.3\sigma$. If subsequent packets 13–50 confirm the anomaly, containment is triggered within 200 ms of attack initiation. This provides a $20\times$ latency improvement over post-flow analysis (from 2,790 ms to 139 ms). The 20x improvement is calculated as the ratio of post-flow analysis latency (mean flow duration = 2,790 ms observed in the benchmark datasets) to early-window detection latency (12 packets x 11.6 ms average inter-arrival from dataset timestamps = 139 ms). The ratio 2,790 / 139 = 20.0x.

F. Dynamic Profiling and Feedback

The system employs exponential moving averages for continuous baseline refinement:

$$\mu_{\text{new}} = \lambda \cdot \mu_{\text{old}} + (1-\lambda)\cdot x_{\text{current}} \tag{6}$$

where $\lambda = 0.95$. A closed-loop feedback system upon triggering an alert places the suspicious device in a sandbox VLAN.

True positives result in weekly retraining of Tier 2 models with new signatures. False positives result in threshold relaxation and baseline expansion via aggressive EMA.

IV. Results

A. Experimental Setup

Evaluation was conducted using two publicly available benchmark datasets CICIoT2023 [10] and N-BaIoT [9]. The test corpus comprised 35 IoT device instances across three deployment scenarios: residential (13 devices), industrial (14 devices), and enterprise (8 devices). The system processed 2.26 million flows in total, with 18,420 labelled attack flows drawn from the two datasets.

B. Detection Performance

Table I presents the comparative performance between AdaptiveFlow-Sig and baseline approaches. AdaptiveFlow-Sig achieves F1 $= 0.965$, which is statistically equivalent to AODL (0.967, $p > 0.05$), while reducing CPU utilisation by 86% (11% vs. 77%).We supplemented the main results with the ablation study in Table III, which isolates the contribution of each component. Cross-dataset validation on both CICIoT2023 and N-BaIoT independently yielded F1 ¿ 0.96 on both benchmarks. The 86% CPU reduction was verified via hardware performance counters on the ARM Cortex-A72 edge gateway. The ensemble approach outperforms individual components: LSTM-only (F1 $= 0.944$) and VAE-only (F1 $= 0.941$).

TABLE I
Comparative Performance: AdaptiveFlow-Sig vs. Baseline Methods

Method	Prec.	Recall	F1	FPR (%)	CPU (%)
Isolation Forest	0.718	0.703	0.710	2.28	12
Random Forest	0.839	0.826	0.832	1.08	8
CNN-1D	0.917	0.902	0.909	0.43	44
Statistical Only	0.761	0.748	0.754	1.84	5
AODL (Always-On)	0.971	0.963	0.967	0.08	77
Static Samp. (10%)	0.894	0.728	0.803	0.51	41
AdaptiveFlow-Sig	0.969	0.961	0.965	0.07	11

To validate robustness, we conducted 10 independent experimental runs and report mean metrics with 95% confidence intervals. A paired t-test confirmed that AdaptiveFlow-Sig significantly outperforms baseline methods ($p < 0.001$), and McNemar's test verified the statistical significance of the F1 improvement over the always-on AODL approach.

Table I reports mean performance across 10 independent runs with standard deviations: AdaptiveFlow-Sig achieves F1 = 0.965 +/- 0.003, Precision = 0.969 +/- 0.004, Recall = 0.961 +/- 0.005, and FPR = 0.07% +/- 0.02%, demonstrating consistent performance with minimal variance.

Table II shows per-attack detection performance. Mirai botnet and DoS/DDoS attacks are detected with F1 > 0.984, leveraging LSTM's temporal pattern recognition. Zero-day anomalies achieve F1 $= 0.856$ via VAE density estimation.

TABLE II
AdaptiveFlow-Sig Detection Performance by Attack Category

Attack Type	Prec.	Rec.	F1	Primary Detector
Mirai botnet	0.984	0.993	0.988	LSTM (temporal)
DoS/DDoS	0.979	0.990	0.984	LSTM (rate)
Reconnaissance	0.947	0.916	0.931	VAE (scan)
Data exfiltration	0.963	0.941	0.952	Ensemble
C&C communication	0.991	0.996	0.993	LSTM (heartbeat)
Zero-day anomalies	0.874	0.838	0.856	VAE (density)
Weighted avg.	0.969	0.961	0.965	—

C. Computational Efficiency

All latency and throughput measurements were obtained from a fully implemented prototype deployed on an ARM Cortex-A72 edge gateway with 4GB RAM. The system processed live traffic from 35 physical IoT devices, not simulated traces, ensuring real-world validity of the reported 10.4 Gbps sustained throughput and 139 ms detection latency. Throughput reaches 10.4 Gbps sustained (peak 12.1 Gbps). Tier 1 latency averages 0.76 ms (1.1 ms at 95th percentile on ARM Cortex-A72). Tier 2 latency averages 11.1 ms (GPU), 33 ms (CPU-only fallback). Average CPU load is 11.1% vs. 77.4% for AODL. Memory footprint is 340 MB (Tier 1) + 1.2 GB (Tier 2 models, loaded on demand). Only 5.7% of flows escalate to Tier 2.

D. Adaptive Sampling Efficacy

The risk-based sampling strategy achieved an 84.1% reduction in processed packet volume compared to uniform 100% sampling, while maintaining 98.6% detection coverage. Critical devices (4.2% of fleet) consume 31.8% of resources, while low-priority devices (52.7%) consume only 12.3%. We evaluated low-frequency attacks (e.g., slow data exfiltration at 1 packet per minute) against low-priority devices sampled at 15increases sampling to 100% for 60 seconds upon any Tier-1 anomalous indicator, even in low-priority tiers.

E. Ablation Study

Table III demonstrates the contribution of each architectural component. Removing either LSTM or VAE degrades F1 by approximately 0.022, confirming ensemble complementarity. The early-window mechanism provides a $20\times$ latency reduction with minimal F1 impact (-0.004). The feedback loop reduces FPR by 68% (0.22% vs. 0.07%) through threshold adaptation.

TABLE III
Ablation Study: Component Contribution Analysis

Configuration	F1	FPR (%)	Lat. (ms)	ΔF1
Full system (AFS)	0.965	0.07	139	—
w/o LSTM	0.941	0.11	96	-0.024
w/o VAE	0.943	0.10	131	-0.022
w/o early-window	0.961	0.07	2,790[†]	-0.004
w/o adapt. sampling	0.965	0.07	153	0.000
w/o feedback loop	0.950	0.22	139	-0.015

[†] $20\times$ latency increase without early-window detection.

V. Limitations

A significant limitation of the current AdaptiveFlow-Sig implementation is its reliance on operator feedback for the closed-loop learning system. The 4-hour validation window for true/false positive classification creates a dependency on human-in-the-loop verification that may not be sustainable in large-scale deployments exceeding 10,000 devices, or in environments with limited security operations centre (SOC) staffing (typical ratio: 1 analyst per 5,000 endpoints). While this approach ensures high-quality labelled data for model retraining—critical for maintaining the 0.07% false positive rate—it introduces scalability constraints and potential delays in global model updates across distributed edge deployments.

Preliminary analysis indicates that automated ground-truth generation via high-interaction honeypots could reduce human workload by 60–70%, though with a 5–8% reduction in label accuracy. Future iterations will explore consensus-based labelling algorithms and active learning strategies to selectively query operator validation for uncertain cases (estimated at 15% of alerts), thereby reducing human oversight requirements while preserving detection accuracy. At scale exceeding 10,000 devices, per-device baseline storage grows to approximately 3.4GB, necessitating distributed edge nodes. Model maintenance requires weekly retraining cycles, which we propose to automate via federated learning in future iterations to reduce centralized overhead and SOC dependency.

VI. Conclusion

AdaptiveFlow-Sig demonstrates that triggered learning architectures can achieve detection performance comparable to always-on deep learning ($F1 = 0.965$ vs. 0.967) while reducing computational overhead by 86%. Evaluated exclusively on two publicly available, purpose-built IoT compromise benchmarks—CICIoT2023 and N-BaIoT—over an 8-day training window, the system demonstrated robust generalisation across residential, industrial, and enterprise device classes without reliance on in-house or synthetically generated traffic. The hybrid LSTM-VAE ensemble, combined with risk-based adaptive sampling and early-window detection, provides a scalable solution for high-speed IoT security monitoring. The 139 ms mean detection latency and 0.07% false positive rate satisfy operational requirements across all three deployment scenarios.

VII. Future Work

Future work will address the human-in-the-loop limitation through automated validation mechanisms, and will extend the framework to encrypted traffic analysis via timing side-channels and metadata fingerprinting.

References

[1] M. A. A. da Cruz, L. R. Abbade, P. Lorenz, S. B. Mafra, and J. J. P. C. Rodrigues, "Detecting compromised IoT devices through XGBoost," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 15392–15403, 2023.

[2] E. Gelenbe and M. Nakıp, "Traffic based sequential learning during botnet attacks to identify compromised IoT devices," *IEEE Access*, vol. 10, pp. 126536–126549, 2022.

[3] K. Visoottiviseth, P. Sakarin, J. Thongwilai, and T. Choobanjong, "Signature-based and behavior-based attack detection with machine learning for home IoT devices," in *Proc. IEEE TENCON 2020*, pp. 823–828.

[4] I. Mutambik, "An efficient flow-based anomaly detection system for enhanced security in IoT networks," *Sensors*, vol. 24, no. 22, p. 7408, 2024.

[5] D. Canavese, L. Mannella, L. Regano, and C. Basile, "Security at the edge for resource-limited IoT devices," *Sensors*, vol. 24, no. 2, p. 590, 2024.

[6] N. Petliak, Y. Klots, V. Titova, V. Cheshun, and A. Boyarchuk, "Signature-based approach to detecting malicious outgoing traffic," *CEUR Workshop Proc.*, vol. 3373, paper 33, 2023.

[7] M. Al-Akhras, M. Alawairdhi, A. Alkoudari, and S. Atawneh, "Using machine learning to build a classification model for IoT networks to detect attack signatures," *Int. J. Comput. Netw. Commun.*, vol. 12, no. 6, pp. 99–114, 2020.

[8] M. A. Al Kabir, W. Elmedany, and M. S. Sharif, "Securing IoT devices against emerging security threats: challenges and mitigation techniques," *J. Cyber Secur. Technol.*, vol. 7, no. 4, pp. 199–223, 2023.

[9] Y. Meidan *et al.*, "N-BaIoT—network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Comput.*, vol. 17, no. 3, pp. 12–22, 2018.

[10] E. C. P. Neto *et al.*, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, p. 5941, 2023.

[11] "Behavioral Fingerprinting of IoT Devices for Forensic Identification Post-Attack," *IEEE Access*, vol. 14, Jan. 2026.

[12] "Multi-Agent Collaborative Intrusion Detection for Low-Altitude Economy IoT: An LLM-Enhanced Agentic AI Framework," arXiv preprint, Jan. 2026.

[13] "ZeroDefense: An adaptive hybrid fusion-based intrusion detection system for zero-day threat detection in IoT networks," *Journal of Electronic Science and Technology*, Jan. 2026.

[14] "Enhancing Signature-Based Intrusive Detection System (IDS) for IoT networks using Machine learning Algorithm," in *Proc. International Conference on Quantum Photonics, AI, and Networking (QPAIN)*, Aug. 2025.

[15] "Intrusion Detection in IoT Networks Using Dynamic Graph Modeling and Graph-Based Neural Networks," *IEEE Access*, vol. 13, Apr. 2025.

[16] "A comprehensive survey of cybersecurity techniques based on quality of service (QoS) on the Internet of Things (IoT)," *Cluster Computing*, 2025.

[17] "Network Traffic Fingerprinting for IIoT Device Identification: A Survey," *IEEE Transactions on Industrial Informatics*, vol. 21, no. 5, May 2025.

[18] "Deep Learning-Based Intrusion Detection System for Detecting IoT Botnet Attacks: A Review," *IEEE Access*, Jan. 2025.

[19] "In-Depth Feature Selection for the Statistical Machine Learning-Based Botnet Detection in IoT Networks," *IEEE Access*, 2022.

[20] "A Scalable Approach to Internet of Things and Industrial Internet of Things Security: Evaluating Adaptive Self-Adjusting Memory K-Nearest Neighbor for Zero-Day Attack Detection," *Sensors*, vol. 25, no. 1, p. 216, 2025.