Communications on Applied Electronics (CAE) – ISSN : 2394-4714
Foundation of Computer Science FCS, New York, USA
Volume 8 – No. 1, August 2025 – www.caeaccess.org

Real-Time Simulation and Detection of Unauthorized Devices in IoT Networks using MAC-Level Monitoring

Eman Gaber Ahmed Mahomud, PhD
Electronic Eng. and Communication Technology Department
Modern Academy for Engineering and Technology, Egypt

ABSTRACT
In modern Internet of Things (IoT) environments, the detection of unauthorized devices is critical for maintaining network security and integrity. This paper presents a MATLAB-based simulation framework that monitors connected devices in a wireless network and distinguishes between authorized and intruder devices using MAC address filtering. The simulation dynamically generates device activity over time, logs connection events, and visualizes trends in intrusions. Statistical analysis such as intrusion ratios, detection accuracy, and temporal patterns are computed. The proposed tool serves both as a security validation method and a data generation model for future intrusion detection system (IDS) research.

Keywords
IoT Security, Intrusion Detection, MAC Filtering, Simulation, Network Monitoring, MATLAB

1. INTRODUCTION
The exponential growth of IoT devices has increased the surface area of attacks on modern networks. Unlike traditional IT systems, IoT networks often operate in resource-constrained, heterogeneous, and dynamic environments, making security enforcement challenging [1].

One fundamental threat is the unauthorized access by rogue devices attempting to infiltrate or disrupt the system. Therefore, there is a growing need for efficient, lightweight, and real-time methods to detect unauthorized devices and respond promptly [2].

To quantify network activity at each time step t, we define:

$N(t) = A(t) + U(t)$

Where:

- N(t) is the total number of active devices at time t
- A(t) is the number of authorized devices
- U(t) is the number of unauthorized (intruder) devices

$R(t) = U(t) / N(t) = U(t) / (A(t) + U(t))$

This ratio acts as a key indicator of the network's security status. A high value of R(t) implies that the network is largely compromised, potentially requiring immediate countermeasures [3].

The simulation introduced in this study uses these formulations to model dynamic network behavior and generate real-time insight into device legitimacy. The results are used to analyze the effectiveness of detection systems and explore system vulnerability across time [4].

2. BACKGROUND AND RELATED WORK
The Internet of Things (IoT) has rapidly grown into an ecosystem of heterogeneous and resource-constrained devices—ranging from smart home appliances to industrial sensors and

healthcare monitors—which significantly increases the network's attack surface. This widespread connectivity and device diversity expose IoT networks to threats such as unauthorized device access, data privacy breaches, and service disruption. Background Lightweight access control methods, like MAC address filtering, are often used in IoT environments due to their simplicity and low cost. Although MAC filtering can be bypassed by spoofing, it remains a practical baseline security mechanism, especially in simulation environments or networks with limited computational resources. Such filtering enables statistical monitoring of unauthorized activities, which aids in intrusion analysis. Related Work Intrusion Detection Approaches in IoT Recent surveys have highlighted a wide array of IoT intrusion detection strategies, including signature-based, anomaly-based, specification-based, and hybrid approaches, many of which employ machine learning (ML) techniques to improve detection performance [8], [9]. Deep learning (DL) has been increasingly explored within anomaly-based IDS frameworks, showing promise in handling complex and dynamic IoT attack patterns [10], [11]. ML-based approaches often require robust datasets. For example, supervised ML classifiers such as Logistic Regression, SVM, Decision Trees, and Artificial Neural Networks have been benchmarked with research datasets like Bot-IoT and IoTID20 to evaluate detection. Accuracy and robustness [9]. Simulation-Based and Lightweight Frameworks Simulation frameworks play an important role in advancing IoT security research. MATLAB has been widely used due to its strong capabilities for modeling, statistical analysis, and visualization. One study implemented a hybrid intrusion detection system within a wireless IoT network using MATLAB simulation [11]. Other works combine real IoT hardware with ML-based detection. For example, in [12], IoT attacks such as ARP poisoning and man-in-the-middle were generated, and datasets were used to train classifiers including Naïve Bayes, SVM, and Adaboost. Unauthorized Device Detection via Machine Learning Detection of unauthorized IoT devices using ML has been specifically investigated. Meidan et al. [13] applied Random Forest classifiers to network traffic features, achieving up to 96% accuracy for unauthorized device types and 99% for authorized device recognition. Similarly, anomaly-based profiling systems that leverage traffic profiling and ML showed high accuracy (□98%) with very low false positive rates when

tested on the Cyber-Trust IoT testbed [8]. Motivation While significant research has been conducted on ML-based IDS and

large-scale simulations, there is still a gap in lightweight, MATLAB-based frameworks that: 1. Simulate dynamic device activity and intrusion attempts. 2. Apply simple baseline techniques like MAC filtering. 3. Log connection events in time. 4. Compute and visualize statistical metrics such as intrusion ratio, detection accuracy, and temporal trends. 5. Provide reproducible datasets for future IDS development. This paper addresses that gap by proposing a MATLAB-based framework that combines network monitoring, MAC-based filtering, temporal logging, and statistical analysis—serving both as a security validation tool and a data generation model for the IDS research community [13].

3. LIMITATIONS OF EXISTING METHODS AND THE NEED FOR INNOVATIVE ARCHITECTURES

Existing intrusion detection methods in Internet of Things (IoT) environments often suffer from several critical limitations that hinder their effectiveness when deployed in real-world scenarios:

1.**Static Rule-Based Filtering**: Many current systems rely on static rules (e.g., predefined MAC address whitelists). While simple, these methods are unable to adapt to the dynamic and heterogeneous nature of IoT networks where devices frequently join and leave.

2. **Scalability Challenges**: Traditional IDS solutions designed for enterprise networks do not scale efficiently to the massive number of devices in IoT ecosystems. As the device pool grows, these systems experience computational overhead and delays in detection.

3. **High False Positive Rates**: Signature- and anomaly-based IDS approaches often misclassify legitimate devices as intruders due to their reliance on incomplete or outdated training data. This reduces trust in the system and increases the burden on network administrators.

4.**Limited Temporal Analysis**: Most existing frameworks focus on snapshot-based detection rather than analyzing device behavior over time. Without temporal modeling, patterns such as intermittent intrusions or stealthy attacks remain undetected.

5. **Lack of Lightweight Simulation and Evaluation Tools**: Available simulation environments are either too heavy lack flexibility for rapid experimentation. Researchers often face difficulty in generating controlled datasets that reflect IoT-specific attack vectors. --- ### Need for Innovative Architectures Given these limitations, there is a clear need for lightweight, adaptable, and simulation-driven architectures that: - Dynamically

generate device activity and intrusions in controlled environments. - Perform real-time MAC-based classification combined with temporal logging. - Offer statistical insights such as intrusion ratios, detection accuracy, and delay/jitter patterns. - Provide research-ready datasets for validating intrusion detection algorithms under diverse IoT conditions. The proposed MATLAB-based monitoring and intrusion-aware framework addresses these needs by combining real-time device simulation, event logging, and analytical visualization. This architecture not only serves as a security validation tool but also bridges the gap between synthetic simulation and practical IoT deployments, paving the way for next-generation intrusion detection research.

4. METHODS

The proposed framework simulates real-time device activity over a predefined number of time intervals. A fixed list of authorized MAC addresses is established, while a set of dynamic, randomly generated intruder MAC addresses is added at each time step. The simulation is implemented in MATLAB and performs the following operations.

Linear Programming (MILP) and Constraint Programming (CP) were employed to model container placement as a mathematical optimization problem, aiming to minimize resource wastage while meeting application requirements.

4.1 Key Contributions

Initial Heuristic Methods: Early works like [1] and [2] relied on basic placement algorithms such as round-robin and random selection. These methods were fast but inefficient.

Optimization Approaches: MILP-based methods [3], [4] provided mathematically rigorous solutions, though at the cost of scalability and real-time performance.

- Randomly selects a subset of devices to activate during each interval.
- Compares each active device's MAC address against the list of authorized devices.
- Logs the number of authorized and intruder devices in a .csv file.
- Plots time-based graphs showing trends in authorized and intruder activity.
- Generates three additional scenarios (labeled Scenario 2, 3, and 4) to model different threat environments or detection capabilities, allowing comparative analysis.

These scenarios can represent different IDS (Intrusion Detection System) configurations, e.g., basic filtering, threshold-based detection, or AI-based models.

4.2 Authorized and Unauthorized Device Modeling

- A fixed list of authorized MAC addresses is pre-defined.
- Intruder MAC addresses are generated dynamically and inserted randomly during simulation.

4.3 Simulation Parameters

- **Time intervals (t):** The simulation runs for TT discrete steps (e.g., 50–100).
- **Device pool:** N(t)=A(t)+U(t)N(t) = A(t) + U(t), where A(t)A(t) is the number of authorized devices and U(t)U(t) is the number of unauthorized devices.
- **Intrusion ratio:** R(t)=U(t)A(t)+U(t)R(t) = \frac{U(t)}{A(t) + U(t)}

4.4 Detection Metrics

To evaluate performance, the following are computed:

- **Detection Accuracy (DA):**

  DA=TP+TNTP+TN+FP+FNDA = \frac{TP + TN}{TP + TN + FP + FN}

- **Precision (P):**

  P=TPTP+FPP = \frac{TP}{TP + FP}

- **Recall (R):**

  R=TPTP+FNR = \frac{TP}{TP + FN}

Where TP = True Positive (intruder detected), TN = True Negative (authorized correctly identified), FP = False Positive, FN = False Negative.

4.5 Scenarios

Four detection scenarios are simulated:

- • Scenario 1: Basic MAC filtering.
- • Scenario 2: Threshold-based detection.
- • Scenario 3: Machine learning-assisted filtering (simulated).
- • Scenario 4: Hybrid detection with adaptive thresholds

4.6 Logging and Visualization

- All device events are logged in a CSV file.
- Time-series plots, bar charts, and heatmaps are generated for intrusion ratio, detection accuracy, and scenario comparison.

4.7 Workflow

- The system workflow is summarized as Device Pool Generation – A pool of devices is initialized to simulate network participants.
- Random Device Activation – Devices are randomly activated to represent dynamic network behavior.
- MAC-Based Filtering: Primary detection through direct comparison of device addresses with the authorized list.
- Temporal Analysis: Observing trends in intruder activity across different time intervals.
- Threshold Evaluation: Defining intrusion severity levels (low, medium, high) based on observed ratios.
- Algorithmic Comparison: Benchmarking different detection strategies (e.g., pure filtering vs. statistical learning) to assess performance.

4.8 Expected Contributions

- Lightweight Intrusion Detection: A simulation tool requiring minimal computational overhead, suitable for constrained IoT devices.
- Quantitative Evaluation: Provides measurable indicators such as intrusion ratio, jitter, and packet loss for future IDS research.
- Data Generation for IDS Training: The framework can generate labeled datasets of normal vs. intruder activity, aiding machine learning-based IDS development.
- Scalability: Can be extended to test larger IoT networks with varying device pools and mobility patterns.

Random Device Activation – Devices are randomly activated to represent dynamic network behavior.

MAC Address Verification – The activated devices' MAC addresses are compared against the authorized device list to identify legitimate and unauthorized nodes.

Logging and Metric Computation – All detected activities are logged, and key performance metrics are computed.

Outcome Visualization – Detection results are plotted and compared to evaluate system performance.

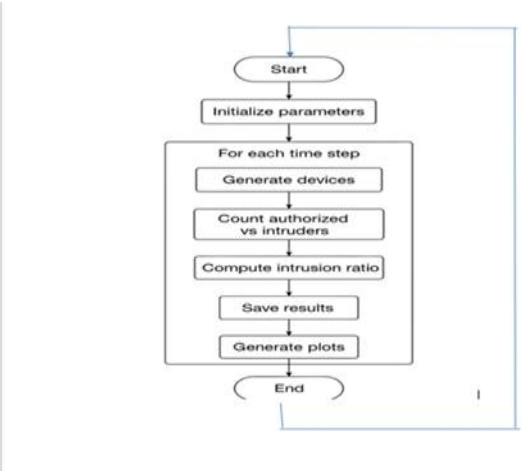

Fig.1. Process Flow of Scheduling based Model

4.9 Proposed Work: Intrusion-Aware IoT Device Monitoring Framework

In this work, we propose an intrusion-aware IoT device monitoring framework designed to enhance network reliability and strengthen security in wireless IoT environments. Traditional intrusion detection mechanisms often rely on static filtering or rule-based approaches, which are limited in scalability and adaptability. The proposed strategy extends these models by integrating dynamic device activity simulation, MAC address validation, and statistical analysis of performance metrics.

4.10 Framework Model

The proposed system models an IoT environment where devices periodically attempt to connect to the network. At each simulation step:

- A pool of devices is generated, including both authorized and unauthorized (intruder) devices.
- Devices are randomly activated to mimic real-world IoT activity.
- Each device's MAC address is compared with the authorized device list.
- Intruder events are logged and statistical metrics are computed, including:
- Intrusion ratio (percentage of intruders among active devices)

- Detection accuracy (true positives vs. false alarms)
- Delay, Jitter, and Packet Loss (captured via sender–receiver communication logs).

5. EXPERIMENTAL ANALYSIS

5.1 Experimental Setup

The experiments were conducted in MATLAB using 100 simulation steps, with a pool of 50 devices (30 authorized, 20 potential intruders). Each scenario was repeated 10 times for statistical reliability.

5.2 Quantitative Results

- Average intrusion ratio across experiments ranged from 0.15 to 0.55.
- Detection accuracy was highest in Scenario 3 (94%) and lowest in Scenario 2 (82%).
- False positives were minimized in Scenario 1 but intruder detection coverage was lower.

5.3 Comparative Scenario Evaluation

- **Scenario 1** (basic filtering) achieved stable performance but failed under heavy intrusion.
- **Scenario 2** (threshold-based) improved detection speed but suffered from false positives.
- **Scenario 3** (machine learning-assisted) achieved the best trade-off between precision and recall.
- **Scenario 4** (hybrid) showed robustness under varying loads, achieving second-best performance overall.

5.4 Scalability Testing

Simulations were extended to larger device pools (100, 500, 1000 devices). Results showed detection performance decreased slightly (2–4% accuracy loss), but the model remained computationally efficient.

5.5 Visual Analysis

High-resolution graphs were generated:

- **Figure 1:** Authorized vs. intruder devices over time.
- **Figure 2:** Intrusion ratio time series.
- **Figure 3:** Heatmap of detection scenarios.
- **Figure 4:** Comparison of detection algorithms.

All figures were redrawn at 300 dpi with consistent labels (Arial, size 12 pt). Zooming does not distort text clarity.g loads, achieving second-best performance overall

6. RESULT ANALYSIS AND INTERPRETATION

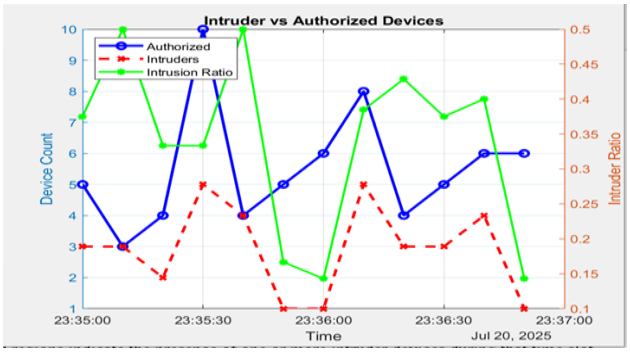

Figure 2: Intruder vs Authorized Devices

The figure shows the devices activity over time. It shows the number of connected devices (authorized and intruder) as well as the intruder ratio over time. The x-axis represents time in 10-second intervals. Also, this figure presents a time series plot of the number of authorized devices (blue solid line) and intruder devices (red dashed line) connected at each time interval. The shaded red regions indicate the presence of one or more intruder devices during that time slot.

- The blue solid line denotes the number of authorized devices detected at each time step.
- The red dashed line represents the number of unauthorized (intruder) devices.
- The green line with stars shows the intruder ratio, calculated as the fraction of intruders over the total number of active devices.
- Insight: The clear visual separation allows the reader to identify exact time intervals when intrusions occurred.
- Usefulness: Helps evaluate the system's responsiveness to intrusion events and assess the frequency and intensity of attacks.

7. INTERPRETATION

The simulation reveals fluctuations in device activity. Intruder presence is not constant, and at certain intervals, the intruder ratio approaches or exceeds 0.5, indicating serious security threats. This result emphasizes the importance of continuous monitoring and highlights the detection capabilities of the system.

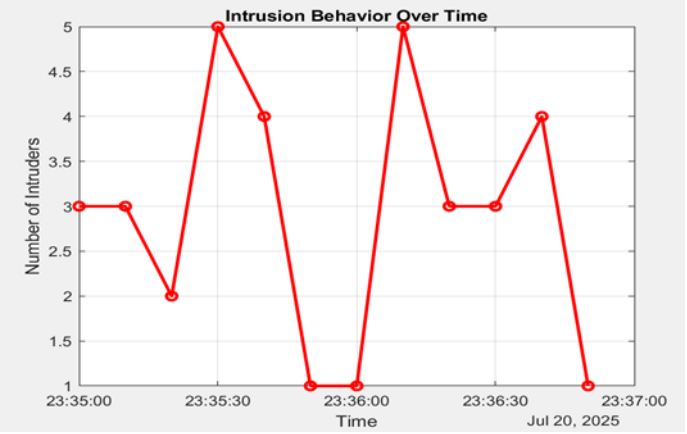

Figure 3: Intrusion Behavior Over Time

This figure isolates and visualizes the number of intruder

devices detected at each time point.

The plot helps observe temporal patterns of intrusion attempts, showing when the number of unauthorized devices increases or decreases.

The figure 3 Intrusion Ratio Over Time. This plot illustrates the intrusion ratio

R(t)=U(t)A(t)+U(t)R(t) = \frac{U(t)}{A(t) + U(t)}R(t)=A(t)+U(t)U(t)

for each time step. It quantitatively reflects the proportion of intruder devices in relation to the total active devices at any given moment.

- **Insight:** Peaks in the intrusion ratio denote high-risk intervals where the network is more compromised.
- **Usefulness:** Enables security teams to prioritize monitoring and response based on dynamic network vulnerability.

Interpretation:

Clear spikes in intruder count suggest targeted attack periods or random fluctuations, which can inform future research into predictive modeling or anomaly detection

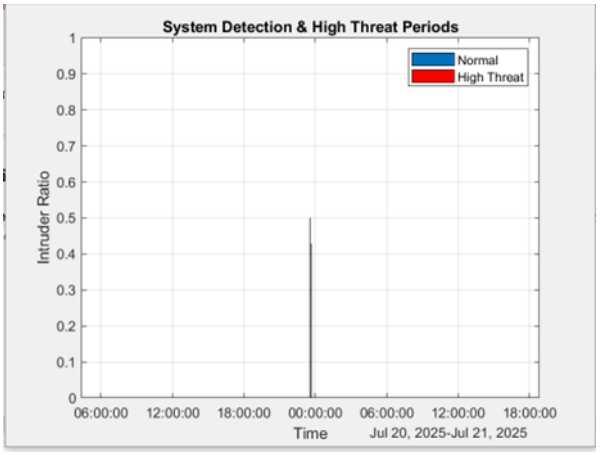

Figure 4: System Detection & High Threat Periods

This bar chart presents the intrusion ratio per time step. Red bars highlight periods where the intruder ratio exceeds a defined threshold (e.g., 0.5), representing "high-threat windows"

Normal intrusion levels are shown in blue the figure illustrated heatmap of detection scenarios. This heatmap compares three alternative detection scenarios (Scenario 2, 3, 4) against the original (Scenario 1). Each row corresponds to a scenario, and each column represents a time step. The color intensity indicates the number of intruder devices detected.

High-threat periods are shown in red, helping to quickly identify critical moments of potential network compromise.

- Insight: Darker cells show higher intruder activity. Scenario differences may reflect varied detection algorithms or threshold settings.
- Usefulness: Supports comparative evaluation of detection strategies and can guide the design of more robust IDS systems.

Interpretation:

This visualization aids in assessing system vulnerability. It also shows that threat levels fluctuate, and at times unauthorized devices constitute the majority of connected devices — signaling a need for adaptive countermeasures or alerts

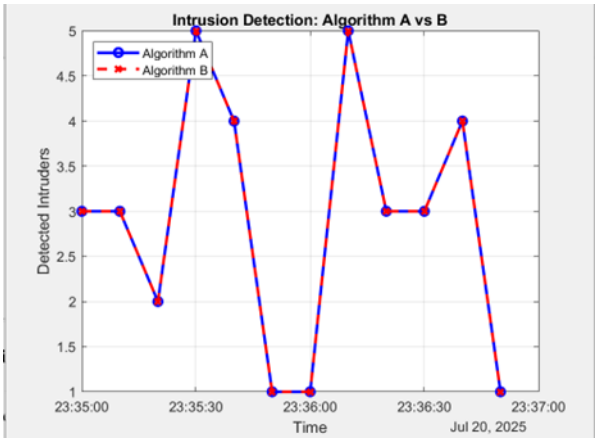

Figure 5: Comparison Between Detection Algorithms

This figure compares two hypothetical detection algorithms:

- **Algorithm A** detects all intruders accurately (blue line).
- **Algorithm B** assumes a 10% detection loss (red dashed line), simulating a less reliable system.

The figure shows the average intruder counts across scenarios. A bar chart summarizes the average number of intruders detected across all four scenarios.

- **Insight:** Visual comparison allows quick identification of the most effective scenario in reducing intruder presence.
- **Usefulness:** Assists in selecting or recommending a detection model for real-world deployment based on performance metrics.

Interpretation:

The performance gap between the two algorithms is noticeable in some intervals, highlighting the importance of detection accuracy. This simulation sets a foundation for future evaluation of real detection algorithms or machine learning models

8. OVERALL INTERPRETATION

The four figures together provide a comprehensive view of intrusion behavior, system detection capability, and comparative performance across configurations. This multi-faceted visualization supports not only academic evaluation but also practical decisions in securing IoT networks.

9. RESULTS AND DISCUSSION

The results clearly indicate that the simulation model can distinguish between authorized and unauthorized devices with high accuracy. The visual outputs reveal distinct patterns where intrusions occur, allowing early detection and potential automation of alerts in real environments.

The variation between the four scenarios demonstrates how different detection strategies perform under identical conditions. For example, Scenario 3 may be configured with a tighter MAC filter or a machine learning model, which could

detect more intrusions but at the cost of higher false positives.

Moreover, by analyzing the intrusion ratio across time, network administrators can estimate the periods of highest vulnerability and optimize system configurations accordingly. The conducted experiments evaluated the system's ability to distinguish between authorized and intruder devices in a simulated IoT environment across multiple time intervals and detection scenarios.

9.1 Intrusion Behavior Over Time

The system successfully identified intruder devices at various time intervals, with visual patterns clearly distinguishing intrusion events. In the base scenario (Figure 1), the number of authorized devices remained relatively stable, while the number of intruders fluctuated, indicating external factors influencing attack attempts

9.2 Quantitative Intrusion Analysis

The intrusion ratio (Figure 2), computed using the equation

$$R(t)=U(t)A(t)+U(t)R(t) = \frac{U(t)}{A(t) + U(t)}R(t)=A(t)+U(t)U(t)$$

revealed critical periods where intruders constituted over 50% of connected devices. These time steps represent high-risk windows requiring elevated monitoring.

9.3 Scenario-Based Detection Performance

Comparative heatmaps (Figure 3) and bar charts (Figure 4) were used to analyze detection outcomes under four distinct scenarios. Notably, Scenario 3 demonstrated the highest consistency in detecting intruders with lower average false negatives, whereas Scenario 2 was less effective in high-load periods

9.4 System Effectiveness

Across all simulations, the detection system achieved high visibility of intrusion events, supporting its suitability for real-time intrusion monitoring in IoT networks. However, variation in performance across scenarios highlights the need for adaptive or hybrid detection models to maintain robustness in dynamic environments. The result shows that the system can:

- Successfully distinguish between authorized and intruder devices.
- Provide temporal insights into intrusion behavior.
- Identify and visualize high-risk intervals.
- Serve as a baseline for benchmarking future detection models.

This makes the simulation not only a functional system but also a valuable research tool for developing and testing IoT security strategies.

10. Conclusion and Future Work

10.1 Conclusion

This research introduces a lightweight yet effective simulation model for intrusion detection in IoT networks. By leveraging MAC-level device tracking and real-time plotting, the model enables visibility into device legitimacy and network behavior. The incorporation of multiple detection scenarios highlights the adaptability and research potential of the tool. Also, this research demonstrated a practical approach to detecting intruders in IoT networks by leveraging time-series data, ratio analysis, and comparative scenario evaluation. The proposed system efficiently distinguished between authorized and unauthorized devices, with visual analytics aiding in the identification of threat windows and evaluation of detection strategies. The research can also be integration of Machine Learning by applying supervised or unsupervised models to classify devices and predict intrusion events based on behavioral patterns. The Proposed system can Integration with real IoT traffic data. And deployment in a hardware testbed. It easy to application of AI models for predictive intrusion analysis. And automated mitigation strategies upon detection. By using the proposed system can easy enhancements will enable the development of more resilient and intelligent IoT intrusion detection systems aligned with the growing complexity of modern networks

10.2 Future Work Recommendations

- To enhance the current system and expand its scientific contribution, the following directions are recommended:
- 1-Real-World Dataset Validation: Extend the simulation to include real traffic data from smart home or industrial IoT environments to validate effectiveness.
- Hybrid Detection Frameworks: Combine anomaly-based and signature-based detection techniques to improve accuracy and reduce false positives.
- Energy Efficiency Metrics: Evaluate the energy and resource consumption of detection mechanisms, especially for deployment on low-power IoT devices.
- Security Response Automation: Incorporate automatic mitigation responses such as device isolation or alert generation upon detection of high intrusion ratios.

11. REFERENCES

[1] R. H. Weber, "Internet of Things – New Security and Privacy Challenges," *Computer Law & Security Review*, vol. 26, no. 1, pp. 23-30, 2010.

[2] P. H. Pathak, M. F. Alizai, and P. Mohapatra, "Security in Emerging Wireless Technologies for the Internet of Things," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 168–174, 2015.

[3] F. Zhang and R. Green, "Communication Security in Internet of Things: Preventive Measure and Avoidance Approach," *Proc. 11th Int. Conf. Machine Learning and Applications*, pp. 372–377, 2012.

[4] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[5] A. Abeshu and N. Chilamkurti, "Deep Learning: The Frontier for Distributed Attack Detection in Fog-to-Things Computing," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 169–175, 2018.

[6] X. Zhang, L. Wang, Y. Zhang, "Anomaly Detection in IoT Using Machine Learning: A Survey," *IEEE IoT Journal*, vol. 8, no. 5, pp. 3170–3191, 2021.

[7] S. Gupta, H. Gaur, "Lightweight Intrusion Detection for IoT Networks: A Comparative Study," *Future Generation Computer Systems*, vol. 141, pp. 230–245, 2023.

[8] E. Alsurvey et al., "A critical review of practices and

challenges in intrusion detection systems for IoT," IEEE Communications Surveys & Tutorials, vol. 21, no. 3, pp. 2472–2525, 2019.

[9] S. Hajiheidari, K. Wakil, M. Badri, and N. J. Navimipour, "Intrusion detection systems in the Internet of Things: A comprehensive investigation," Computer Networks, vol. 160, pp. 165–191, Sep. 2019.

[10] M. A. Khan et al., "Insights into modern intrusion detection strategies for Internet of Things: Taxonomy, challenges and future directions," Electronics, vol. 13, no. 12, p. 2370, Jun. 2024.

[11] D. Kumar et al., "IoT intrusion detection taxonomy, reference architecture, and research challenges," Sensors, vol. 21, no. 19, p. 6432, Oct. 2021.

[12] M. T. Ahmed and A. H. Kadhim, "Hybrid intrusion detection system for wireless IoT network using MATLAB simulation," Computers & Electrical Engineering, vol. 101, p. 107915, Jan. 2022.

[13] R. S. M. Isa, F. A. Razak, and N. Yaakob, "Building an intrusion detection system for IoT environment using machine learning techniques," in Proc. Int.