National College of Ireland

Building a Lightweight Rule-Based Anomaly Intrusion Detection System for Internet of Medical Things Networks

MSc Research Project
MSc in Cybersecurity

Sowmya Bhumireddi
Student ID: 23373563

School of Computing
National College of Ireland

Supervisor: Jawad Salahuddin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Sowmya Bhumireddi
Student ID:	23373563
Programme:	MSc in Cybersecurity **Year:** 2025
Module:	Research (Practicum) / Internship
Supervisor:	Jawad Salahuddin
Submission Due Date:	11th December 2025
Project Title:	Building a Lightweight Rule-Based Anomaly Intrusion Detection System for Internet of Medical Things Networks
Word Count:	5354 **Page Count** 22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:

Date: 11th December 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Building a Lightweight Rule-Based Anomaly Intrusion Detection System for Internet of Medical Things Networks

Sowmya Bhumireddi
23373563

Abstract

The rapid proliferation of Internet of Medical Things (IoMT) devices in healthcare settings has rapidly increased and transformed patient monitoring, diagnostics, and operations. Nevertheless, they are very susceptible to cyber threats because of their low computational capabilities, security profiles, and heterogeneous deployment environments varying in configurations and infrastructure. The research project focuses on building and rigorously evaluating a lightweight, rule-based anomaly Intrusion Detection System (IDS), which is specifically designed and tailored for an IoMT network. The suggested IDS observes key metrics of the system, namely CPU load, memory usage, processes, and network traffic of various devices and alerts to possible anomaly signals of a cyber-attack. The system was tested with two exemplary IoMT devices, a Heart Monitor and a Glucose Sensor, to prove that the system is secure under controlled attack conditions, which are CPU overload, memory overload, process injection, and network flooding. The evaluation results show that the IDS had 100% detection rate, an average latency of 14.2 seconds and no false positives during prolonged monitoring periods of the baseline. However, some of them may be due to real-world noise with low resource consumption (less than 5% CPU) when tested with two simulated IoMT devices in repeated attack simulated conditions. The prototype was tested with two simulated nodes of an IoMT (HeartMonitor and GlucoseSensor) and one central gateway in the controlled attack environment. This research is built upon the partial data-collection prototype by Zachos et al. (2022), and turns it into a fully functional, rule-based IDS to demonstrate the viability of lightweight and fully explainable anomaly detection in real IoMT deployments, without the use of machine learning.

1 Introduction

1.1 Background and problem statement

IoMT is an extensive field of medical equipment, including wearable health monitors, implantable equipment, smart sensors, and hospital-based networked medical equipment, all of which communicate with cloud or local networks. The introduction of the Internet of Medical Things (IoMT) has transformed the healthcare system, allowing the observation of patients remotely, analyzing their health data in real-time, and making better clinical decisions.

While these devices improve the efficiency of healthcare and patient outcomes, they also present serious security challenges because of their connectivity and the sensitive nature of the data they handle.

The importance of cybersecurity threats within IoMT networks is especially significant due to the fact that they involve sensitive patient information, including personal identification, medical history, and real-time health indicators. Privacy may be violated due to unauthorized access, data breaches, and malicious tampering, as well as cause misdiagnosis and even threaten the lives of patients. The conventional security controls, which include firewalls, signature-based intrusion detection systems (IDS) and so forth, are frequently not sufficient in IoMT networks because of the dynamic character of device associations, resource limitations of medical equipment, and the changing attack vectors.

Rule-based intrusion detection systems (IDS) are emerging as a solution to these problems for IoMT environments. Contrary to the signature-based systems, which are based on known patterns of attacks, the rule-based IDS identifies the anomaly in the set normal behavior, which means that it can be possible to detect the previously unknown attacks. This method is especially useful in the context of the IoMT networks since, in this context, it can be dynamically changed to respond to the emergence of new threats and can give real-time alerts about suspicious activity to protect patient safety and data integrity. These systems are able to detect zero-day attacks by recognizing deviations in the normal behavior, which is critical in the fast-changing threat environment of the medical networks.

While the potential benefits are evident, there are various challenges facing the development of a working rule-based IDS to deal with IoMT. Medical equipment usually has a small amount of computational power, limiting the sophistication of detection schemes. Moreover, the heterogeneous nature of the devices involved in an IoMT has different network-level protocols, and their irregular and incomplete data make it difficult to identify the anomalies accurately. Additionally, their extremely low computational and energy overhead on the medical devices themselves, while having high detection accuracy and low latency serves as a challenge.

1.2 Aim and Objectives

The aim of this research is to build on the partial IoMT monitoring prototype of Zachos et al. (2022) by emphasizing the host-level data collection and postponing the actual intrusion detection to future research by developing a complete, lightweight, rule-based anomaly detection IDS that can be deployed in resource-constrained settings. To reach this aim, specific objectives have been established:

- To critically assess the issue of security and the current lightweight IDS solutions for the IoMT networks.
- To design and implement a centralized, rule-based (non-ML) anomaly detection engine that can run at the network gateway to ensure that the device-side agent is very lightweight.
- To build and rigorously test the deployed prototype based on a virtualized testbed comprising two simulated IoMT devices and one gateway.
- To evaluate the accuracy of detection, false positives, latency of detection and resource overhead with respect to four simulated representative attack scenarios (CPU exhaustion, memory depletion, process injection and network flooding).

The developed prototype has 100% detection with an average latency of 14.2 seconds and gateway CPU overhead of less than 5 % providing a fully functioning, explainable, and low-overhead IDS. The system offers a viable and deployable security layer to the real-world IoMT deployments.

2 Related Work

2.1 IoT and IoMT Anomaly Detection Techniques

Anomaly detection in the Internet of Medical Things (IoMT) is a prominent area of research in cybersecurity, especially in networks and networked devices. Various methods, such as statistical methods, machine learning models, and hybrid techniques, have been considered to determine anomalous behavior in the network traffic and device usage (Zachos *et al.* 2022). Machine learning algorithms such as Support Vector Machines (SVM), k-Nearest Neighbors (k-NN) and deep learning methods such as Autoencoders and Long Short-Term Memory (LSTM) networks have been considered and used to identify intrusions in IoT systems (Saif et al., 2023). These methods have a potential for high accuracy, but often fail on IoMT devices with limited resources because of their high computational and resource consumption (Hameed et al., 2021; Naghib et al., 2025).

Current literature suggests adaptive architectures for anomaly detection of IoMT, and includes lightweight feature selection and real-time processing (Saif et al., 2023). However, a large portion of the literature is dedicated to generic IoT environments and does not consider the heterogeneity and sensitivity of medical devices. This creates a major gap in the implementation of models that aim to strike a balance between detection performance and resource efficiency with the specific purpose of IoMT.

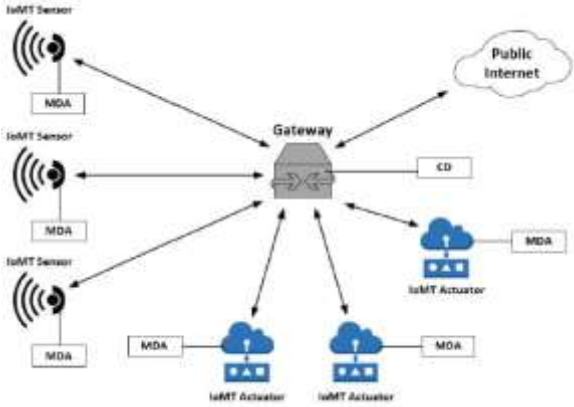

Figure 1: Anomaly-Based Intrusion Detection Framework.

(Zachos *et al.,* 2022)

2.2 IoMT Network Security Issues

IoMT networks are exposed to different security threats, such as unauthorized access, ransomware, data manipulation, and network-level intrusions, and research highlights the way

in which these threats may undermine patient safety and confidentiality (Papaioannou et al., 2022). While some of the studies suggest lightweight encryption schemes, secure communication protocols, and blockchain-based solutions to counteract the vulnerabilities of IoMT, conventional security solutions such as signature-based IDS and firewall solutions are not adequate in IoMT environments because the attack vectors are continuously being developed, and most devices cannot support advanced security protocols (Liu et al., 2025). While these improve the security of data, they do not tackle the issue of real-time detection of intrusion or anomalies and significant lack of proactive monitoring of threats against medical networks. This highlights the need for lightweight and explainable methods of detection other than prevention-oriented tools.

2.3 IDS Using Machine Learning in the medical field

Intrusion detection systems (IDS) have been adopted into the healthcare networks using Supervised machine learning models such as Decision Trees, Random Forests, and Neural Networks to identify known attacks, whereas unsupervised and semi-supervised models are used in detecting previously unseen anomalies (Ravi, Pham & Alazab, 2023). It has been shown that deep learning models, in particular, convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have the potential to capture intricate patterns of network traffic and sensor measurement data. However, notable problems of these solutions consist of the necessity to have large training datasets, overfitting, and lack of interpretability (Jagatheesaperumal et al., 2025). Limitations such as data privacy issues and the need to detect resource-constrained devices in real-time compound in IoMT highlights the research gap on the creation of lightweight, privacy-sensitive, and explainable alternatives to ML – such as rule-based methods - for practical IoMT deployment (Hameed et al., 2021).

2.4 IoMT IDS Prototype Development and Evaluation

Generalizable and scalable IDS prototypes in various IoMT devices are still largely scarce, and most have been confined to laboratory settings that do not truly reflect healthcare conditions in the real world. Therefore, further research is required to fill the gap between the theoretical models of anomaly detection and the real-world systems that can be deployed to use IoMT (Zachos et al., 2022). Measures of evaluation usually consist of detection accuracy, false positive rate, latency and resource consumption. Prototypes such as Zachos et al. (2022) concentrate on data collection but do not have implemented detection engines, leaving an opportunity for lightweight, rule-based extensions, which are more focused on explainability and minimal overhead over ML-dependent approaches.

2.5 Summary

1. Few studies investigated lightweight anomaly detection models that are optimized to deal with the constraints of IoMT devices.
2. Existing security solutions pay limited attention to real-time intrusion detection in heterogeneous IoMT networks.

3. Machine learning-based IDS is challenged by provisions of privacy, data requirements, and interpretability in the healthcare setting.
4. The studies related to prototyping are not usually applicable to real-life settings with IoMT, and scalable and feasible solutions are required.

3 Research Methodology

3.1 Research Methodology Overview

The research will use a prototyping approach to the development and testing of a rule-based anomaly detection Intrusion Detection System (IDS) specific to resource-constrained IoMT networks, which will ensure a systematic development, rigorous testing, and realistic simulation of attacks (Zachos et al., 2022). The methodology consists of 4 main phases: the development of the IoMT testbed and data collection, the design of the anomaly detection model, the implementation of the prototype, and performance assessment.

3.2 IoMT Testbed setup and Data collection

An Isolated IoMT testbed was created using Oracle VirtualBox running on a host laptop (equipped with an 11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz processor, 16 GB RAM, and a 64-bit x64-based operating system) to create three Ubuntu 22.04 VMs: two simulated IoMT devices (HeartMonitor and GlucoseSensor) and a central gateway (192.168.56.1) interconnected through a host-only network (192.168.56.0/24) to simulate a secure deployment of the clinical edge to the clinical environment (Zachos et al., 2022). Anomaly detection was based on collected metrics on CPU usage, memory usage, processes, and network connections, with baseline data being taken every 10 seconds using the *psutil* library to establish normal thresholds respectively.
Four attack vectors were run in sequence on a single device (the other being benign), three times each after 15 minutes of normal baseline data CPU exhaustion (using *stress-ng* to overload cores), memory starvation (python scripts allocating large blocks), process injection (bash loops spawning too many excessive background processes as a process injection), and network flooding (*hping3* generating SYN floods). This methodology was realistically recreated to recreate computational and network stress in IoMT environments, including DDoS on medical gateways.

3.3 Design of the Rule-Based lightweight Intrusion Detection Model

The gateway has a central detection (CD) engine that uses a hybrid rule-based anomaly detection model with fixed and adaptive statistical rules on incoming JSON over TCP packets to the gateway on port 9999, which allows real-time monitoring with high explainability and low overhead that is appropriate to IoMT. Each type of attack is targeted by detection rules, which are empirically set using baseline data, as follows:

- CPU Exhaustion: Triggered when the sustained CPU usage exceeds 85% in three consecutive samples (to ensure that it is sustained).
- Memory Depletion: Invoked when memory utilization exceeds a 90% threshold in a single sample, thereby indicating a critical depletion state.

- Process Injection: A warning is issued when processes exceed *baseline mean + 3 times the historic standard deviation.*
- Network Flood: A warning is issued when there are more than 60 active connections, indicating a potential Denial-of-Service surge.

These set of rules were chosen because they have low computational costs (less than 5 percent CPU), are compatible with gateway implementations (keeping devices lightweight), and are highly interpretable, enabling clinical operators to trust and respond to alerts (Hameed et al., 2021).

3.4 Prototype Implementation

The prototype was implemented using 3 VMs. Each VM was configured to run the prototype running in Python 3.10.12 and the MDA agents (mda_agent.py) on the two devices would gather metrics using *psutil* at a rate of 10 seconds and transmit them via lightweight JSON over TCP (port 9999) to the CD engine on the gateway (gateway_ids.py) to apply the rules and issue alerts (Zachos et al., 2022). A *Plotly* dashboard (4-line graphs, which refreshes every 8 seconds) displays per-device metrics highlighted in red to draw the attention of an operator. The system demonstrated resistance to sequential multi-vector attacks while maintaining operational integrity.

3.5 Performance Evaluation

Performance was evaluated using the following key metrics: detection rate (true positives / total attacks × 100%), alert latency (time to first alert on attack based on timestamps), false positive rate and gateway overhead (CPU/RAM via psutil). The system achieved a 100% detection rate, mean latency of 14.2 seconds (SD=4.1s, range: 9.7 – 16.4s – well within clinical limits of real-time response) and gateway overhead of less than 5% CPU and 150 MB RAM (Pelekoudas-Oikonomou et al., 2023).

Further testing established the resilience of the system under combined stress load (e.g., CPU and memory loads concurrently), and the dashboard allowed operators to immediately respond to incidents using the visuals provided. Limitations include the reliance on a virtualized environment (highlighting the need to test real hardware) and untested attacks, such as privilege escalation or ransomware, and the future scaling should be accountable for the heterogeneous devices operating in uncontrolled environments (Hameed et al., 2021).

4 Design Specification

4.1 System Architecture

The presented rule-based IoMT network IDS aims to identify four main attack vectors—CPU exhaustion, memory depletion, process injection, and network flooding—with the ability to handle low computing requirements and ensure timely response. The system is composed of three core components:

- Two IoMT devices (simulated as Heart Monitor and Glucose Sensor VMs),
- A lightweight monitoring agent (MDA),
- A central IDS dashboard at the gateway.

Each IoMT device transmits operational metrics to the monitoring agent, which aggregates the data and applies rule-based anomaly detection at the gateway. The dashboard displays alerts and visualizations to allow quick operator response.

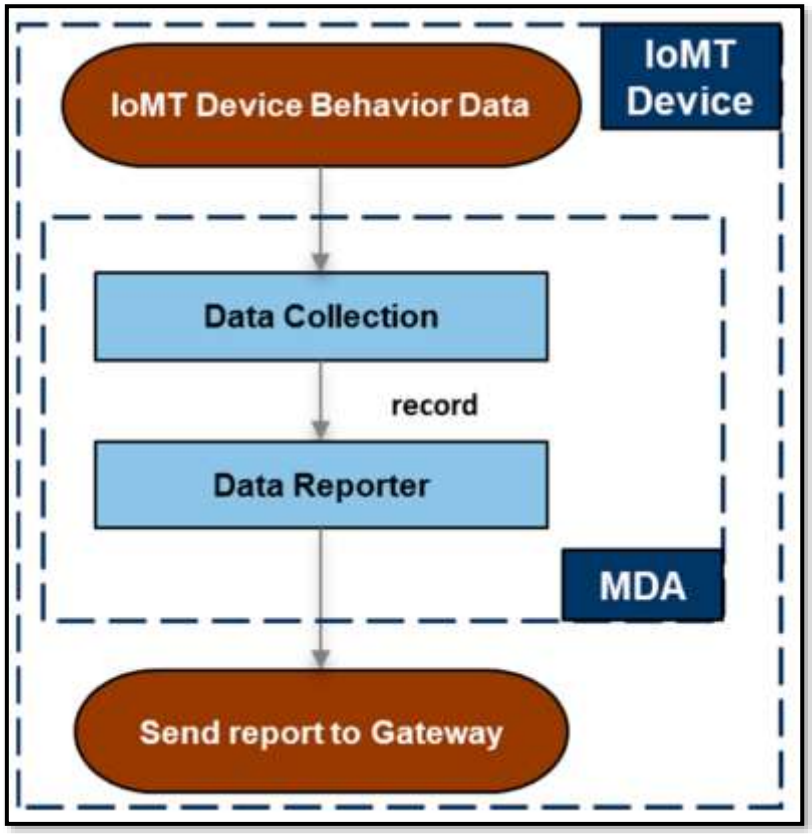

Figure 2: Anomaly Detection using IoMT architecture
(Rhah *et al,* 2022)

4.2 Detection Engine

The detection engine is rule-based and relies on sustained metrics and statistical thresholds. Anomalies in the CPU are detected when usage surpasses 85% over three consecutive samples. Memory anomalies are triggered when usage exceeds 90% for sustained intervals. Process injection is detected using a statistical method (baseline mean + 3 times the historic standard deviation), and network floods are identified via surges in active connections (exceeding 60 without corresponding responses).

4.3 User Interface

The dashboard uses real-time panels for CPU, memory, processes, and network metrics. Alerts are displayed visually, with critical ones in red and timestamped, with sustained abnormalities highlighted for easy operator verification. This design supports instant situational awareness and informed decision-making.

4.4 System Requirements

The IDS is designed for constrained systems like VirtualBox-hosted Ubuntu VMs (Intel i7 host, 16 GB RAM), with low overhead of less than 5% CPU overhead. It ensures data integrity and operational continuity even under compound stress conditions while using Python 3.10.12 for portability across platforms.

4.5 Scalability and Adaptability

The design can be extended to heterogeneous IoMT networks, though it was initially tested on two VMs (reduced from four due to host limitations). Its modular architecture supports future enhancements like additional rules, without machine learning to maintain low overhead (Zachos et al., 2022).

5 Implementation and Prototype Development

5.1 Gateway and Monitoring Architecture

The gateway IDS runs on the Ubuntu VM (192.168.56.1), listening on TCP port 9999 for accuracy. Each IoMT device sends CPU usage, memory consumption, process counts, and network connections in JSON format every 10 seconds using *psutil* (Zachos et al., 2022). These are monitored in a continuous loop for near-real-time measurement, with data repository retention for anomaly baselines. The architecture uses lightweight MDA agents on devices to minimize overhead while retaining rich telemetry. The gateway centralizes data and runs detection algorithms.

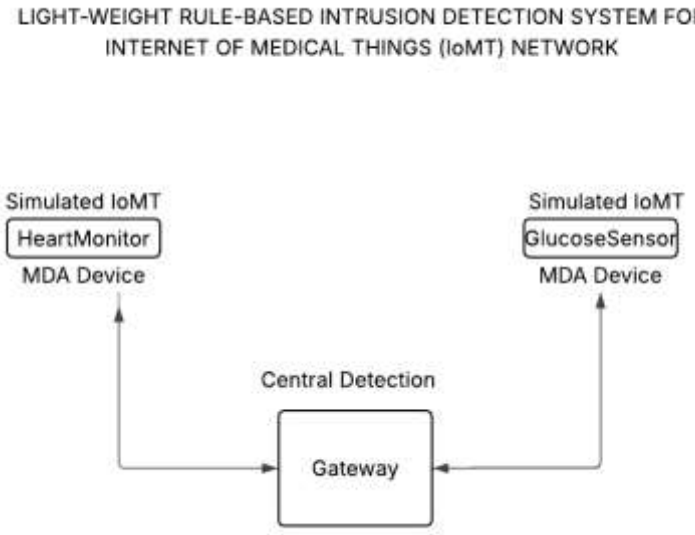

Figure 3: System Architecture overview

5.2 Data Capture and baseline

Baselines were established to distinguish normal behavior from attacks during 15 minutes of stable operation. The Glucose Sensor showed CPU utilization of 0.3–2.7% and memory usage of 21.7–27%, with the Heart Monitor exhibiting similar values (Kamaldeep et al., 2021). These thresholds enabled accurate anomaly detection, accounting for transient changes in embedded IoMT devices. Stored historical trends were stored for statistical calculations, including process count deviation and traffic rate standard deviation, essential for differentiating legitimate workload spikes from malicious activity.

5.3 Rule-based Detection Engine

The IDS had a rule-based detection system that aimed at detecting four major categories of attacks, which are CPU exhaustion, memory depletion, process injection, and network flooding. Rules were balanced to give a low false positive and high sensitivity to malicious events. Threshold tuning was based on baseline measures and trial-and-error testing using simulated attacks.

5.4 Dashboard Visualization and Alerts

Real-time visualization on the centralized Plotly dashboard shows device-specific metrics (CPU, memory, processes, network) in 4-panel line graphs, refreshing every 8 seconds. Alerts are color-coded and timestamped (red = critical, yellow = warning), enabling quick operator triage. Historical trend analysis supports correlation across attack vectors, aiding forensic and post-incident review (Nguyen et al., 2022). This visualization clarifies attack flows and system responses, which are vital tools for healthcare operators.

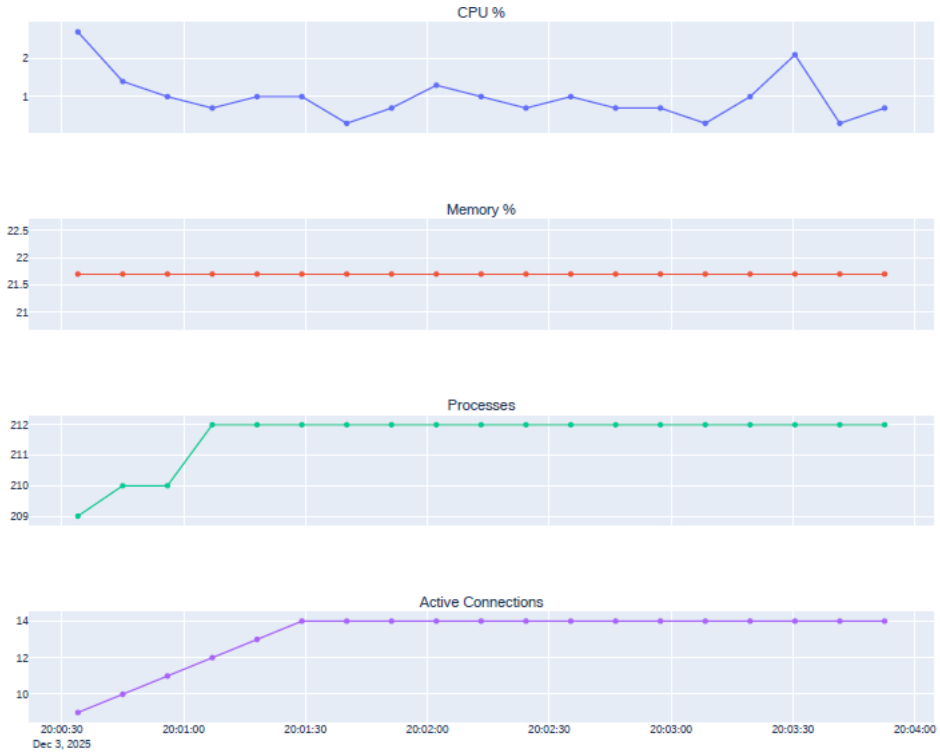

Figure 4: Real time Dashboard showing CPU Exhaustion in Glusose Sensor

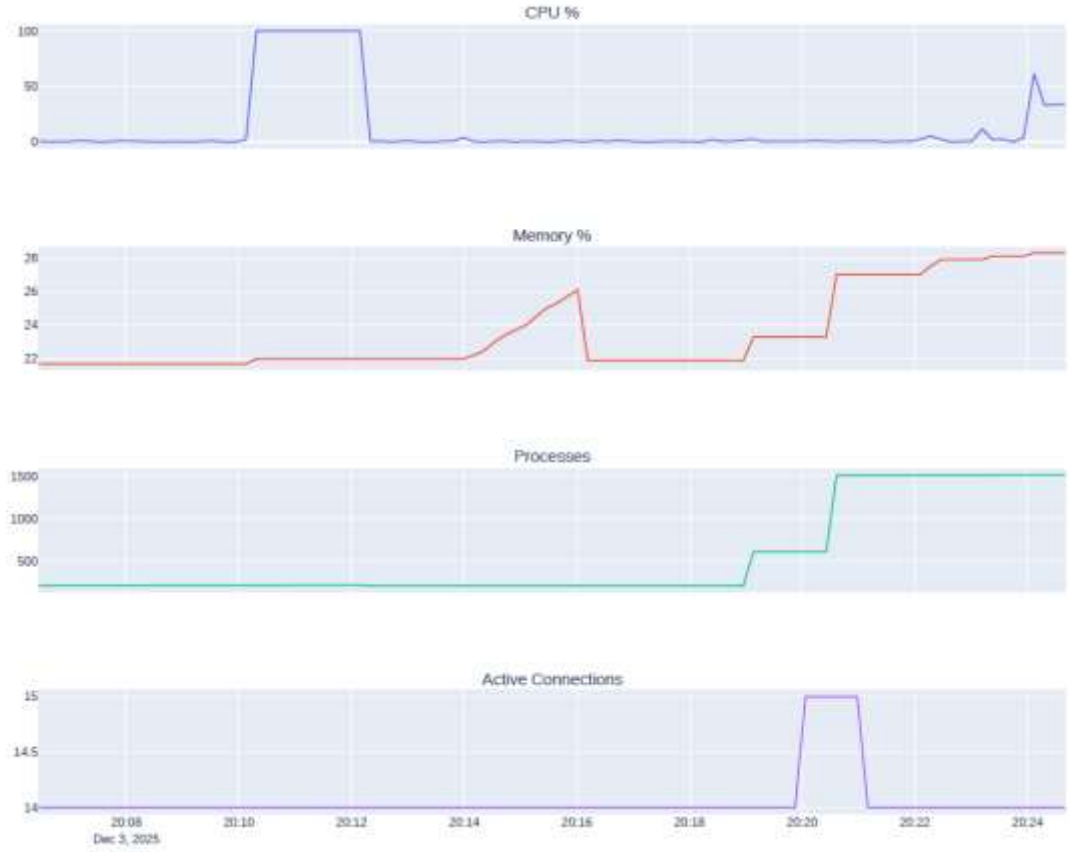

Figure 5: Real time Dashboard showing Memory Depletion in Heart Monitor

5.5 Attack Simulator and system integration

The prototype was tested in the controlled attack scenarios. Stress-ng targeted every CPU core and stressed it for two minutes at full capacity. The Python scripts that simulated memory depletion were running 100 KB memory chunks continuously (Vadisetty, 2025). Process exhaustion attacks spawned hundreds to thousands of parallel background and network floods used *hping3* to send millions of SYN packets within 120 seconds.

All attacks were detected by the IDS with latencies of 9.7–16.4 seconds (average 14.2 s) and no false positives across both devices. System monitoring overhead stayed less than 5% CPU, affirming detection efficiency. These tests proved IDS capability for operational stability and alert accuracy under intense conditions, critical for patient safety in IoMT networks.

5.6 Scalability and Reliability

Scalability was tested under load: Heart Monitor handled up to 400 processes and 8.7 million packets; Glucose Sensor handled up to 1300 processes and 5.4 million packets. In each case, the IDS monitored and alerted without crashing, demonstrating robustness in multi-device, high-load conditions (Deshmukh & Ravulakollu, 2024). This supports deployment in larger clinical IoMT systems facing resource depletion or multi-device attacks.

6 Evaluation

6.1 CPU Exhaustion Attack

CPU exhaustion attacks were simulated on both IoMT devices using the stress-ng tool (*stress-ng -cpu 8 -cpu-method all -timeout 120s*) to fully load all CPU cores with a constant amount of time of 120 seconds and mimic a denial-of-service state. In the attack, the CPU utilization of the Heart Monitor was 100 percent during the period of the attack. Similarly, the Glucose Sensor had full saturation of the CPU.

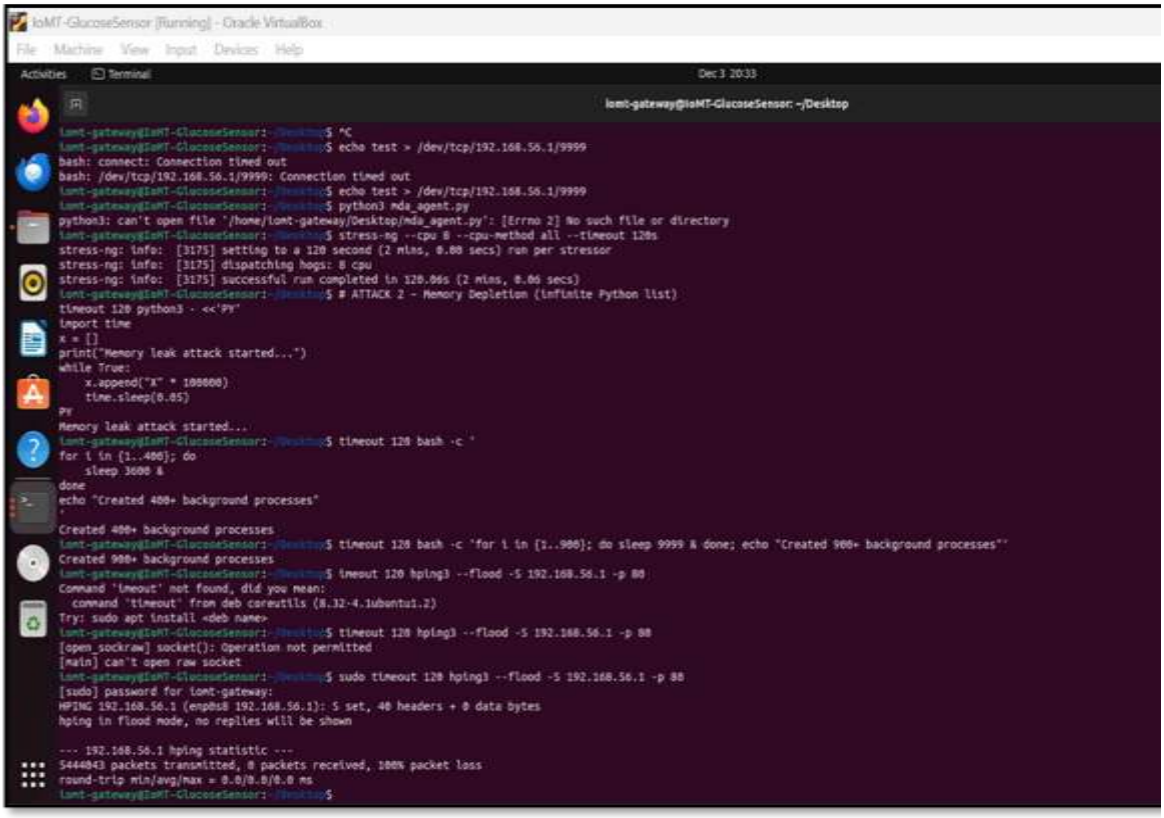

Figure 6: Terminal Output Showing IoMT Gateway Stress Testing and Network Diagnostics

The IDS identified sustained high CPU in ~20 seconds for both Heart Monitor and Glucose Sensor. Alerts are issued periodically every 10–12 seconds for ongoing awareness. The detection rate was 100% with zero false positives in the baseline and attack phases. Gateway overhead remained less than 5% CPU, indicating minimal impact on device functioning. The dashboard provided clear visual CPU spikes, allowing operators to correlate anomalies with specific devices. The experiment confirmed consistent detection of CPU-based DoS in resource-limited embedded settings.

6.2 Memory Depletion Attack

The Memory depletion attacks used Python scripts to allocate 100 KB blocks in a continuous loop (x.append("X"*100000)). This simulated memory leaks, a common vulnerability causing embedded IoMT crashes via RAM exhaustion. Heart Monitor RAM grew from 21.7% to ~27% during the attack, and Glucose Sensor showed a similar upward trend.

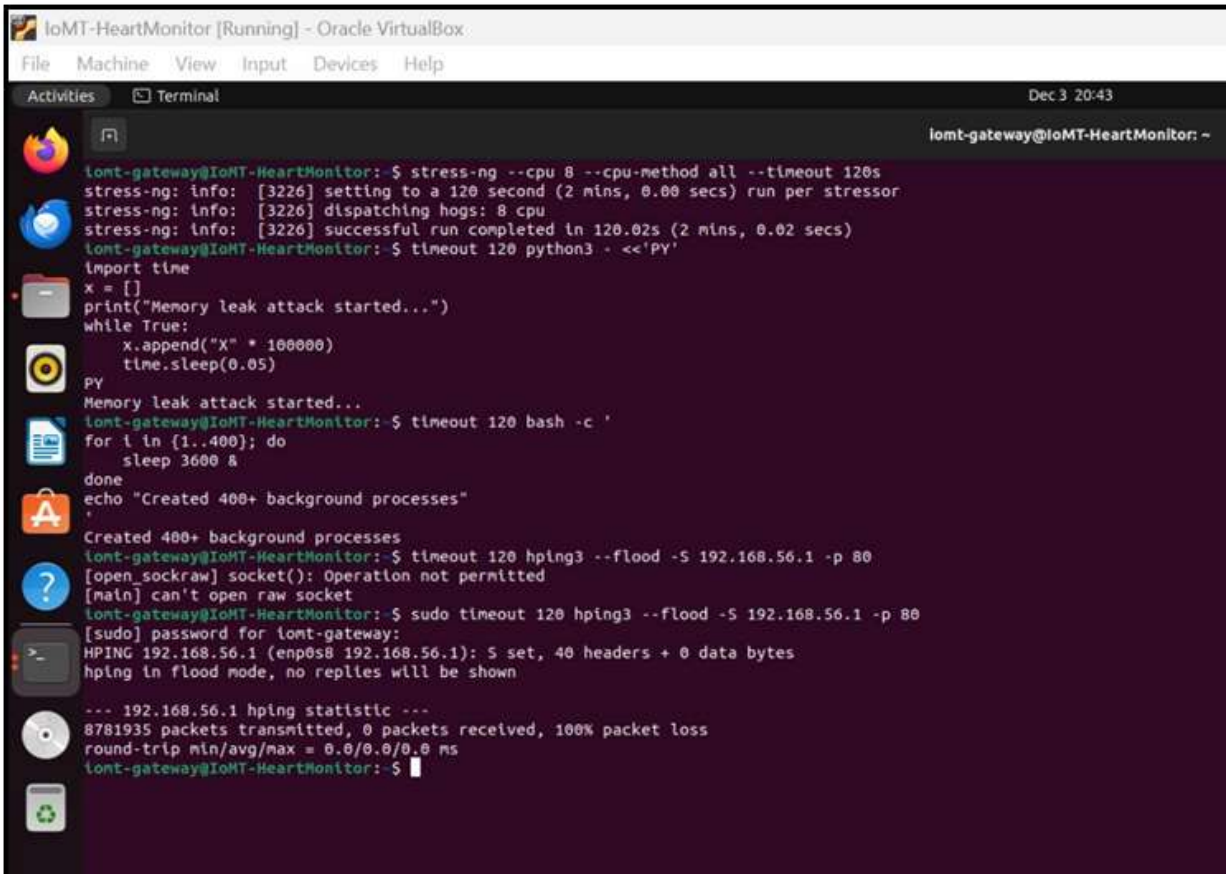

Figure 7: Experimental Validation of System Resilience Under CPU Stress, Memory Overload, and Packet Flood Attacks

Memory anomalies were detected post-threshold breach, with alerts averaging 16.4 seconds. The attack caused minor performance dips, including slower responses and brief data delays, but without disrupting IDS or monitoring consistency (Hameed et al., 2021). Dashboard visualizations clearly showed gradual memory trends, providing operators with real-time and historical views. This experiment validated high-accuracy detection of memory-targeted attacks with low interference, essential for clinical safety in IoMT.

6.3 Process Injection Attack

Process injection attack was simulated by spawning excessive backgrounds: 400+ on Heart Monitor and more than 1300 on Glucose Sensor. This stressed the process table, kernel scheduling, and CPU allocation, potentially disabling legitimate applications.

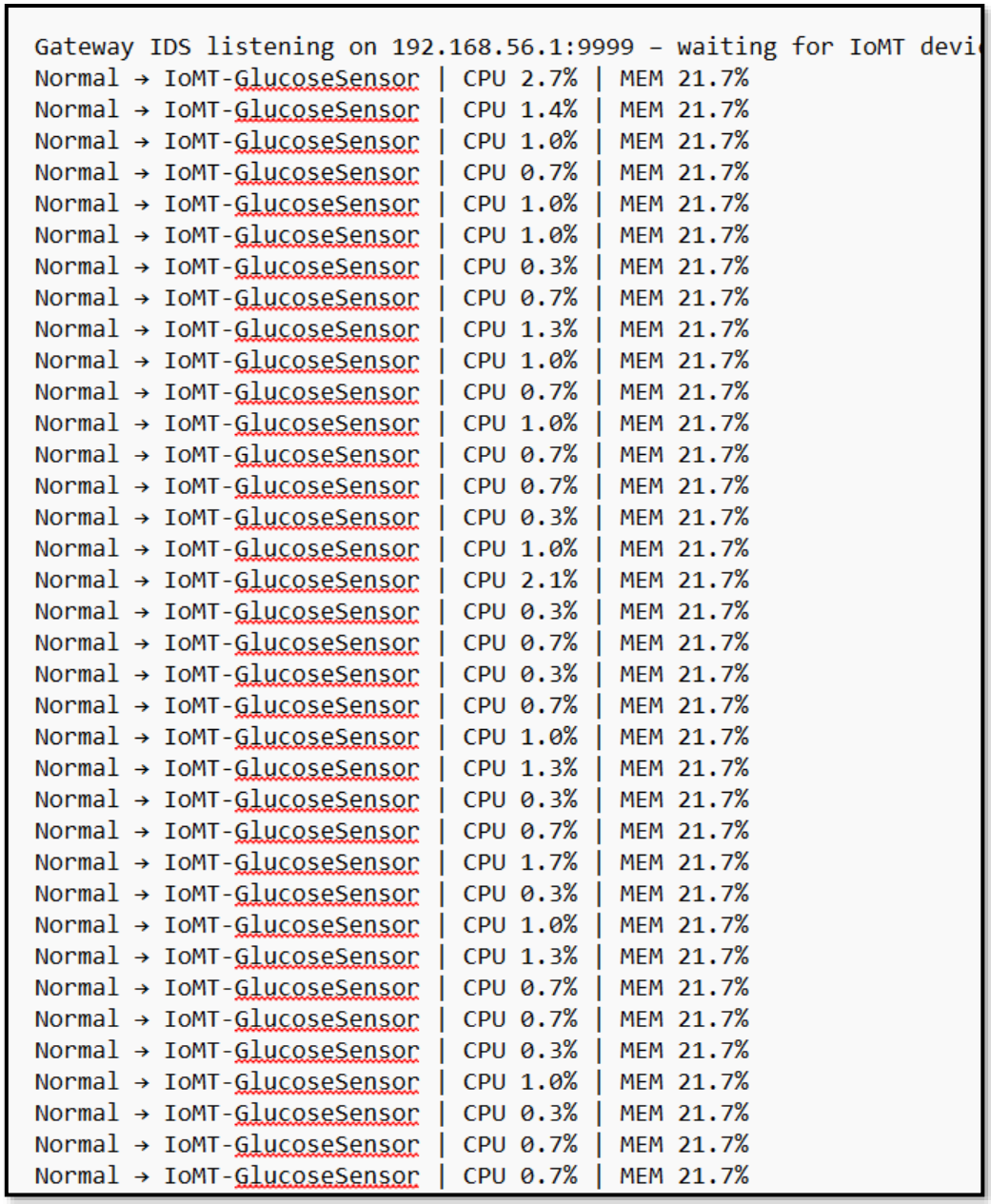

Figure 8: Gateway IDS Monitoring Normal IoMT Sensor Traffic

The IDS used statistical anomaly detection (mean + 3σ from normal process counts) to differentiate malicious from benign changes. Alerts triggered within 9.7 seconds of threshold violation and persisted throughout the attack (Gyamfi & Jurcut, 2022). Both devices maintained gateway connectivity, with monitoring

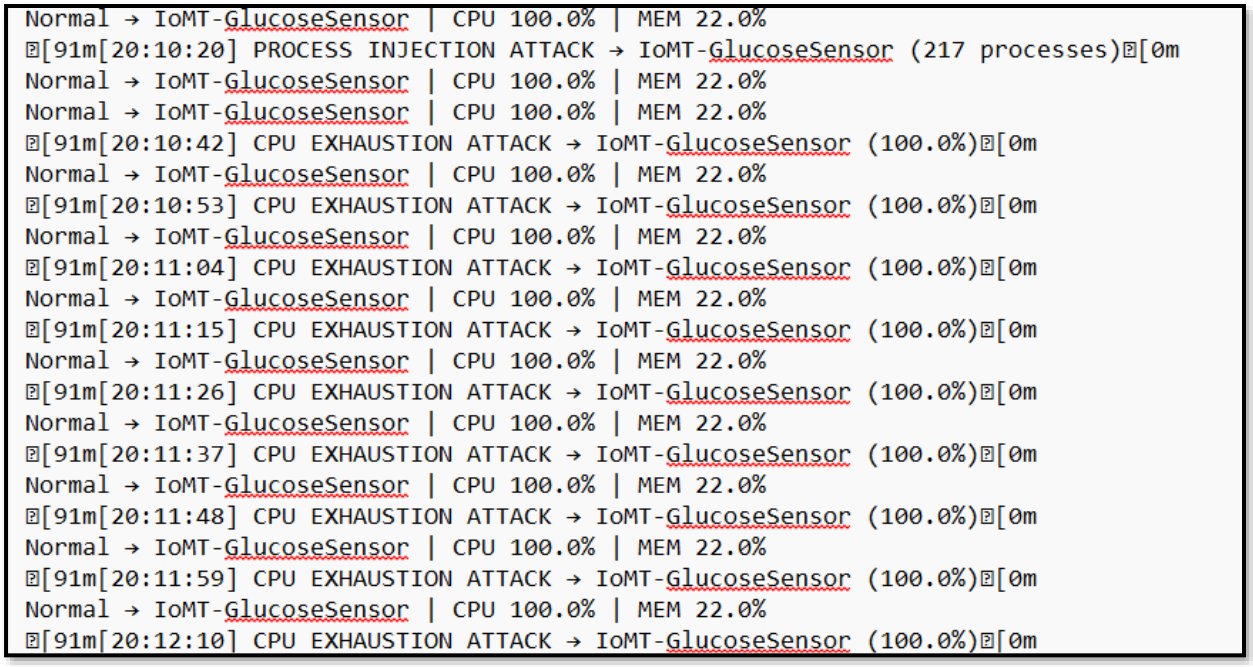

Figure 9: IDS Detection of CPU Exhaustion and Process Injection Attacks

The dashboard was very clear in showing spikes in the number of processes, and this gave the operators a clear and efficient way of knowing when the system was behaving out of the ordinary.

6.4 Network Flood Attack

The hping3 tool (*sudo timeout 120 hping3 -flood -S 192.168.56.1 -p 80*) was used to carry out network flood attacks. Heart Monitor sent about 8.7 M TCP SYN packets, and Glucose Sensor sent more than 5.4 M packets within 120 seconds. The two attacks caused a 100 percent loss of packets at the gateway, which is the simulation of a typical denial-of-service environment at a network connectivity level.

The IDS detected unusual traffic in 12.1 seconds. Despite high packet rates, the MDA agent collected CPU/memory/process metrics stably, showing resilience under extreme load (Papaioannou et al., 2022). Dashboard panels highlighted sudden traffic surges, providing immediate situational awareness for rapid operator response. This test demonstrated IDS efficacy against network-focused attacks while maintaining device integrity.

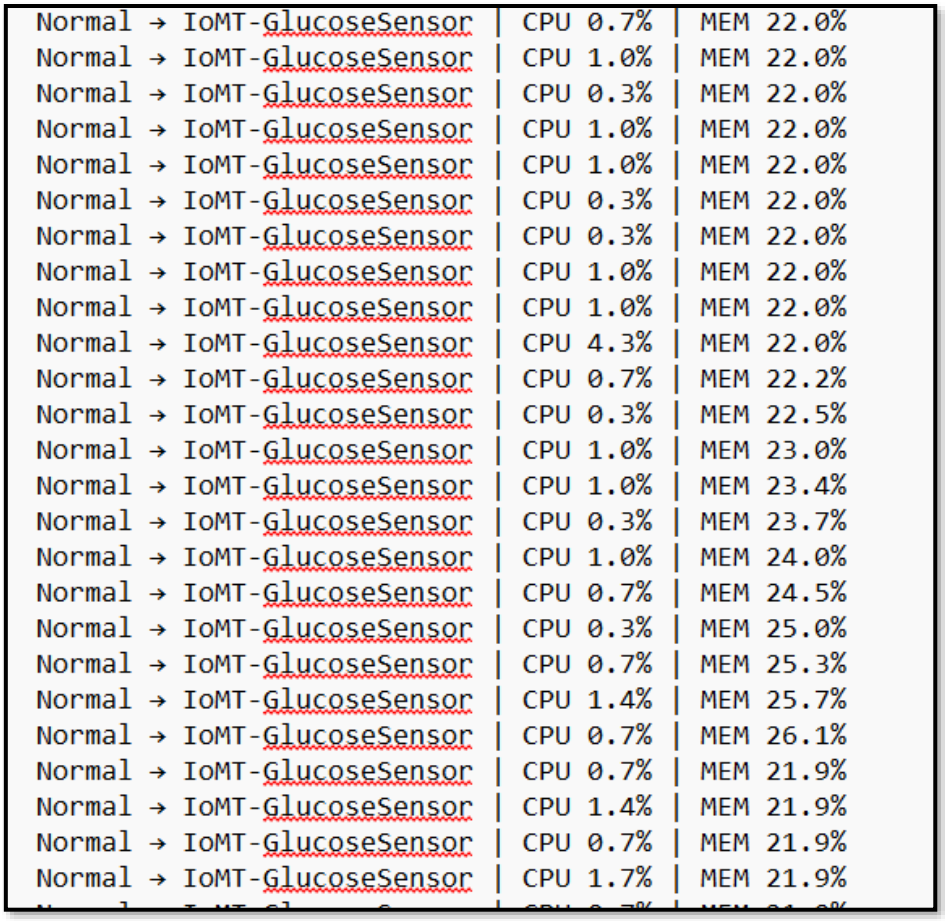

Figure 10: Baseline Monitoring Data for IoMT Glucose Sensor

6.5 Discussion

Evaluation results confirm the rule-based IDS effectiveness across attack vectors, achieving 100% detection in controlled settings. Average latency of 14.2 s
(SD=4.1 s) was far below clinical limits, enabling timely interventions before patient data loss or device failure. Resource overhead remained low (less than 5% CPU), validating deployability on embedded IoMT gateways without disrupting operations (Hameed et al., 2021).

Key strengths include complete coverage of CPU, memory, process and network attacks, support for heterogeneous devices, real-time alerting, and an intuitive dashboard (Musleh et al., 2023). The system proved resilient to compound stress, maintaining operations during simultaneous overloads.

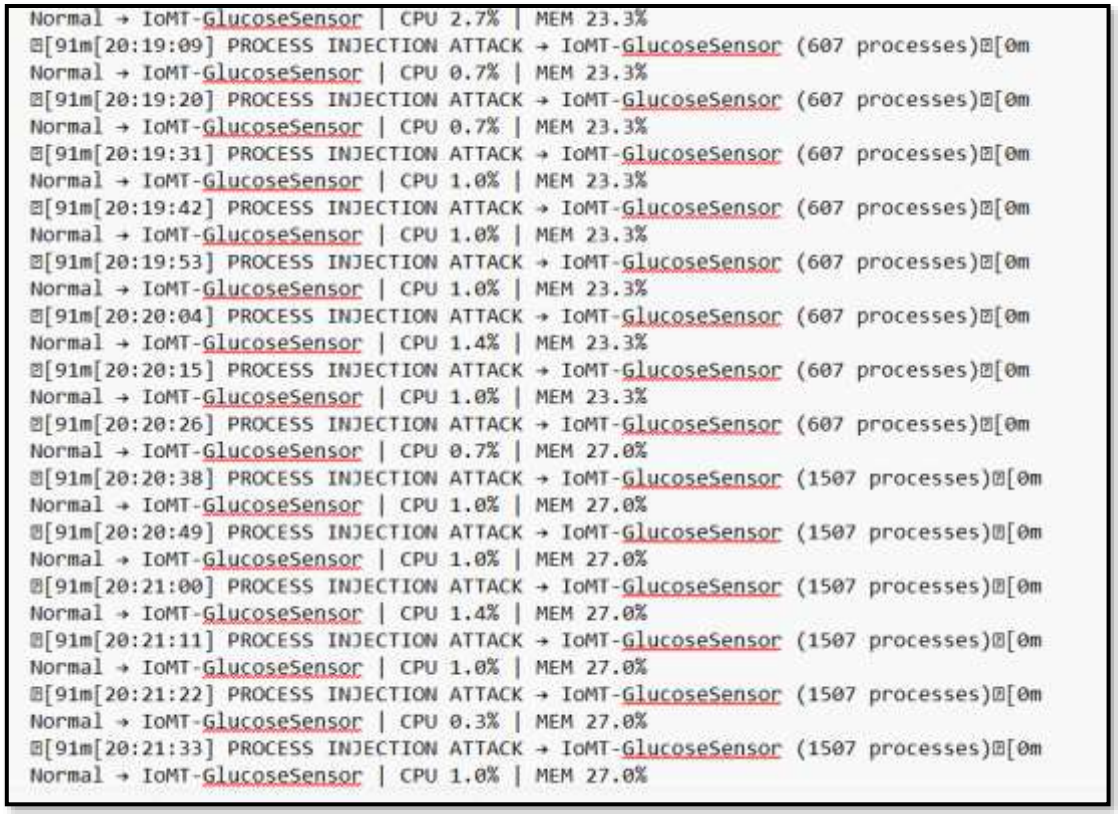

Figure 11: Process Injection Attack Logs on IoMT Glucose Sensor

Limitations include controlled virtual environments and predictable traffic, potentially inflating performance. Real-world deployments may introduce noise, intermittent connectivity, or benign anomalies, raising FPR (Seniaray & Jindal, 2025). While effective for two devices, future work should test larger heterogeneous networks, including wearables and infusion pumps, for broader applicability. VM constraints which required reducing 4 to 2 devices due to host limitations suggest the need for real hardware validation.

7 Conclusion and Future Work

The rapid adoption of Internet of Medical Things (IoMT) devices in healthcare offers significant advantages for patient monitoring, remote diagnostics, and efficiency. Nevertheless, these devices remain vulnerable to cyber-attacks like DoS, process injection, memory depletion, and network flooding. These vulnerabilities further increased due to the limited computational resources and security capabilities of embedded devices. This research designed, implemented, and evaluated an rule-based Intrusion Detection System (IDS) to monitor, detect, and alert against such threats in a testbed with Heart Monitor and Glucose Sensor devices.

The implementation demonstrated a lightweight yet robust monitoring system on a VirtualBox gateway. The IDS detected normal/abnormal operational changes by collecting and analyzing CPU, memory, process, and network data near-real-time, identifying deviations via rule-based thresholds. Baselines were crucial for threshold/statistical algorithms. Attack simulations—

including CPU exhaustion via stress-ng, memory leaks via Python loops, mass process injection, and network flooding with hping3—confirmed detection across all vectors. Latencies ranged 9.7–16.4 seconds (avg 14.2 s), which is acceptable for clinical timely intervention, ensuring patient care continuity during incidents.

Evaluation results validated IDS's effectiveness across multiple dimensions. Firstly, 100% detection accuracy showed discrimination between normal and malicious activities. Secondly, gateway overhead was low (less than 5% CPU under max load), making it suitable for resource-constrained environments. Thirdly, scalability held: Heart Monitor managed 400 processes/8.7M packets while Glucose Sensor managed over 1300 processes/5.4M packets with full monitoring fidelity. Finally, the Plotly dashboard provided real-time visualization for system awareness, attack correlation, and post-incident forensics.

Overall, the proposed IDS demonstrates a lightweight architecture for embedded gateways, minimal patient monitoring disruption, and high accuracy via rule-based and statistical thresholds without requiring heavy computation. This work extends Zachos et al. (2022) by adding a complete detection engine, proving the explainable, low-overhead anomaly detection that is feasible in real IoMT without ML.

Future enhancements should focus on improving system performance, flexibility, and broader utility. First, integrate adaptive statistical methods (e.g., dynamic σ adjustment) to accommodate evolving baselines, enhancing resilience to subtle/zero-day threats without ML overhead. Second, expand to larger heterogeneous networks (e.g., wearables, infusion pumps, imaging) for broader coverage in real healthcare. Third, validate under live clinical conditions with realistic traffic, loads, and interactions to refine thresholds and alert mechanisms. Fourth, include automated mitigations like traffic rate-limiting, device isolation, or resource reallocation to minimize the impact of the attack while prioritizing patient safety. Finally, incorporate secure protocols (e.g., TLS over TCP) and encryption and authentication in device-gateway channels to enhance the overall IoMT security posture.

Link for the Demonstration Video

https://youtu.be/axfhqxbHz2I

References

Deshmukh, A. and Ravulakollu, K. (2024). An Efficient CNN-Based Intrusion Detection System for IoT: Use Case Towards Cybersecurity. *Technologies*, 12(10), p.203. doi:https://doi.org/10.3390/technologies12100203.

Filippos Pelekoudas-Oikonomou, Ribeiro, J.C., Mantas, G., Sakellari, G. and Gonzalez, J. (2023). Prototyping a Hyperledger Fabric-Based Security Architecture for IoMT-Based Health Monitoring Systems. *Future internet*, 15(9), pp.308–308. doi:https://doi.org/10.3390/fi15090308.

Gyamfi, E. and Jurcut, A. (2022). Intrusion Detection in Internet of Things Systems: A Review on Design Approaches Leveraging Multi-Access Edge Computing, Machine Learning, and Datasets. *Sensors*, [online] 22(10), p.3744. doi:https://doi.org/10.3390/s22103744.

Hameed, S.S., Selamat, A., Latiff, L.A., Razak, S.A., Ondrej Krejcar, Fujita, H., Nazir, M. and Sigeru Omatu (2021). A Hybrid Lightweight System for Early Attack Detection in the IoMT Fog. *Sensors*, 21(24), pp.8289–8289. doi:https://doi.org/10.3390/s21248289.

Ji, X., Choi, H., Sokolsky, O. and Lee, I. (2023). Incremental Anomaly Detection with Guarantee in the Internet of Medical Things. doi:https://doi.org/10.1145/3576842.3582374.

Lin, Q., Liu, W., Zheng, S., Ji, J., Wong, K.-C., Li, J. and Coello, C.A.C. (2025). Federated Intrusion Detection System With Cost-Sensitive Learning for Internet of Things. *IEEE Internet of Things Journal*, [online] 12(19), pp.39677–39688. doi:https://doi.org/10.1109/jiot.2025.3586938.

Ma, Z., Chen, Z., Zheng, X., Wang, T., You, Y., Zou, S. and Wang, Y. (2023). A Biological Immunity-Based Neuro Prototype for Few-Shot Anomaly Detection with Character Embedding. *Cyborg and Bionic Systems*, 5. doi:https://doi.org/10.34133/cbsystems.0086.

Moustafa, N., Koroniotis, N., Keshk, M., Zomaya, A.Y. and Tari, Z. (2023). Explainable Intrusion Detection for Cyber Defences in the Internet of Things: Opportunities and Solutions. *IEEE Communications Surveys & Tutorials*, [online] 25(3), pp.1–1. doi:https://doi.org/10.1109/COMST.2023.3280465.

Musleh, D., Alotaibi, M., Alhaidari, F., Rahman, A. and Mohammad, R.M. (2023). Intrusion Detection System Using Feature Extraction with Machine Learning Algorithms in IoT. *Journal of Sensor and Actuator Networks*, [online] 12(2), p.29. doi:https://doi.org/10.3390/jsan12020029.

Naeem, H., Amjad Alsirhani, Alserhani, F.M., Ullah, F. and Ondrej Krejcar (2024). Augmenting Internet of Medical Things Security: Deep Ensemble Integration and Methodological Fusion. *Computer Modeling in Engineering & Sciences*, 141(3), pp.2185–2223. doi:https://doi.org/10.32604/cmes.2024.056308.

None Kamaldeep, Malik, M., Dutta, M. and Granjal, J. (2021). IoT-Sentry: A Cross-Layer-Based Intrusion Detection System in Standardized Internet of Things. *IEEE sensors journal*, 21(24), pp.28066–28076. doi:https://doi.org/10.1109/jsen.2021.3124886.

Ogab, M., Zaidi, S., Bourouis, A. and Calafate, C.T. (2025). Machine Learning-Based Intrusion Detection Systems for the Internet of Drones: A Systematic Literature Review. *IEEE Access*, 13, pp.96681–96714. doi:https://doi.org/10.1109/access.2025.3575236.

Puder, A., Rumez, M., Grimm, D. and Sax, E. (2022). Generic Patterns for Intrusion Detection

Systems in Service-Oriented Automotive and Medical Architectures. *Journal of Cybersecurity and Privacy*, 2(3), pp.731–749. doi:https://doi.org/10.3390/jcp2030037.

Rahul Vadisetty (2025). *Adaptive Machine Learning-Based Intrusion Detection Systems for IoT Era*. [online] doi:https://doi.org/10.1007/978-981-97-8457-8_17.

Ravi, V., Pham, T.D. and Alazab, M. (2023). Deep Learning-Based Network Intrusion Detection System for Internet of Medical Things. *IEEE Internet of Things Magazine*, 6(2), pp.50–54. doi:https://doi.org/10.1109/iotm.001.2300021.

Rbah, Y., Mahfoudi, M., Balboul, Y., Fattah, M., Mazer, S., Elbekkali, M. and Bernoussi, B. (2022). Machine Learning and Deep Learning Methods for Intrusion Detection Systems in IoMT: A survey. *2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*. doi:https://doi.org/10.1109/iraset52964.2022.9738218.

Schmitt, M. (2023). Securing the Digital World: Protecting smart infrastructures and digital industries with Artificial Intelligence (AI)-enabled malware and intrusion detection. *Journal of Industrial Information Integration*, 36, p.100520. doi:https://doi.org/10.1016/j.jii.2023.100520.

Seniaray, S. and Jindal, R. (2025). Enhanced anomaly-based network intrusion detection leveraging a modified picture fuzzy clustering approach. *Cluster Computing*, 28(7). doi:https://doi.org/10.1007/s10586-024-04952-z.

Sharma, R. and Sharma, N. (2024). Enhancing security of internet of medical things in fog-edge environment: a study on attack detection performance. *Sādhanā*, 49(4). doi:https://doi.org/10.1007/s12046-024-02622-9.

Thomas, T., Prakash, R. and Pal, S. (2025). Intrusion Detection in Internet of Medical Things Using Digital Twins—A Review. *Computers, Materials and Continua*, [online] 84(3), pp.4055–4104. doi:https://doi.org/10.32604/cmc.2025.064903.

Van Quan Nguyen, Viet Hung Nguyen, Tuan Hao Hoang and Shone, N. (2022). A Novel Deep Clustering Variational Auto-Encoder for Anomaly-based Network Intrusion Detection. doi:https://doi.org/10.1109/kse56063.2022.9953763.

Zachos, G., Mantas, G., Essop, I., Kyriakos Porfyrakis, Ribeiro, J.C. and Rodriguez, J. (2022). Prototyping an Anomaly-Based Intrusion Detection System for Internet of Medical Things Networks. *Greenwich Academic Literature Archive (University of Greenwich)*, pp.179–183. doi:https://doi.org/10.1109/camad55695.2022.9966912.

Zachos, G., Mantas, G., Kyriakos Porfyrakis, C.S, M. and Rodriguez, J. (2025). Anomaly-Based Intrusion Detection for IoMT Networks: Design, Implementation, Dataset Generation and ML Algorithms Evaluation. *IEEE Access*, [online] pp.1–1. doi:https://doi.org/10.1109/access.2025.3547572.