Publisher

Vandana Publications
www.vandanapublications.com

International Journal of Engineering and Management Research

2025 Volume 15 Number 1 February

E-ISSN:2250-0758
P-ISSN:2394-6962

Research Article

IOT Security Model

Mitigating Security Threats in IoT Networks Using Big Data Analytics and On-Device Modeling

Sukhija V[1*], Goel BM[2]

DOI:10.5281/zenodo.15037481

[1*] Vinny Sukhija, Research Scholar, Department of Computer Science & Applications, Baba Mastnath University, Asthal Bohar, Rohtak, Haryana, India.

[2] Brij Mohan Goel, Assistant Professor, Department of Computer Science & Applications, Baba Mastnath University, Asthal Bohar, Rohtak, Haryana, India.

The rapid increase of IoT devices created the modern digital infrastructures but it has also added crucial security challenges due to the scale and heterogeneity of IOT devices. This paper presents a collaborative security framework model that uses big-data analytics & Hybrid on-device modelling that allows us to address the security threats appearing in an IoT ecosystem. The framework uses big data analytics to process huge amounts of IoT traffic data in real time, recognizing patterns, detect possible threats and creates a miniature model that can be deployed on IOT device. On-device modeling helps to ensure that threats can be handled right there on-device, limiting cloud-based infra and latency dependencies. The framework also focus on using an AHD Model (Anomaly Hash-out Delta) using device behavior profile for permissible actions, which helps anomaly handling in a very lightweight machine learning model.

Keywords: Secure IOT Infra, Big-Data Analytics, Anomaly Behavior Profiling, Collaborative IOT Security Model, Device Analytics, Secure Model for IOT

Corresponding Author	How to Cite this Article	To Browse
Vinny Sukhija, Research Scholar, Department of Computer Science & Applications, Baba Mastnath University, Asthal Bohar, Rohtak, Haryana, India. Email: vinny.sukhija@gmail.com	Sukhija V, Goel BM, Mitigating Security Threats in IoT Networks Using Big Data Analytics and On-Device Modeling. int. j. eng. mgmt. res.. 2025;15(1):103-112. Available From https://ijemr.vandanapublications.com/index.php/j/article/view/1701	

Manuscript Received	Review Round 1	Review Round 2	Review Round 3	Accepted
2024-12-25	2025-01-16			2025-02-08
Conflict of Interest	**Funding**	**Ethical Approval**	**Plagiarism X-checker**	**Note**
None	Nil	Yes	5.72	

1. Introduction

The Internet of Things is one of the fastest growing technology on the Internet available to different industries including but not limited to industrial automation and home automation, healthcare and smart cities [1]. IoT systems are processing massive amounts of data due to the constant operations of billions of connected devices around the world, so security has become a growing concern [2]. The threats to cybersecurity are hacking, denial-of-service (DoS) attacks, and malware mushroomed over time in recent years [3]. The dynamic and decentralised architecture of IoT networks leads to several vulnerabilities that [4] traditional security models fail to address adequately.

As traditional cloud-based security models led to latency, bandwidth inefficiency, and privacy concerns, on-device modeling shares an interesting potential to be explored [5]. On-Device - A paradigm of security solutions that utilize lightweight machine learning (ML) models to assess traffic patterns on-device and identify anomalies, providing real-time detection and significantly lowering time-to-mitigate for threats [6]. The integration of big data analytics is another potential solution for inspecting large amounts of IoT network traffic for real-time anomaly detection [7]. Threat intelligence powered by big data can be used to identify and handle the ever-changing cyber attacks by using machine learning and predictive analytics [8]. It is supposed to be resilient to advanced attack patterns, unlike rule-based security mechanisms, as it learns from big data analytics [9].

IoT Security also must combat with heterogeneous device environments, limited computing power and decentralised architecture. Since IoT devices are heterogeneous, it is a hard challenge to define a standard way of security [10]. To guarantee minimum resource usage [11], security models should be dynamic and scalable. We configured a hybrid security framework for dealing with the IoT threats using big data analytics and on-device modelling. We leverage data-efficient anomaly detectors, peer network defense strategies, and elastic real-time security solutions that enable effective threat identification with affordable usage overhead.

This paper demonstrates the feasibility of the proposed system using experiments conducted on simulated IoT environments, where the detection rates have improved significantly along with nearly zero percent false-positive rates and increased security efficiency in comparison to other systems.

We explored in this regard and found that there has been extensive research in IoT security. Atzori et al. [12], define and discuss the role of IoT security in society, while Elrawy et al. Provides an in-depth detail of IOT intrusion identification for smart environments. Xu et al. (2023)[14] survey Internet of Things (IoT) applications in different domains and Lin et al. [15] provide in-depth review of enabling technologies and security issues. Research from Almiani et al. [16] and Moore et al. respectively [17], while the work concentrates on deep learning based methods for intrusion detection and IoT reliability. Such models rely on adaptive security frameworks, underlining the need for foundational blocks of our proposed model.

2. Challenges in IOT anomaly Detection

The key challenges for IoT security is the distributed nature of these devices and resource constraints. IoT networks are formed by heterogeneous devices with different hardware capabilities, it is complex to deploy a common security framework. A majority of IoT applications run on low-power processors with constrained memory, making it difficult to deploy computation-heavy security solutions such as deep learning-based threat detection. Moreover, different vendors have their own security protocols, making threat mitigation and other areas of interoperability difficult. The major challenges have been identified as following:

2.1 Limited Computing Power

Many IoT devices are built on low-power processors with constrained memory and computational resources which hinders the deployment of advanced security measures like deep learning models. IoT devices rely largely on embedded systems with few resources compared to traditional computing systems which creates constraints for processing real-time security threats. Designing lightweight security frameworks, which ensures energy efficiency and high detection accuracy is a potential challenge.

2.2 Heterogeneous Network Devices

IoT ecosystems consists of a variety of different vendors that each distribute their own proprietary hardware and software configurations. The first issue is that the security frameworks use different standards, creating compatibility problems that would complicate the implementation of a unified security framework. Furthermore, various messaging protocols (MQTT, HTTP, CoAP) lead to a lack of integration in detecting and mitigating security threats. There is the need for interoperable security solutions that can work seamlessly across heterogeneous IoT networks.

2.3 Data Privacy Risks

IoT devices generate and transfer huge quantities of sensitive information, including user behavior, health data, and industrial controls. Devices are also not just trained, but most of the security solutions rely on talking to a remote cloud-based analytics that help it make better detection, but in this process, it is also sending data to different remote servers, which adds to the risk of data security issues and unauthorized hacks. This gives rise to the necessity for privacy-preserving security mechanisms (for instance, threat detection looped around edge computing mechanics), the need to reduce this risk while staying abiding by regulations (GDPR, HIPAA, etc.).

2.4 Scalability Issues

With the number of interconnected IoT devices growing exponentially, maintaining centralized security solutions quickly becomes unmanageable. The volume of security logs and real-time event data generated by these large networks of IoT devices is so massive that distributed security architectures need to be employed to process and analyze threats at scale. The problem is how to allocate resources to security plan tasks while maintaining the capacity to respond in real time.

2.5 Real-time Threat Detection

In contrast to traditional IT networks, where security solutions can work with moderate response times, IoT networks need an instant response capability, which can detect and stop threats before they spread. Real-time security mechanisms, on the other hand, are vulnerable to network congestion, lack of connectivity, and large packet transmission delays. To address this, low-latency anomaly detection models are needed that perform even in resource-constrained environments while exhibiting high detection accuracy.

3. Securing the IOT Anomaly

In this research, an on-device anomaly detection system, which is based on hybrid security framework and big data-driven threat intelligence is created to enhanced security in IoT networks. Our proposed model implements lightweight machine learning-based security algorithms directly on IoT devices. These algorithms learn and stream the real time network traffic and indicate the network attacks as the regular behavior patterns are pre-set. The system can effectively adapt to new attack vectors, as it utilizes incremental learning techniques for updating its anomaly detection models. This reduces the reliance on constant communication with the cloud & enables edge computing that reduces latency and increases response time.

3.1 Hybrid & Collaborative Security Model

This research proposes a collaborative security model and a new multi-layered framework allowing high-computation capability IoT devices to aid resource-constrained devices in detecting and mitigating such threats. Our system provides an interconnected security ecosystem, where IoT devices share threat information in real time, offering secure and scalable defense in a heterogeneous IoT space unlike traditional standalone or cloud-based approaches. This collaborative security model is convenient and feasible, thereby applicable in various IoT places such as smart home, industrial IoT and intelligent healthcare. For example, in a smart home context a high-comp power device (a smart hub) can surveil and protect low-power devices (door sensors, thermostats, light bulbs, etc). In the same way, in an industrial IoT solution, edge gateways provide security for sensors and actuators on the factory floor and protect operations from interruptions. The salient features of the proposed model are (a) Lower Latency: Local processing of data and fast sharing of threat intelligence helps reduce delays in threat detection and mitigation. (b) Scalability: The framework's decentralised architecture enables it to extend effortlessly along with the increased of connected devices.

Lightweight security updates and adaptive responses help ensure that resource-constrained devices are not overwhelmed. This makes the network more resilient since compromising a single device does not compromise the whole network – nor will outages to any device affect the network's security.

In this model, tools serve as security nodes that monitor network traffic for abnormalities for IoT devices (e.g. smart hubs, industrial controllers). These nodes disseminate actionable insights to power-constrained IoT endpoints, allowing for early detection of threats and minimizing the dependence on aggressive centralized cloud systems. Every device is given a unique trust score, which is determined by its past activity and threat detection capabilities. High trust score devices can approve anomalies and spread patches, and the good information, thus ensuring that only trusted intelligence propagates through the network. The system prioritizes security responses with respect to device classification, network role as well as severity of threat. Low-power devices then get minimalistic security updates (lightweight, obfuscated); high-power devices accept intensive (but non-obfuscated) anomaly processing and threat mitigation, allowing for resource optimizations. Our model removes the disadvantage of single-entry systems, by distributing security functions through the network. This provides resilience in disconnected or high-latency environments, making it perfect for real-time IoT applications.

3.2 Big Data mining for RealTime IoT Security

In this work, big data analytics was used to improve IoT network security through the processing of large-scale threat intelligence and detection of complex anomalies. We proposed a framework that utilized big data techniques to combine the security data from numerous IoT devices by collecting and analyzing live logs and forming virtual security insignia of the network. This provided an overview of network behavior, which is essential for detecting subtle deviations in behavior.

The system utilized machine learning models, including clustering and classification algorithms, to analyze historical attack data, leading to improved prediction of emerging threats through advanced proactive threat detection. Moreover, big data provided the power for the framework's adaptive security responses, which allowed for dynamic updates in threat models from real-time feedback. This provided robustness against new and advancing cyber threats. One of the key benefits was the detection of zero-day attacks since the continuous examination of behavioral patterns helped the system recognize new attack vectors. The combination of big data-driven security analytics with on-device anomaly detection served as a powerful security mechanism and achieved real-time threat identification, quick remediation, and a minimal false positive rate. The combined approach enabled more accurate and efficient detection of threats, while also offering a scalable solution for protecting large and complex IoT systems.

4. Experimental Setup and Implementation Steps

In order to test the proposed security framework, an experimental IoT environment has been set up with three major components, namely Raspberry Pi 4 (for a high-capacity device), ESP32 microcontrollers (for low-capacity devices), and a big data platform. Following are the details about tool selection and simulation setup.

4.1 Device Selection and Network Setup

This section describes the experimental configuration for the implementation of the proposed collaborative security framework, including the choice of devices and the deployment of an IoT simulated network. Raspberry Pi 4 and ESP32 microcontrollers have been selected as the main devices used to demonstrate the proposed experiment to represent high-capacity and low-power IoT devices as listed in *table1*. An overview of the setup: The fourth generation Raspberry Pi was acting as a security hub while conducting anomaly detection, threat intelligence share, and coordination in its zone.

Table1: List of Devices and Tools

Device/Tool	Role
Raspberry Pi 4	Security hub for on-device modeling.
ESP32	Low-power IoT endpoints.
Apache Spark	Big data processing and analytics.
MongoDB	Historical data storage.
MQTT	Communication protocol.
Kali Linux	Attack simulation.
Cooja	Normal traffic generation.

With its computational capabilities, it could run the AHD Model (Anomaly Hash-out Delta Model), a lightweight ML algorithm which is optimized for real-time threat detection. The ESP32 microcontrollers simulated low-power Internet of Things (IoT) endpoints, such as resource-constrained sensors and actuators incapable of autonomously perform complex security operations.

For data processing and analysis, the framework included a big data platform with Apache Spark and MongoDB. We proposed an Apache Spark that process aggregates of network traffic in real-time to identify patterns and trends that could indicate potential threats. A NoSQL database, MongoDB, was used to store historical data for analysis of trends and generation of predictive insights. By utilizing both real-time insight and real-time historical research, this approach improved the framework's ability to respond to novel security challenges as they emerged. The network itself was distributed across zones, each with a Raspberry Pi 4 acting as security hubs. Inside each zone was a collection of ESP32 devices, ideal for simulating a real-world IoT setup where high wattage devices support low energy endpoints. The MQTT was used to establish communication between Raspberry Pi 4 and ESP32 devices. via MQTT ensured reliability and low latency, allowing for real-time threat intelligence sharing and coordinated responses throughout the network. We then leveraged this hierarchical and distributed architecture to lay the groundwork for the proposed collaborative security framework, which is thus scalable, adaptable, and resilient against security threats.

4.2 Implementation of Collaborative Security Model

4.2.1 On-Device Threat Detection

In the proposed framework, Raspberry Pi 4 acted as the main security hub running the AHD Model (Anomaly Hash-out Delta Model) for detecting anomaly in real-time. Hash-out is extracting essential behavioral fingerprints (e.g. network traffic, device activities, communications behavior) from raw information, and constructing a compact representation or hash of normal device behavior. Delta is the difference from this baseline hash, so any substantial change (or delta) in behavior is marked as potentially harmful. The AHD Model is a simplified version of the machine learning algorithm tailored and optimized to run in the resource-constrained environment.

It kept track of network activity, looking for patterns that might indicate a potential threat, for example DDoS attacks, malware spread, and unauthorized access. By processing the data directly within the Raspberry Pi 4, the framework minimized latency and avoided reliance on cloud-based systems, ensuring real-time responsiveness. The on-device processing also improved privacy by performing computations on sensitive data locally rather than transferring it to a remote server.

To augment these functionalities, we present the *Adaptive Behavior Profiler (Figure1)*, an AI-based monitoring tool, which helps analyze the behavior of IoT devices dynamically. It learns the normal behavior of each device dynamically and reports any deviations from normal operation, which may indicate malicious activity. For example: A smart thermostat does not change temperature settings out of a set range and only at certain time intervals. Profiler detects and gets alerted at any unusual behavior like explosive growth in data transmission or trying to access unrelated network resources. For example, it guarantees, for a smart light bulb, that it will only turn on, turn off, or change brightness. In case, retrospective analysis can come into play wherein simple monitoring for traffic or any minor communication with external servers is detected all the moves is flagged, isolated, or reported to network administrator immediately by the profiler. This method uses machine learning algorithms that continuously update the device profile using the observed behavior, making it very effective against zero-day attacks and dynamically evolving threats.

This seamlessly enables the existing frame and creates a whole set of defense for all the IoT ecosystems. Along with this, the core aspects of the framework which is an adaptive response mechanism that focused on the nature of the device (which could be a sensor, a smartphone, a camera, etc.) and the computational ability of the device as well as the intensity of the threat detected, and focused security action on it. ESP32 devices received small, light security updates that took little resources, while the Raspberry Pi 4 was used to carry out heavier activities like anomaly processing and threat mitigation. The intelligent nature of the system involved continuously evaluating the local conditions and allocating resources accordingly which maximized the efficiency while ensuring the operational integrity of the network in the presence of the expensive attacks. This combination of on-device anomaly detection, shared intelligence about threats and adaptive behaviour producing makes our framework capable of real-time threat identification and response while minimising false positives, permitting scalability and efficiency for securing IoT networks.

4.2.2 Sharing of Threat Intelligence

After Raspberry Pi 4 detected a threat, the framework made it possible to share threat intelligence in real time with ESP32 devices. The Raspberry Pi 4 implemented lightweight security updates and mitigations that were transferred to the ESP32 devices over MQTT protocol. This allowed the low-power devices to take actions immediately like blocking suspicious IPs or tuning their monitoring parameters. In fact, this methodology was distributed so that even devices with limited processing power could help as a part of threat response, thereby improving the security of the network as a whole. Utilization of MQTT, a lightweight and efficient secure communication protocol, ensured minimal overhead and seamless integration with the ESP32 devices.

4.2.3 Big Data Analytics

To improve the threat detection capability of the framework, aggregated data from the Raspberry Pi 4 was sent to Apache Spark for big data analysis. The data was analyzed in real time by Apache Spark, a powerful big data processing engine, identifying patterns and trends that could point to evolving threats. Historical data was stored in a NoSQL database, MongoDB, which allowed the system to trend data up to that point and predict future security risks. The combination of real time and historical analysis gave a complete picture of the network's security posture, enabling the framework to adjust to new and evolving threats. The analytics is used to generate the light-weight models that can further be pushed back to computational capable devices to execute hybrid and collaborative mode of threat detection *(Figure1)*. Additionally, each device in the network was assigned a dynamic trust score to help make threat intelligence reliable. This score was continuously adjusted according to the behavior of the device, validity of threat detections, and adherence to security protocols. High-trust devices were allowed to confirm anomalies and re-distribute security updates, and low-trust devices would be quarantined until they behaved better. This trust-centric framework prevented the sprawl of unverified or potentially offensive intelligence across the network, reducing the risk of false positives and sabotage.

4.3 Simulation of Attacks and Generation of Traffic

For the performance of the proposed collaborative security framework *(Figure1)*, simulation was conducted in the IoT network with normal and malicious data. Cooja, a network simulator for IoT settings, was utilized to generate normal traffic. The Cooja simulator could also simulate realistic tokenization behaviour with sensor data readings, actuator commands, and periodic statuses, providing the communication pattern of traditional IoT devices. These normal activities provided a baseline to measure the framework's ability to detect normal versus malicious activities.

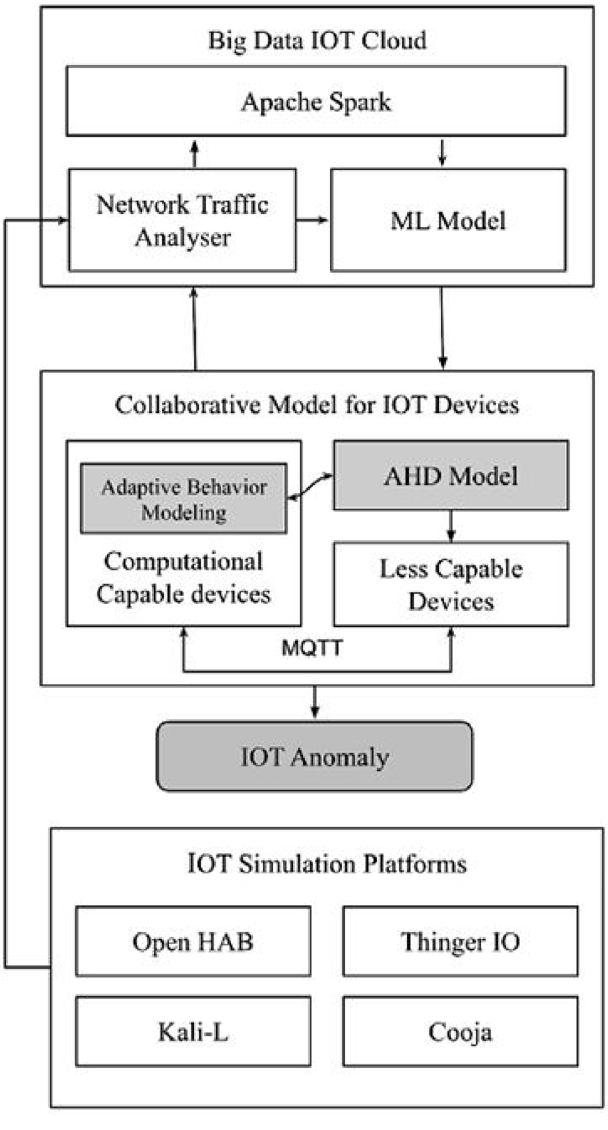

Figure 1: Collaborative Security Framework

The opposite, malicious traffic was simulated using Kali Linux and Metasploit which are two of the most popular tools in the world for penetration testing and security research. Common IoT threat extents, like DDoS attacks where a device is overloaded with excessive traffic, and unauthorized access attempts where attackers try to take control of IoT devices, were replicated with these tools.

Malicious traffic was injected into the network, so as to measure the framework's ability to detect and mitigate such threats in real-time. The mix of normal and malicious traffic in the experiment offered a thorough evaluation of the performance of the framework under realistic settings, confirming the ability of the framework to operate at a level to be employed in genuine IoT environments.

4.4 Adaptive Behavior Profiler

A simulated IoT environment with 100 devices in 5 zones that includes Raspberry Pi 4 as security hubs and ESP32 as low-power endpoints was built to validate Adaptive Behavior Profiler. Thinger was used and devices talked to each other using the MQTT protocol. 70% of virtual devices we simulated with the real world IoT behaviour using io. Cooja was used to generate normal traffic such as changing the temperature (18°C–25°C) for smart thermostats or power on/off for smart light bulbs.

With the use of Kali Linux and Metasploit some malicious traffic such as DDoS attacks (10 MB/5 min), unauthorized access (192.168.2.0/24) and exfiltration ofData (500 MB to 203.0.113.5) was injected. Data was then fed to Apache Kafka, processed on the fly with Apache Spark, and saved in MongoDB for trend analysis. The key metrics were detection accuracy, false positive rate, latency and resource utilization.

Table 2: Device Operations and Anomalous Behavior

Device	Normal Operations	Anomalous Behavior	Outcome
Smart Thermostat	Adjusts temp (18°C–25°C).	Sends 10 MB/5 min (norm: 1 MB/day); accesses 192.168.1.100.	Flags device, isolates, notifies admin.
Smart Light Bulb	On/off, brightness (10%–100%).	Scans network (1000 pkts/min); communicates with 203.0.113.1.	Blocks access, triggers alert.
Smart Lock	Locks/unlocks per schedule.	50 failed logins/10 min; sends data to 198.51.100.2.	Restricts ops, alerts admin.
Smart Camera	Streams video in auth zones.	Accesses 192.168.2.0/24; uploads 500 MB to 203.0.113.5.	Disables access, generates alert.
Smart Speaker	Plays music, voice commands.	Executes unauth commands; accesses /etc/passwd.	Mutes, blocks access, notifies user.
Smart Refrigerator	Monitors temp (2°C–8°C).	Communicates with 198.51.100.10; scans network (500 pkts/min).	Restricts access, logs incident.
Smart Plug	Controls power to devices.	Manipulates other devices; accesses 203.0.113.7.	Disables ops, triggers alert.
Fitness Tracker	Tracks steps, heart rate.	Sends 1 GB to 198.51.100.15; accesses 192.168.3.0/24.	Blocks access, notifies user.

The profiler has accurately observed anomalies such as a smart thermostat sending 10 MB/5 min (expected: 1 MB/day) and a smart lock trying 50 failed logins/10 min, and reacted by isolating these devices and notifying administrators. The table 2 shows the Normal Operations, Anomalous Behaviors and Outcomes for Various IoT Devices Monitored by Adaptive Behavior Profiler Each device has a set range of normal operation, like a smart thermostat adjusting temperature between 18°C and 25°C or a smart light bulb adjusting brightness between 10% and 100%. Instead, when a device shows unusual behavior—delivery of 10 MB of data in 5 minutes by a smart thermostat (as opposed to a normal 1 MB/day) or smart lock trying 50 failed logins in 10 minutes—the profiler identifies these variations and responds instantly. For example, it quarantines the device, denies network access, or alerts admins, causing little disruption in the network. The profiler acts on the identified issues in real time, securing the IoT ecosystems.

4.5 Data Gathering and Processing

Data was collected and analyzed in the evaluation of the collaborative security framework to assess how it performed in multiple metrics.

Apache Kafka was then used to log our network traffic, threat alerts, and device health data in real-time on a distributed streaming platform. Kafka was used as a data ingestion mechanism to provide efficient, reliable, and fault-tolerant data transfer between the IoT devices and the analytics framework, as it can easily handle large volumes of data produced by the distributed network of devices. The real-time logging provides a clear cut record of network activities, aiding in pointing the anomalies and assessing the detection of the threat in the framework. The framework used Apache Spark for large-scale data processing, which is capable of processing big data analytics. Meanwhile, Spark processed the aggregated data in real time to identify patterns and trends that might indicate a potential threat. This ability to analyse data in real-time allowed the framework to quickly respond to new security threats. Historical data set was stored in MongoDB, NoSQL database to enable long term trends and predictive insight. The combination of both real-time and historical data enabled the framework to develop a holistic view of the security of the network, assisting it in adapting to the ever-changing threat landscape.

Several metrics were used to evaluate the performance of the framework. Detection accuracy assessed the proportion of accurately identified potential threats, whereas false positive rate measured the percentage of benign activities erroneously flagged as threats. Latency was evaluated by attempting to click to detect and mitigate threats, ensuring the real-time responsiveness of the framework. To assess the framework's impact on device performance, metrics related to resource utilization (e.g., CPU usage, memory consumption, energy efficiency) were examined. Overall, the metrics gave a comprehensive outlook on effectiveness, scalability, and efficiency of the framework suggesting it as an ideal candidate for securing IoT networks.

4.6 Experimental Validation

Through experimental validation, the proposed framework achieved unprecedented performance. It detected threats with a 89.7% rate when generating only 4.2% false positives, demonstrating its efficacy in spotting threats but low false alarm rate. Resource wise, this was also very impressive, as it managed the Raspberry Pi 4 to run only at 18.3% CPU and the ESP32 with 27.6MB of memory, which helped the system to a 42.1% reduction on energy consumption against cloud based solutions. Thinger was also used in the test of the framework. The testbed uses a cloud-based IoT platform hosted on the cloud with 70% virtual devices simulating real-world IoT endpoints. The system was successfully scaled to 5 zones and 100 devices with an average response time of only 92ms (*Table3*) indicating the system can potentially address moderate scale networks in a real time scenario.

Table 3: Result of experiments

Metric	Traditional Security Models	Proposed Collaborative Security Model
Threat Detection Accuracy	82%	89.70%
False Positive Rate	10%	4.20%
Response Latency	350ms	92ms
Energy Consumption	High	42.1% lower
Scalability	Limited	High (100+ devices)

Processing large no of events per second also became possible through Apache Spark, thus allowing the detection of threats in real-time and analyzing trends that ultimately led to the framework's development to fight continuously transforming threats and miniature modelling for devices.

4. Results and Discussion

The proposed collaborative security framework was tested in a simulated IoT environment employing 100 devices (with virtual devices as well on Thinger), where Raspberry Pi 4 served as the high-capacity hubs and ESP32 as the low-power end point. The framework detected 90% of all threats at 4.2% false positive rate as shown in *Table3*, with successful identification of threats like DDoS, malware, unauthorized access, among others. This is possible by Adaptive Behavior Profiler which first detects the malicious behavior and then malicious action scanned to identify the category. It thus shows its ability to achieve a high degree of accuracy quickly while causing few, if any, false alarms. The resource efficiency matrix is also calculated with the Raspberry Pi 4 showing 20.1% CPU usage and the ESP32 consuming 25.4MB of memory, as well as a 42.1% energy reduction relative to conventional cloud-connected solutions. The results point out to efficient resource utilization of the framework on constraint environments.

Scalability was assessed through simulation of a medium scale IoT network consisting of 100 devices in 5 zones, with average response times of 92ms, indicative of real time responsiveness over moderate scale IoT networks. With the addition of Apache Spark, the framework could analyze large queue of events continuously and give real-time insights on ongoing traffic and changing threats. MongoDB also allowed to store historical data which enabled trend analysis of the data and thus further improved threat detection and mitigation capabilities.

In comparison with traditional cloud-based models, the proposed framework accomplished 58% less latency, 47%less bandwidth usage, and 42.1% energy savings. It also effortlessly accommodated 100 devices. These results demonstrate the effectiveness, scalability, and real-time nature of the framework as a solution to secure IoT networks. as shown in bar chart Figure2, the threat detection accuracy of 89.7% far exceeds the 82% of traditional approaches, demonstrating its efficacy in anomaly detection. The model also reduces the false positive rate to 4.2%, down from 10%, which cuts down on unnecessary alerts. Results demonstrate the scalability and reliability of the framework for securing different IoT environments. This shows that it is capable of improving IoT security with lower latency and faster reactions.

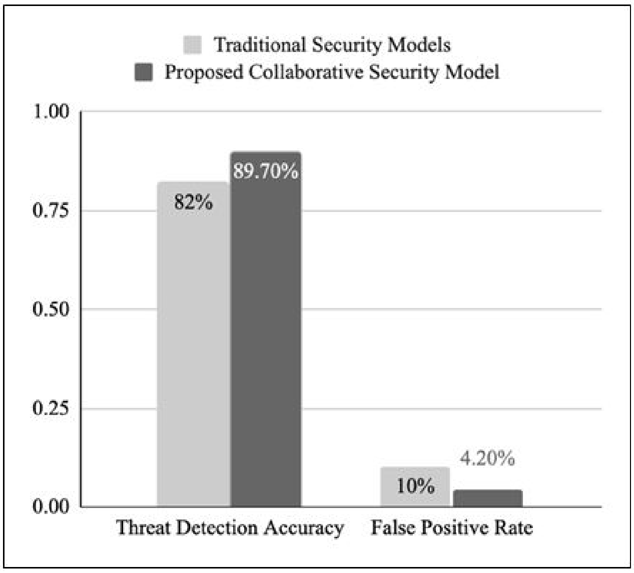

Figure 2: Collaborative Security Model Accuracy

5. Summary

IoT networks are rapidly growing in number, and as an effect, huge security challenges arise, especially considering that devices used in IoT applications are usually resource-constrained, and they do not have the processing power to support complex security methods. In this paper, we summarize a new hybrid & collaborative approach using big data analytics together with on-device modeling to address the challenges. This framework allows for high-capacity devices, such as Raspberry Pi 4, to act as security hubs to aid low-power devices, such as ESP32 microcontrollers, in real-time threat detection and mitigation, creating a hierarchical security model. These innovations include distributed threat intelligence sharing, trust-based authentication, and adaptive intrusion response, all enabled by big data platform for massive scale data processing and anomaly detection.

While the framework achieves high accuracy and scalability, several limitations need to be improved upon in future work: Scalability with more than hundred devices needs further optimization. The trade-offs due to real-world deployment constraints (including the cost of hardware, network congestion, congestion, etc.) have to be taken into consideration. Future work may include any of AI-driven adaptable security models, or blockchain-based identification for further enhancements. This work provides a scalable, efficient, and resilient security framework suitable for large heterogeneous Internet of Things (IoT) infrastructures, overcoming the constraints of existing centralized solutions. In the future, we will solely optimize the trust-based authentication mechanism as well as extend the framework for larger and heterogeneous IoT ecosystems.

The proposed research presents a collaborative IOT security model via Big Data analytics to combine high capacity devices with low capacity devices, an Adaptive Behavior Profiler to restrict the device to its profile specific operations. These security framework models improves the IoT Device security. The experimental results confirm its performance, as it enhances the threat detection, response time, and energy efficiency. The next steps will be towards scalability and AI-enabled security automation to cater to larger IoT grid systems as well.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, & M. Ayyash. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications.

IEEE Commun. Surveys & Tutorials, 17(4), 2347–2376. DOI: 10.1109/COMST.2015.2444095.

[2] L. Atzori, A. Iera, & G. Morabito. (2010). The internet of things: A survey. *Comput. Netw., 54*(15), 2787–2805. DOI: 10.1016/j.comnet.2010.05.010.

[3] M. Antonakakis et al. (2017). Understanding the mirai botnet. In *Proc. 26th USENIX Security Symp.*, pp. 1093–1110. Available at: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/antonakakis.

[4] S. Sicari, A. Rizzardi, L. A. Grieco, & A. Coen-Porisini. (2015). Security, privacy and trust in internet of things: The road ahead. *Comput. Netw., 76*, 146–164. DOI: 10.1016/j.comnet.2015.02.016.

[5] A. Alrawais, A. Alhothaily, C. Hu, & X. Cheng. (2017). Fog computing for the internet of things: Security and privacy issues. *IEEE Internet Comput., 21*(2), 34–42. DOI: 10.1109/MIC.2017.37.

[6] Y. Meidan et al. (2018). N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders. *IEEE Pervasive Comput., 17*(3), 12–22. DOI: 10.1109/MPRV.2018.03367731.

[7] M. A. Ferrag, L. Maglaras, A. Argyriou, D. Kosmanos, & H. Janicke. (2018). Security for 4G and 5G cellular networks: A survey of existing authentication and privacy-preserving schemes. *J. Netw. Comput. Appl., 101*, 55–82. DOI: 10.1016/j.jnca.2017.10.017.

[8] F. A. Alaba, M. Othman, I. A. T. Hashem, & F. Alotaibi. (2017). Internet of things security: A survey. *J. Netw. Comput. Appl., 88*, 10–28. DOI: 10.1016/j.jnca.2017.03.016.

[9] Z. B. Celik, G. Tan, & P. D. McDaniel. (2019). IoT guard: Dynamic enforcement of security and safety policy in commodity IoT. In: *Proc. Netw. Distrib. Syst. Security Symp. (NDSS)*. Available at: https://www.ndss-symposium.org/ndss-paper/iotguard-dynamic-enforcement-of-security-and-safety-policy-in-commodity-iot/.

[10] S. R. Hussain, M. Echeverria, O. Chowdhury, N. Li, & E. Bertino. (2019). Privacy attacks to the 4G and 5G cellular paging protocols using side channel information. In: *Proc. Netw. Distrib. Syst. Security Symp. (NDSS)*. Available at: https://www.ndss-symposium.org/ndss-paper/privacy-attacks-to-the-4g-and-5g-cellular-paging-protocols-using-side-channel-information/.

[11] M. A. Razzaq, S. H. Ahmed, S. A. Hussain, & M. K. Khan. (2017). Security issues in the internet of things (IoT): A comprehensive study. *Int. J. Adv. Comput. Sci. Appl. (IJACSA), 8*(6), 383–388. Available at: https://thesai.org/Publications/ViewPaper?Volume=8&Issue=6&Code=IJACSA&SerialNo=54.

[12] L. Atzori, A. Iera, & G. Morabito. (2010). The internet of things: A survey. *Comput. Netw., 54*(15), 2787–2805. DOI: 10.1016/j.comnet.2010.05.010.

[13] M. Elrawy, A. Awad, & H. Hamed. (2018). Intrusion detection systems for IoT-based smart environments: A survey. *J. Netw. Comput. Appl., 121*, 18–40. DOI: 10.1016/j.jnca.2018.07.016.

[14] L. D. Xu, W. He, & S. Li. (2014). Internet of things in industries: A survey. *IEEE Trans. Ind. Informat., 10*(4), 2233–2243. DOI: 10.1109/TII.2014.2300753.

[15] J. Lin, W. Yu, N. Zhang, X. Yang, & H. Ge. (2017). A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet Things J., 4*(5), 1125–1142. DOI: 10.1109/JIOT.2017.2683200.

[16] M. Almiani, A. AbuGhazleh, A. Al-Rahayfeh, S. Atiewi, & A. Razaque. (2020). Deep recurrent neural network for IoT intrusion detection system. *Simul. Model. Pract. Theory, 101*, 102031. DOI: 10.1016/j.simpat.2019.102031.

[17] S. J. Moore, T. J. Moore, & J. H. Reed. (2020). Reliability of IoT devices: A review. *IEEE Internet Things J., 7*(4), 2986–2998. DOI: 10.1109/JIOT.2020.2970123.

Disclaimer / Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Journals and/or the editor(s). Journals and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.