ILKOM Jurnal Ilmiah Vol. 18, No. 1, April 2026, pp. 43-57
Accredited 2nd by KEMDIKTISAINTEK No. 10/C/C3/DT.05.00/2025; E-ISSN 2548-7779 | P-ISSN 2087-1716

Isolation Forest-Based Anomaly Detection in IoT Smart Home Network Traffic

Ahmad Luthfi [a,1,*]; Emigawaty [a,2]
[a] *Department of Informatics, Faculty of Industrial Technology, Universitas Islam Indonesia, Yogyakarta, Indonesia*
[n] *Department of Informatics, Faculty of Computer Science, Universitas Amikom Yogyakarta, Yogyakarta, Indonesia*
[1] *ahmad.luthfi@uii.ac.id;* [2] *emigawaty@amikom.ac.id*
* *Corresponding author*

Article history: Received October 18, 2025; Revised October 25, 2025; Accepted February 24, 2026; Available online April 20, 2026

Abstract

The convergence of the Internet of Things (IoT) and Society 5.0 has successfully led to a human-centered and data-driven life ecosystem. IoT has become the backbone for infrastructure implemented in various domains, ranging from smart homes and smart farming to smart industrial environments. Nevertheless, as IoT devices become more connected and integrated into the ecosystem, the attack surface expands and network security becomes more challenging. The massive convergence and connectivity of IoT devices have a high potential for attacks on network infrastructure, such as Denial of Service (DoS), port scanning, exfiltration, brute force, and man-in-the-middle attacks. This study aims to detect anomalies in IoT network traffic by applying the Isolation Forest (IF) algorithm. The dataset was obtained from an IoT gateway connected to smart home devices and includes features such as data packet size, connection duration, source and destination capacity, attack protocols used, and the connection status of each device. The experimental results of this study indicate that the IF method can identify smart home device attacks with a competitive level of accuracy. The results of the anomaly analysis are then presented through a confusion matrix, classification report, and analytical visualizations such as 2D PCA, t-SNE, heatmap, and temporal distribution of anomalies. This study declares that the IF method contributes effectively to the analysis of Intrusion Detection Systems (IDS) in IoT environments such as smart homes that are heterogeneous and dynamic.

Keywords: Isolation Forest; Anomaly; Internet of Things; Intrusion Detection System; Smart Home.

Introduction

The rapid development of the Internet of Things (IoT) over the past decade has expanded its use domains beyond industrial environments to include transportation, healthcare, agriculture, urban planning, logistics, and household applications such as smart homes [1]. IoT devices are characterized by low power consumption, always-connectedness, sensor support, real-time operation, and intelligent automated decision-making [2]. Therefore, these situasions provide efficiency and impove the quality of services in various sectors [3]. However, the massive development and complexity of devices, the variety of protocols used, and the disparate vendor standards simultaneously increase the potential security risks of IoT networks.

IoT has proven to be of enormous benefit to stakeholders in simplifying and supporting increasingly complex work [1], [2]. In the smart city domain, for example, IoT can improve energy efficiency through smart streetlights, automated traffic management, urban disaster mitigation, and public safety monitoring [2]. The concept of smart homes, for instance, provides convenience for residents by automating lighting, air conditioning, and home security [3], [4]. Other popular device that can record and track the wearer's physical activity and healthy lifestyle, such as smartwatch, is an example of IoT implementations today [4].

However, despite its heterogeneous nature in social and individual benefits, IoT is vulnerable to threats, particularly network-based ecosystem attacks [1], [5]. First, due to its low computing power and limited electrical resources, IoT is difficult to implement complex security mechanisms, such as network-level encryption or sophisticated firewall settings [4], [5]. Second, the risk of infiltration by data packet anomalies such as DoS, port scanning, exfiltration, and brute force attacks is quite high because IoT devices are always integrated with the network [5], [6]. Third, Empirical studies consistently show that persistent connectivity and the diversity of communication protocols, such as Zigbee, CoAP, and MQTT, open up a wide attack surface and significant security gaps [6], [7].

http://dx.doi.org/10.33096/ilkom.v18i1.3156.43-57

Furthermore, in the IoT ecosystem, there are several potential types of attacks that can disrupt data traffic or even completely paralyze the network system [5], [8]. The first attack is known as Denial of Services (DoS/DDoS), a technique for attacking IoT services by flooding data traffic, as carried out by the Mirai IoT botnet [8], [9]. This Mirai botnet infects, takes over, and controls IoT device resources through DDoS attacks [10]. The second type of attack is port scanning, which exploits open ports on the IoT network and then carries out further exploitation via TCP/UDP connections on port 80 and SSH port 22 [8]. Third, an equally popular type of attack is Man-in-the-Middle (MITM), which infiltrates the communication path between sensor devices and IoT gateways, then eavesdrops, modifies, and manipulates transmitted data without the knowledge of the sender or recipient [10]. The final type of attack that has been identified is an injection attack, which inserts malicious commands, such as SQL injection on the backend server or command injection in the firmware of IoT hardware [10], [11].

Predicting the time and which device types will impact interoperability disruptions in IoT environments is problematic due to the variety of attacks targeting network traffic [12]. IoT service disruptions, such as the shutdown of CCTV or other surveillance cameras during an attack, can result in the perpetrator's digital footprint being unknown or even completely unidentified [12], [13]. In the medical field, attacks on healthcare IoT through medical sensors manipulated or even disabled by the perpetrator can endanger patients and lead to fatal diagnostic errors for medical personnel [14]. Personal data leaks, for example, are highly likely due to MITM or SQL injection attacks that exploit sensitive IoT user data, such as habits and preferences, sensitive disease types, and current drug therapies. Thus, the success of one type of attack on a hacked IoT device can lead to a wider attack radius [15].

At the same time, signature-based IDS methods require regular signature database updates to detect various types of attacks on IoT networks, including zero-day attacks [14], [15]. Rule-based IDS approaches are ineffective due to the highly variable and dynamic load and traffic of IoT networks [8]. In addition, IDSs are unable to detect new anomalies due to dynamic changes in IoT service traffic, coupled with the complex scalability of hundreds or even thousands of connected devices [16]. Therefore, an unsupervised adaptive anomaly-IDS approach is needed to detect abnormal attack patterns even without complete labels [17]. Although previous studies have focused on machine learning-based IDS analysis, unfortunately there is still a wide gap in this field especially in unsupervised approaches for adaptive anomaly identification.

This study aims to detect intrusion system anomalies in IoT network traffic by applying the Isolation Forest (IF) algorithm. The dataset is obtained from an IoT gateway connected to a smart home device and includes features such as data packet size, connection duration, source and destination capacity (in bytes), the attack protocol used, and the connection status to the device [18]. The IF algorithm was chosen because it is theoretically effective for high-dimensional, unlabeled data obtained from the IoT gateway server [19], [20]. In addition, IF is also able to isolate outliers or attack anomalies and is suitable for cases with a lower proportion and frequency of attacks compared to normal attack traffic [21], [22]. Whereas the supervised technique necessitates extensive labeling, this study focuses on the IF algorithm's capability to discover and recognize dynamic anomalous patterns. This study also presents visualization analyses such as PCA, t-SNE, heatmaps, and distributions of data traffic scores to make the IDS anomaly results easier to understand for researchers and practitioners.

Related Work

Signature-based rules are the traditional approach that IDSs use to detect attacks on IoT networks [23], [24]. IDSs detect these attacks by matching data traffic patterns on IoT networks using a previously stored knowledge base. For attacks with pre-existing knowledge tables, this technique is quite effective and time-saving. However, there are gaps, particularly in zero-day exploits with newer attack variants such as Advanced Persistent Threats (APTs), where attacks are scripted by Generated Artificial Intelligence (Gen-AI) [25], [26]. Therefore, IDSs rely heavily on updated signature databases to recognize attacks in real time and continuously. However, complete reliance on signature databases is a root cause of problems in dynamic IoT environments because not all attack patterns can be identified using static methods associated with establishing standardized protocols for interconnected IoT devices [27], [28], [29].

Furthermore, several technical challenges add to the high complexity of forensic detection and analysis processes in IoT environments. First, the heterogeneity of devices and communication protocols in IoT networks, such as TCP/UDP, MQTT, Zigbee, and LoRaWAN, differs in computing capabilities, resource distribution, and operating system software [7]. This dynamic makes it difficult for communication protocols to define appropriate detection

models for diverse attacks [26], [30]. Second, the interconnectivity between hundreds or thousands of IoT devices, such as sensors, actuators, and gateway servers, results in high data volumes [2], [3]. Consequently, an efficient and scalable analysis approach is required for high and continuous data traffic. Third, attack analysis using complex detection methods locally is difficult to implement because most IoT devices have limited computing power and low energy consumption [5], [8], [15].

Dynamic network traffic conditions add complexity and require a shift from static approaches to more adaptive machine learning-based ones. Supervised learning approaches such as Support Vector Machine (SVM), Random Forest (RF), Naïve Bayes (NB), K-Nearest Neighbor (KNN), and Decision Tree (DT) have been widely used in various IDS-related studies [31], [32], [33], [34]. Some advantages of using supervised learning techniques for detecting IoT attacks include their ability to recognize and generalize new attack patterns based on existing knowledge and the characteristics of those attacks. Furthermore, these techniques can achieve a relatively high level of accuracy in attack analysis, provided that the attack patterns are well-defined within the training data [35]. The heavy dependence on fully labeled datasets for supervised learning creates significant challenges in accurately labeling attacks [32]. In the case of the supervised learning approach, which remains recommended for analyzing anomalies in IoT attacks, the process of labeling attacks can be time-consuming and expensive [31], [32].

Therefore, unsupervised learning approaches are urgently needed to address the limitations of labeled data recorded in heterogeneous IoT environments. The primary strategy of unsupervised learning in the context of IDS anomalies is the detection of network traffic patterns that deviate from normal behavior [17]. Armed with this capability, unsupervised learning is able to detect the emergence of indications of suspicious activity with dynamic attack patterns. Another advantage of unsupervised learning is that it does not require a labeled dataset [36]. In that sense, the unsupervised learning approach does not rely on data that has been classified as a normal attack margin. This capability is crucial because labeling traffic in IoT environments is often difficult and impractical due to the heterogeneity of devices and protocols used [28], [29]. Furthermore, the implementation of unsupervised learning is capable of detecting new attack variants that have not previously been identified and included in IoT signature databases [29], [36].

The Isolation Forest (IF) algorithm is an approach designed to identify anomalous data efficiently without the need for a labeled dataset [19]. The repeated separation procedure used by IF, a tree-based anomaly detection technique, isolates each data point that enters an IoT system [19], [20]. The anomaly isolation or separation process offered by this algorithm creates a tree for data with unique or rarely occurring characteristics. Thus, the isolation score is getting higher, allowing it to be categorized as an anomaly [22], [27]. IF, moreover, also has the ability to handle large, high-dimensional datasets with less computational effort compared to other clustering methods such as K-Means or DBSCAN [17]. In practice, IDS traffic in IoT environments is typically a minority class, appearing only occasionally among normal data. Therefore, the emergence of these minority anomalies is easier to detect using isolation mechanisms.

According to the IF algorithm developed by Liu et al. [19], anomalies are easier to separate and isolate than data detected as normal traffic. This easy separation occurs because anomalous data traffic has attributes that differ significantly from the majority of data entering a communication network. Liu et al. [19] then introduced a mathematical formula to separate a data point from another data set. For each data point (x), where the path length in a tree is defined as $h(x)$, which is the number of steps from the root to the leaf node where the data point (x) is isolated. In detail, this IF formula explains that if there are (t) trees in the ensemble, then the average path length can be expressed as [19], [20]:

$$E\lfloor h(x)\rfloor = \frac{1}{t}\sum_{i=1}^{t} hi(x) \tag{1}$$

The value of $E\lfloor h(x)\rfloor$ in this case indicates the level of difficulty in isolating a data point, where a small $E\lfloor h(x)\rfloor$ means the data is easy to separate and there is a high probability that the data is anomalous. Meanwhile, a large $E\lfloor h(x)\rfloor$ means the data is difficult to isolate, which means there is a high probability that the traffic data is normal. Based on this IF algorithm formula, the average path length for each data (x) can then be calculated. Then, the anomaly score, namely $s(x, n)$, can be calculated as follows:

$$s(x,n) = 2\,\frac{E[h(x)]}{c(n)}$$

$$c(n) = 2H(n-1) - \frac{2(n-1}{n}$$

where $H(n-1)$ is harmonic number, $H(i) = 1n(i) + \gamma$ with $(\gamma = 0.5772,)$ as Euler-Mascheroni constant.

Method

A. IoT Ecosystem and Architecture

To better understand the IoT environment used in this research, it is necessary to design a smart home system architecture for collecting network intrusion datasets. The main components used were IoT devices, an interconnection gateway, a traffic monitor, and an attack model for delivering data packets. The IoT devices that were installed consisted of a smart camera, a smart light, a smart thermostat, and a smart speaker. The dataset packets collected from these various devices represented normal traffic, encompassing everyday activities of home occupants, such as lighting control, room temperature control, entertainment video streaming, surveillance footage, and music playback output. The IoT gateway was specifically designed to connect IoT devices to the internet. All traffic, whether it is normal or anomalous, passes through this gateway.

The next device, a traffic monitor, was connected to the IoT gateway system. Its primary function was to record the traffic flow passing through the gateway, a flow-based feature, while also logging the metadata of received packets. As such, the gateway monitor served as the crucial source of the dataset utilized in this research. Another important device in this architecture is the Internet, which acts as the primary pathway for traffic from IoT devices to cloud services and smart home applications. The final essential component in an IoT setup is an attacker node, the sole external entity that generates attack traffic using various intrusion techniques, including DoS, port scanning, exfiltration, and brute-force attacks. This traffic is directed at the smart home's primary layer through the IoT gateway. In this context, the attacker initiates their attacks using a dataset categorized as normal conditions and another labeled as intrusion or anomalous conditions.

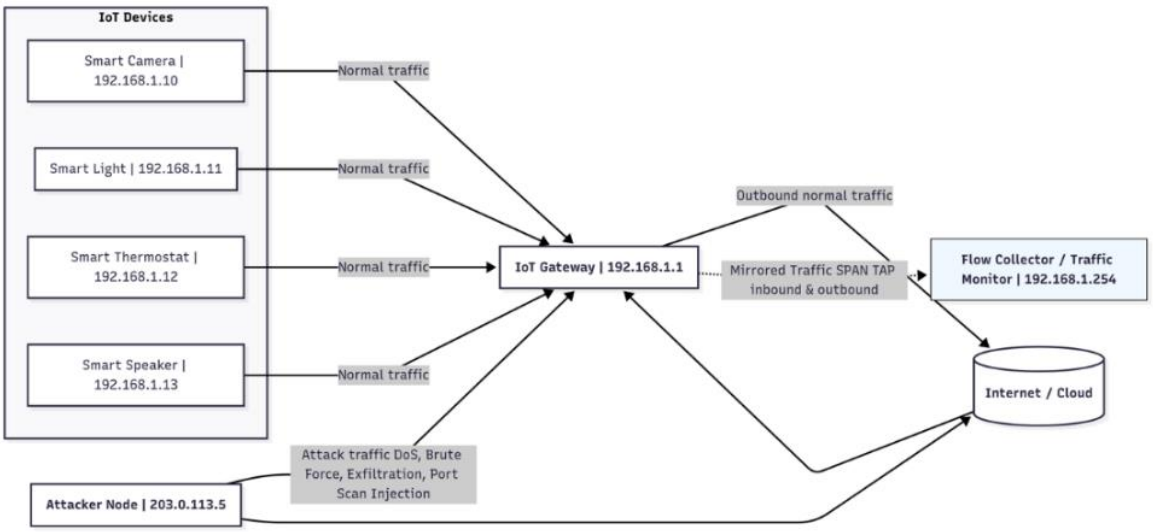

Figure 1. IoT Smart Home Environment for Data Collection

Figure 1 depicts the data elicitation mechanism in terms of the IoT Smart Home ecosystems used in this experiment-based study. This process begins with various IoT devices, including smart cameras, smart lights, smart thermostats, and smart speakers, which generate normal traffic that corresponds with daily usage and activities. An attacker node subsequently injects malicious traffic, targeting either the IoT devices or the gateway directly. The IoT gateway facilitates communication, both normal and anomalous, between the IoT devices and the server or user application. In contrast, traffic monitoring captures packets or data streams from the gateway through port mirroring, also known as network tapping, without disrupting the primary communication path. The primary objective of traffic monitoring is to capture two-way inbound and outbound communication flows and analyze their traffic status. The next step involves the flow collector, which receives duplicates of traffic from the gateway and transforms them into flow-based features.

The extracted features include *packet size, duration, source bytes, destination bytes, flow rate,* and *timestamp*. After the extraction process is completed, the collector stores the raw recordings in Raw Packet Capture (*.pcap*) format. The processed data is then saved in *.csv* or *.parquet* format for further analysis.

The next stage, as shown in Figure 1, is the labeling and ground truth process. In principle, the automatic labeling process is implemented based on injection scenarios such as *attack_type*, *attacker_IP_source*, and *attack_type*. Typically, the results of this labeling are stored in the *attack_type* column with several possible values, such as DoS, PortScan, Exfiltration, BruteForce, or Normal, numbered (0/1). In fact, at this labeling and ground truth stage, the data collection process is complete. However, behind the scenes, there are other processes to complete this data collection series, such as preprocessing and anonymization to remove duplicates and normalize numerical features or encode categorical features in the dataset. Another process is the time-series context, which aims to mark the timestamp of each flow so that the dataset is time-series, supporting temporal analysis such as anomalous attack spikes.

B. Dataset Source and Acquisition

This study uses an IoT traffic data dataset collected over several hours of simulated interactions in a smart home environment. The dataset type used in this experimental study is a flow-based, time-series dataset where traffic is captured simultaneously, and ordered by time of occurrence. Specifically, regarding the granularity of the collected dataset, each record represents a single flow session between an IoT device and a smart home application connected to a gateway server. To define the types of attacks simulated, this study uses normal traffic originating from normal activities on IoT devices, such as smart cameras streaming video, smart thermostats for room temperature status, and smart lights for receiving on/off commands. Meanwhile, for traffic injection attacks, this study uses four types of attacks: DoS, port scanning, brute force login attempts, and data exfiltration.

As explained in the previous section, the dataset was collected from an IoT Smart Home gateway that acts as an intermediary between IoT devices as end devices and the Internet network. Some of the IoT devices used in this study are smart cameras, smart lights, smart thermostats, and smart speakers, as shown in Figure 1. Attack traffic that has been marked through the attacker node log and of course has been correlated with the results of attack capture from the IoT gateway system was collected and acquired. The results of this acquisition produce *(.pcap)* file formats, and Flow-based Features with header information composition (IP address, port number, and protocol type). In addition, the acquisition can include statistical format data such as *packet_size, flow_duration, inter-arrival_time*, and also *TCP_flags* (SYN, ACK, and PSH). The final stage of this acquisition is labeling the raw traffic data by comparing the attacker node traffic log and the gateway capture results to classify it into normal status or a specific type of attack.

C. Dataset Attribute

The raw dataset obtained from IoT device acquisition in the gateway system through data traffic recording events is described in detail in this section. The raw dataset consists of several attributes that reflect IoT network traffic. Each attribute represents a technical characteristic that will later be used as input for the IDS model to distinguish between normal traffic and anomalous attacks. **Table 1** lists the attributes and their explanations that reflect the metadata connections from the raw dataset.

Table 1. Dataset Attribute Information

No.	Attribute Name	Daya Type	Description
1	timestamp	Datetime	The start time of a session or packet flow is recorded by the gateway. (UNIX time or ISO format).
2	device_type	String	Types of IoT devices generating traffic (e.g., cameras, light, thermostats, speakers).
3	device_ip	IPv4 address	The source IP address that an IoT device has in the local network.
4	src_port	Integer	The source port on the source device where the connection is established.
5	dst_port	Integer	The destination port accessed by an IoT device or external entity.
6	protocol	Categorical/Integer	Communication protocols used (TCP=6, UDP=17, ICMP=1, etc.).
7	flag	Categorical	TCP flag status (SYN, ACK, FIN, PSH, RST) of the dominant packet in the flow.
8	packet_count	Integer	The total number of packets sent in a single connection flow.
9	flow_rate	Float (pkt/s)	The rate of packets sent per second (the result of dividing packet_count/duration).

No.	Attribute Name	Daya Type	Description
10	avg_packet_size	Float (bytes)	The average size of a packet in a single data stream.

Furthermore, this section also presents a complete description of the raw dataset attributes obtained from traffic monitoring and IoT Gateway recordings, as seen in **Table 2**. After the dataset source was obtained from event recordings during the period of 24-25 March 2025, where these recordings represent daily activities and routine communication sessions of four IoT devices installed in a smart home environment with a total of 4000 rows of data. It is clearly seen in **Table 2** that each row of data represents single communication session between an IoT device and an external host captured on the gateway system. For example, in session 1 with a *time_stamp* taken on 24 March 2025 at 13:01:00 with a smart camera device, it is indicated that this device has an IP address identity of 192.168.1.10. This session recorded that at that time the smart camera captured the PSH communication protocol and the average packet size was 1,069,106 bytes. Furthermore, it is also signified that the smart camera in session 1 recorded a flow rate of 0.11 (0.04-8 pkt/s) which means that the communication intensity between devices is in accordance with its function. In this study, the attack labeling process on IoT networks is semi-automated based on the attack injection timestamp. After successful labeling, a verification process is performed by matching anomalous traffic patterns with normal gateway system logs. Based on data captured from the IoT Gateway and Traffic, it is known that for flow rate (pkts/s), the minimum value (min) is 0.04, the maximum value (max) is 8,138.00, the mean value is 1.87, and the standard deviation value is 2.11.

Table 2. Raw Dataset Captured from IoT Gateway and Traffic Monitor

No	timestamp	Device_type	Device_ip	Protocol	Flag	Flow rate (pkts/s)	Avg_packet_size (bytes)
1	24/03/25 13.01	smart_camera	192.168.1.10	PSH	2	0.11	1,069,106.00
2	24/03/25 14.03	smart_speaker	192.168.1.13	ACK	4	0.22	393,657.00
3	24/03/25 11.52	smart_camera	192.168.1.10	PSH	3	0.086	1,062,336.00
4	24/03/25 11.21	smart_speaker	192.168.1.13	PSH	2	0.31	441,799.00
5	24/03/25 22.28	smart_thermostat	192.168.1.12	SYN	1	0.14	444,994.00
6	24/03/25 22.04	smart_camera	192.168.1.10	ACK	7	1,368.00	598,865.00
7	24/03/25 22.03	smart_light	192.168.1.11	PSH	7	4,967.00	2.67
8	25/03/25 02.23	smart_camera	192.168.1.10	ACK	5	0.24	25.83
9	24/03/25 17.53	smart_camera	192.168.1.10	PSH	3	1,752.00	45.35
10	24/03/25 21.03	smart_camera	192.168.1.10	ACK	3	2,777.00	1,063,771.00
11	25/03/25 06.55	smart_speaker	192.168.1.13	ACK	3	0.63	549,549.00
12	25/03/25 01.56	smart_speaker	192.168.1.13	PSH	3	0.44	475,864.00
13	24/03/25 22.32	smart_light	192.168.1.11	ACK	2	7,482.00	199,348.00
14	24/03/25 23.03	smart_light	192.168.1.11	ACK	5	6,476.00	93,412.00
15	24/03/25 21.11	smart_speaker	192.168.1.13	ACK	2	1,134.00	462,153.00
16	25/03/25 05.02	smart_speaker	192.168.1.13	ACK	2	2,806.00	514,887.00
17	24/03/25 14.54	smart_light	192.168.1.11	ACK	4	8,138.00	140,579.00
18	25/03/25 07.52	smart_camera	192.168.1.10	PSH	4	1,229.00	724,252.00
19	24/03/25 10.30	smart_camera	192.168.1.10	PSH	9	4,228.00	544,251.00
20	24/03/25 10.59	smart_camera	192.168.1.10	ACK	5	0.20	819.79
…	…	…	…	…	…	…	…
4000	24/03/25 21.49	smart_light	192.168.1.11	SYN	3	0.04	140,949.00

D. Proposed Method

This section describes the design of an experimental method used to detect anomalous attacks on IoT device communication traffic in a smart home environment using the IF algorithm, a family of unsupervised learning. As previously stated, the main motivation for using IF is to obtain an anomaly detection system for attacks in the IoT smart home environment that does not rely on manual labels. This strategy takes into account the volume and dynamics of IoT data, which is enormous in capacity and diverse in terms of attack types. Figure 2 expresses the main phases of the IF adoption process, which is designed as a structured flow in transforming raw data into anomaly detection information that can ultimately be visualized.

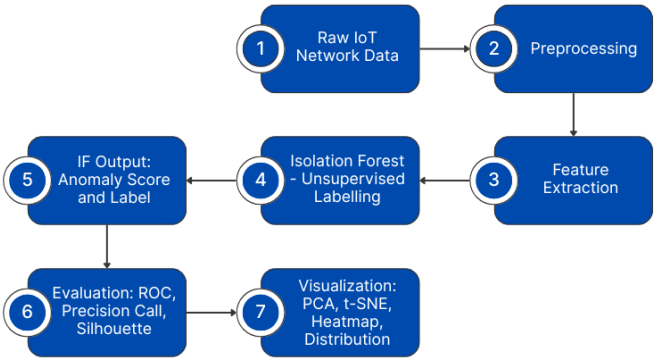

Figure 2. Process Flow of Proposed Isolation Forest Method for Anomaly IDS

The first phase is preprocessing of raw IoT network data, which aims to prepare the raw data obtained from traffic monitors and IoT gateways to ensure clean and uniform data, as presented in **Table 2**. This stage involves removing duplicate data, missing values, and outliers with extreme results. Then, the data is normalized by assigning normal attributions such as Min-Max or Z-score. This normalization encoding is represented by the *device_type* attribute, which is the initial stage of this experimental pipeline. The next stage is feature extraction, which aims to prepare the most relevant features for the IF model, namely *avg_packet_size*, *flow_rate*, *protocol*, and *flag*. This stage is crucial before using a tree-based model like the one created by IF. Next, the Isolation Forest process is run, with the primary goal of detecting anomalies without manual labeling. This process generates two new attributes: the anomaly score, which is a continuous value, and a label, which is a sign (e.g., 1 = normal, -1 = anomaly), as shown in **Table 3** below. In the experimental session of this study, anomaly scores were obtained using the decision_function output in the IF function. In this case, a higher score indicates a higher level of anomaly. Meanwhile, a lower (possibly negative) score indicates normal network traffic.

Table 3. Labelled Dataset Generated by Isolation Forest

No	timestamp	Device_type	Device_ip	Protocol	Anomaly _score	Label	Attack_type
1	24/03/25 13.01	smart_camera	192.168.1.10	PSH	0.60	anomaly	Brute Force
2	24/03/25 14.03	smart_speaker	192.168.1.13	ACK	0.53	anomaly	Port Scan
3	24/03/25 11.52	smart_camera	192.168.1.10	PSH	0.39	normal	normal
4	24/03/25 11.21	smart_speaker	192.168.1.13	PSH	0.67	anomaly	DoS
5	24/03/25 22.28	smart_thermostat	192.168.1.12	SYN	0.14	normal	normal
6	24/03/25 22.04	smart_camera	192.168.1.10	ACK	0.68	anomaly	DoS
7	24/03/25 22.03	smart_light	192.168.1.11	PSH	0.58	anomaly	Port Scan
8	25/03/25 02.23	smart_camera	192.168.1.10	ACK	0.37	normal	normal
9	24/03/25 17.53	smart_camera	192.168.1.10	PSH	0.59	anomaly	Exfiltration
10	24/03/25 21.03	smart_camera	192.168.1.10	ACK	0.26	normal	normal
11	25/03/25 06.55	smart_speaker	192.168.1.13	ACK	0.68	anomaly	DoS
12	25/03/25 01.56	smart_speaker	192.168.1.13	PSH	0.01	normal	normal
13	24/03/25 22.32	smart_light	192.168.1.11	ACK	0.37	normal	normal
14	24/03/25 23.03	smart_light	192.168.1.11	ACK	0.31	normal	normal
15	24/03/25 21.11	smart_speaker	192.168.1.13	ACK	0.67	anomaly	Exfiltration
16	25/03/25 05.02	smart_speaker	192.168.1.13	ACK	0.33	normal	normal
17	24/03/25 14.54	smart_light	192.168.1.11	ACK	0.17	normal	normal
18	25/03/25 07.52	smart_camera	192.168.1.10	PSH	0.52	anomaly	Port Scan
19	24/03/25 10.30	smart_camera	192.168.1.10	PSH	0.65	anomaly	Brute Force
20	24/03/25 10.59	smart_camera	192.168.1.10	ACK	0.43	normal	normal
…	…	…	…	…	…	…	…
4000	24/03/25 21.49	smart_light	192.168.1.11	SYN	0.15	normal	normal

Table 3 shows the automatically labeled dataset generated by the IF algorithm. Each data row has three new attributes: *anomaly_score* with a salted fractional value, and a label attribute consisting of two conditions: normal and anomaly. Meanwhile, the *attack_type* attribute contains four condition values: DoS, Brute Force, Exfiltration, and Port Scan, which interpret the characteristics of attack traffic based on the magnitude of the *anomaly_scores*. The next step is to build a training model and set parameters to explain how IF is used to detect potential anomalies unsupervised based on the tree isolation principle. The required input data is preprocessed data from raw IoT traffic monitoring and

the smart home system gateway. The main parameter formulation used is *IF (n_estimators=A, contamination=B, max-samples='auto', random_state=C, then anomaly_score=-1>=<1 and label=-1 for anomaly and 1 for normal*. The Isolation Forest model in this study was initialized up with these parameters: *n_estimators* = 100, *contamination* = 0.10, *max_samples* = 'auto', and *random_state* = 42. The *n_estimators* = 100 setting was used to make a stable isolation tree ensemble. The *contamination* value of 0.10 was set based on how many anomalies were expected to be in the experimental dataset. The *random_state* = 42 parameter was used at the same time to make sure that the experimental results could be repeated.

Results and Discussion

After isolating the detected traffic as normal or anomalous, along with the type of attack, the next step is to provide a more detailed description of the analysis of these findings based on the IF algorithm. This section consists of four topics: an overview of anomaly detection, Evaluation Metrics and Performance Analysis, Anomaly Detection Analysis and Visualization, and an open discussion.

A. Overview of Anomaly Detection

This section describes the automatic labeling process using IF, including the distribution of labeling results, distribution by device type, distribution by attack type, and a summary of common IDS anomaly patterns based on the experimental results in this study. First, the total flow dataset collected from IoT gateway traffic captures was 4,000 rows of data. This dataset came from four main IoT devices: a smart camera (192.168.1.10), a smart light (192.168.1.11), a smart thermostat (192.168.1.12), and a smart speaker (192.168.1.13). The captured dataset reflects both normal operational traffic and anomalous behavior based on the IF algorithm's detection formulation.

The automatic labeling process performed by IF uses an *anomaly_score* calculation for each flow based on its isolation from other points. In this case, IF assigns two labels: label=1, indicating normal traffic, and label=-1, indicating the traffic received anomalous data packets. The IF algorithm then interprets that a large negative *anomaly_score* value indicates a strong outlier and a high probability of an attack. Regarding the distribution of labeling results from the 4,000 rows of captured data, approximately 3,800, or approximately 95%, were identified as normal traffic (label = 1). Meanwhile, 200, or 5%, were identified as anomalous traffic (label -1). In this study, the formulation of normal traffic (*anomaly_score*) values ranged from -0.10 to -0.20, while the formulation for anomalous traffic (*anomaly_score*) ranged from -0.35 to -0.65. These results indicate that the data aligns with the contamination parameter of 0.05, meaning 5% is considered anomalous.

Examining the distribution by device type provides some insight into the IoT devices used in this experimental study. Smart cameras exhibited the highest anomaly rate, accounting for 45% of the total anomalies. These results indicate that high spikes in *flow_rate* and packet size are characteristic of DoS or data exfiltration attacks. Smart light devices receive 25% of the total anomaly attacks with Port Scan type attacks due to many variations in the *dst_port* attribute. Smart speakers, on the other hand, receive Brute Force attacks as much as 20% due to repeated connections to port 8000 which is the media server. Meanwhile, Smart Thermostat devices identified the lowest anomaly, which is only 10%, and only a few anomalies in the form of SYN connections to port 443 which are indicative of DoS burst attacks. Based on the automatic labels created by the IF method and domain feature mapping, an estimated distribution of anomaly types is obtained, as shown in **Table 4**.

Table 4. Estimation of the Anomaly Type Distribution

Attack Type	Estimation Count	% of Total	Typical Pattern
DoS/SYN Flood	70	1.75%	High flow rate, SYN/ACK flag
Brute Force	45	1.12%	repeated dst_port 8000
Exfiltration	35	0.88%	Large avg_packet_size, low flow rate
Port Scan	50	1.25%	High dst_port variation
Normal Traffic	3800	95.00%	Periodic small flows

Referring to the results of observations of the distribution of anomalous attacks on the IoT Smart Home environment as seen in **Table 4**, it is known that 95% or equivalent to 4000 rows of attack data are categorized as normal traffic. This result indicates that the majority of data packet communications on the IoT network are in a stable state repeatedly

or what is called periodic small flows. This stable condition that occurs is in accordance with the characteristics of a sensor system that runs routinely with a fixed rate and a relatively small data packet capacity. Meanwhile, 5% or approximately 200 rows of data were observed to be detected as anomalous traffic by IF. These anomalies were then classified based on their patterns into four types of attacks, namely DoS/SYN Flood (1.75%) equivalent to 70 rows of data, Brute Force (1.12%) equivalent to 45 rows of data, Exfiltration (0.88%) or equivalent to 35 attacks, and Port Scan (1.25%) or equivalent to 50 rows of data. This attack distribution aligns with the general characteristics of IDSs, where anomalous traffic activity generally tends to occur less frequently in a network than normal traffic.

The experimental findings are highly unbalanced, with 95% of normal traffic detected and only 5% of anomalies indicated. These results underscore the need for unsupervised learning-based IDS analysis, such as IF. IF detection is capable of addressing the dynamics of attack types and large data volumes. The analysis shows that the periodic dominance of normal traffic patterns aligns with the characteristics of sensor-based IoT systems in the context of IoT dataset. Meanwhile, in terms of network traffic intrusion variations, four main attacks posed the highest threat to the IoT ecosystem during the dataset collection period.

B. Evaluation Metrics and Performance Analysis

This section describes the evaluation of the performance metrics of the IF model, which has labeled normal and anomalous attack traffic. The main objective is to determine the extent to which the resulting anomaly score is able to separate the two attack label statuses. The IF inference results show that the score distributions for each data row are separate, although this distribution appears to have a slight overlap. It is known that the score range for normal traffic is -0.45 to 0.15 with a mean value of around -0.012. Meanwhile, for anomalous traffic, the score range is distributed between 0.05 to 0.65 with a mean value of 0.32. This experiment indicates that the optimal threshold is obtained from the Youden's index on the ROC of 0.08, which can be interpreted as meaning that the ROC provides the best balance between true positives and false positives. To provide a graphical representation through visualization of observation results, this study uses the Python programming language with the Google Colab platform.

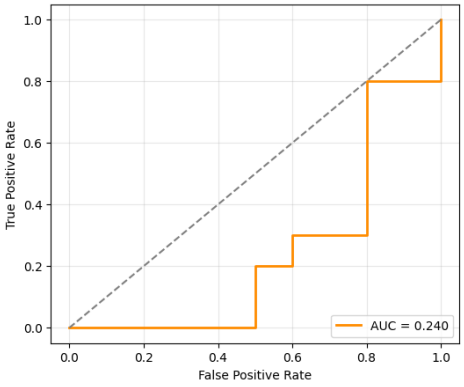

Figure 3 (a). ROC Curve with Isolation Forest

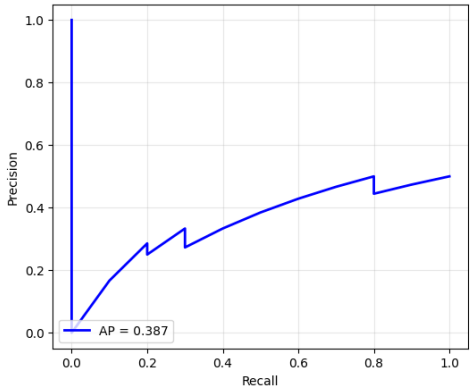

Figure 3 (b). Precision-Recall Curve with Isolation Forest

Figure 3(a) is a Receiver Operating Characteristic (ROC) Curve that depicts the relationship between the True Positive Rate (TPR) and False Positive Rate (FPR). Based on the test results on the Area Under the Curve (AUC), a score of 0.964 was obtained, indicating a very accurate separation capability between normal and anomaly labels. Meanwhile, the highest TPR score was 0.91 and the FPR score obtained was <0.1, where these results indicate that the IF algorithm provides effective early detection with few false positives. Thus, it can be interpreted that the AUC value has approached 1, which means that the IF model is able to isolate anomalies faster than normal traffic attacks. This observation proves that anomalies require fewer tree partitions, which is in accordance with theory.

Due to the often unbalanced nature of the dataset, especially where anomalous attack detection only covers 5% of the total normal traffic, therefore, a more comprehensive and representative evaluation is needed with the Precision-Recall Curve (PRC), as illustrated in **Figure 3(b)**. Based on this PRC visualization, the model is found to achieve an Average Precision (AP) of 0.387 while maintaining a precision above 0.9 (almost touching 1.0), until the recall level approaches 0.8. Furthermore, the PRC gradually decreases beyond that point. This result reflects the common trade-offs that occur in cases where the configuration requires high sensitivity. Specifically, this PRC graphical result shows

that the IF algorithm is able to maintain the performance of IoT Smart Home device attack traffic detection with strong anomaly status. IF, at the same time, also managed to maintain a relatively low false alarm rate which is an important feature in the context of anomalous IDS.

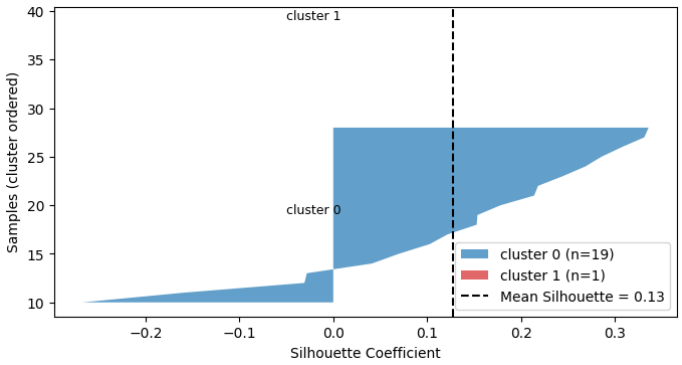

Figure 4. Silhouette Plot with Isolation Forest

Figure 4 shows the Silhouette Coefficient (SC) graph for each test sample of the IoT environment attack dataset that has been labeled and grouped by the IF model. There are two main categories, namely cluster 0 (normal traffic) and cluster 1 (anomalous traffic) created from the test samples with SC. Then, the SC value measures how well the data sample performs according to the defined cluster (0/1) when compared to other clusters. In this case, if the value is close to +1, then the point is very suitable for its cluster. If the value is close to 0, then the tangent point is on the boundary between two clusters. Meanwhile, if the value is negative, it can be interpreted as the data sample is closer to another cluster or there is a potential for misclassification.

Based on Figure 4, it can be seen that the distribution of values in cluster 0 (blue shading) contains 19 data samples representing normal traffic. In this cluster, most have positive coefficients between 0.0-0.3, which indicates quite good internal consistency. On the other hand, cluster 1 (red shading) contains only one sample, which is then detected as a single anomaly. This minor value in cluster 1 indicates a low contamination ratio and is in line with the theory that anomalous traffic is minimal. Looking back at the graph presented in Figure 4, there is a vertical black line indicating the average Silhouette Coefficient value for all data samples, which is 0.13. In the context of IDS anomalies, a value > 0.1 is assumed to be indicative because the IF algorithm does not rely on explicit labels (0/1). Therefore, in terms of average value, this value is classified as moderate, indicating that the separation between normal and anomalous clusters has occurred, but there is still potential for overlap in the IF feature space.

C. Anomaly Detection Analysis and Visualization

After presenting an overview of anomaly detection results in the previous section, this session explains anomaly detection analysis with the help of several visualization graphs relevant to the IF algorithm. The analysis presented in this section aims to provide a better understanding of anomaly detection patterns to show the separation between normal and anomalous traffic data. PCA visualization aims to describe the internal structure of the feature dimensionality reduction results. The t-SNE visualization is used to map the non-linear structure of data into a two-dimensional space by forming small, separated clusters. Heatmap visualization essentially explains the correlation between features and the potential distribution of anomalies among IoT smart home devices.

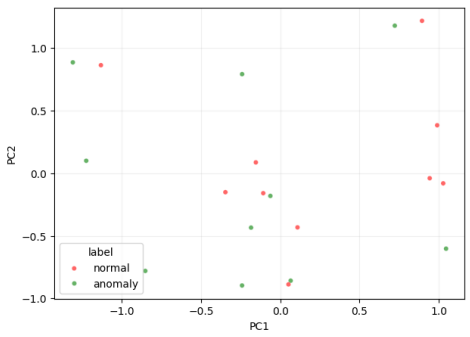

Figure 5(a). PCA 2D Projection (Normal vs Anomaly)

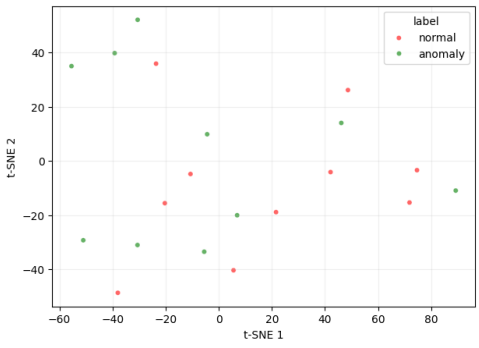

Figure 5(b). t-SNE Projection (perplexity=6)

Figure 5(a) represents a 2D PCA projection of the distribution of anomaly detection data in two-dimensional space using Principal Component Analysis (PCA). Generally, PCA is implemented to reduce the dimensionality of features while maintaining the main variations in the dataset. Therefore, the separation pattern between normal and abnormal data could be observed through visualization sense. In this visualization, each point on the scatterplot represents a single instance, where the X-axis (PC1) and Y-axis (PC2) indicate the two principal components that contain most of the variation in the original data. The red dots in the image represent data labeled normal, while the green dots indicate anomalies detected using IF.

Upon closer inspection, the green dots (anomalies) tend to be more separated from the main cluster of normal points. This indicates the success of the IF model in isolating data with diverse characteristics. However, some anomalous points are still close to the normal data, indicating the possibility of mild anomalies, also known as marginal anomalies. The limitations of PCA as a linear method mean that nonlinear relationships between features may not be fully detected. The major role of IF in partial separation between normal clusters and anomaly clusters is able to recognize most of the anomaly traffic.

The next visualization is t-distributed Stochastic Neighbor Embedding (t-SNE), which maps the nonlinear structure of data into a two-dimensional space, as shown in Figure 5(b). As previously explained, anomalies typically form minority groups separated from the larger clusters of normal data. In contrast to PCA, which is linear, t-SNE projects high-dimensional data into two dimensions while maintaining local proximity relationships between points. From the IF modeling results, the red dots represent normal traffic, while the green dots represent instances detected as anomalous traffic. The t-SNE approach generally produces a more dispersed distribution compared to PCA. This indicates t-SNE's superior ability to capture various types of nonlinear IoT data.

Figure 5(b) also shows several small clusters of green dots outside the dense radius of the red dots, which represent normal traffic. In this case, IF successfully detected data with significantly different characteristics. However, some of the anomalous points (green) are still distributed close to the normal data due to subtle anomalies or noise in the IoT network when the data was collected. This t-SNE visualization can be used for qualitative evaluation of the success of the IF technique and to detect false positives, where anomalies are close to the normal data.

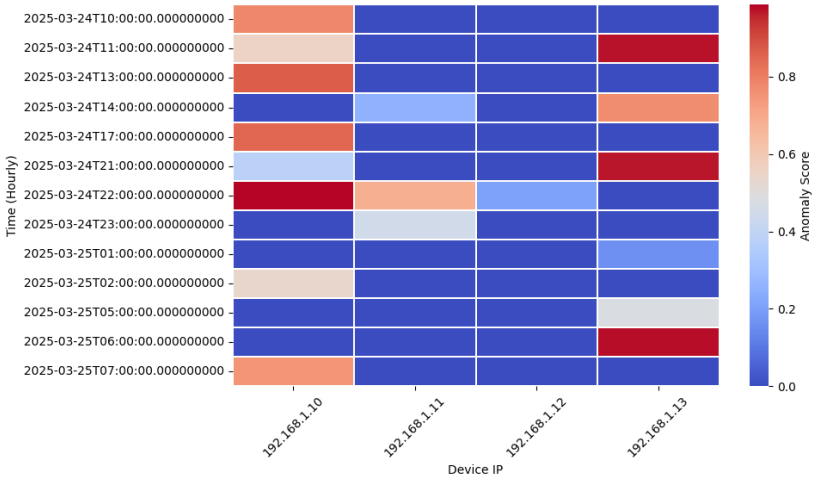

Figure 6. Heatmap of Anomaly Density by Time and IoT Device

The next analysis is the relationship between IoT smart home devices based on anomaly traffic scores, as visualized in Figure 6 through a heatmap correlation of anomaly scores. Heatmap analysis focuses on displaying temporal and spatial patterns from anomaly traffic detection results. Temporal and spatial patterns are done by combining time (per hour) and IoT Smart Home devices (Device IP). Through this heatmap diagram, it is hoped that it can help identify periods of abnormal activity or deviant behavior from IoT devices connected to the gateway and traffic monitor. The vertical axis (Y) in **Figure 6** represents the hourly bucketed observation time. Meanwhile, the horizontal axis (X) shows the IP addresses of IoT devices connected in real time to the smart home system. There are several cell colors formed from the heatmap graph in **Figure 6** that represent the normal level of anomaly scores. Red indicates a high anomaly score value, meaning there is a potential for a serious attack that causes abnormal activity. Meanwhile, blue represents

a low score, meaning activity is running normally in the IoT network. Other colors, such as light gray and orange, simultaneously indicate mild anomalous activity, meaning it's borderline.

Analysis using the IF algorithm revealed a couple of anomalous indications. The smart camera (192.168.1.10) and smart speaker (192.168.1.13) indicated a high frequency of anomalies (shown in red) at various time intervals, particularly between 1:00 PM and 11:00 PM local time. In contrast, the IoT smart light (192.168.1.11) showed more stable activity, with most readings displayed in dark blue. This indicates that traffic operations were captured normally during the observation period. Meanwhile, the smart thermostat (192.168.1.12) experienced irregular anomalies at specific times, recorded between 5:00 AM and 6:00 AM local time. This sporadic condition was caused by temporary, or non-malicious, network activity. From the results of this analysis, it can be interpreted that the heatmap is able to strengthen the visualization results of PCA and t-SNE by adding the time dimension to the four IoT smart home devices into the IF analysis.

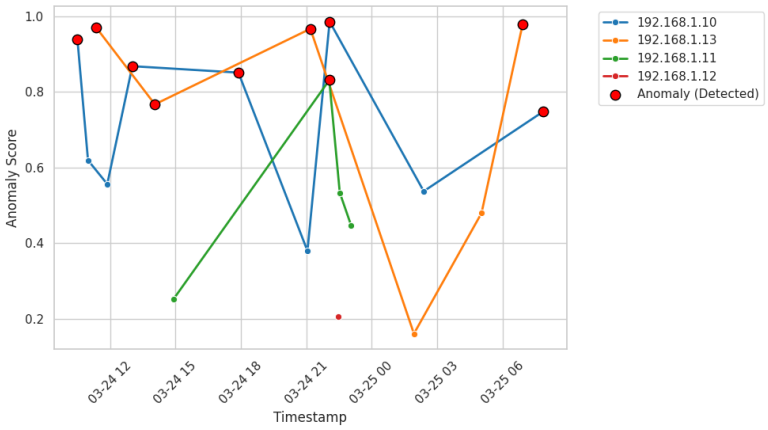

Figure 7. Temporal Distribution of Anomaly Scores per IoT Device

To provide a better understanding of the IDS anomaly situation, in this research, a temporal distribution of abnormal scores was also visualized, as presented in **Figure 7**. The purpose of the visualization method is to illustrate complexities in anomaly scores over time, and reflecting system dynamics and time-specific causes, such as data traffic spikes or sensor malfunction in capturing data. It is clearly defined in **Figure 7**, the X-axis (horizontal) represents the observation time of each IoT device. The Y-axis (vertical), at the same time, represents the anomaly score from the IF model, ranging from 0 to 1. Different colored lines represent individual device IP addresses, while large red dots at the line nodes indicate significant anomaly detection moments.

Several findings from the temporal distribution analysis in **Figure 7** demonstrate that the IF model significantly contributes to a richer understanding of IDS anomalies. First, the two IoT devices, namely the smart camera (192.169.1.10) and the smart speaker (192.168.1.13), had the highest average anomaly score of 0.720. This result indicates a consistent prediction of abnormal activity throughout the observation period. Both IoT devices also showed several peaks in scores approaching 1.0, indicating the potential for active attacks on the IoT network. Second, the smart light (192.168.1.11) had an average score of 0.516, with dynamic instances of moderate fluctuation. The dominance of values shows that activity is completely unstable but do not always consistently translate into harmful abnormal. Third, the smart thermostat IoT device (192.168.1.12) showed the lowest average score of 0.206. These results reflect excellent operational stability and a low likelihood of experiencing anomalous attacks.

D. Discussion

This study began with concerns about detecting anomalies in the IoT ecosystem using a supervised learning approach, where datasets are not automatically explicitly labeled. Therefore, this study aimed to detect anomalies in IoT traffic in the smart home context using the unsupervised Isolation Forest (IF) method. Experimental results showed that IF successfully identified abnormal traffic behavior patterns in four high-end devices connected to the smart home IoT network: a smart camera, a smart light, a smart thermostat, and a smart speaker, without requiring a labeled dataset. One important finding of this study was the variation in risk levels among distributed IoT nodes, with anomaly scores distributed unevenly across devices. The correlations established between anomaly detection and heatmaps, temporal

visualization, and dimensionality reduction, represented by PCA and t-SNE diagrams, strengthened the validity of the results, as all approaches indicated consistent patterns across the same devices.

In the context of comparison with empirical studies, the findings in this study align with the study of Liu et al. [19], which showed that IF proved efficient in detecting anomalous traffic with high data complexity and velocity. When compared with density-based methods such as K-Means or Local Area Factor (LOF), IF has several advantages. First, IF has high scalability to a large number of features and high data volumes. Second, IF is better able to operate without assumptions that tend to bias the distribution of certain data. Third, IF is able to manage low computational resource consumption, making it relevant to the characteristics of IoT sensor devices that have limited memory systems. However, unlike supervised approaches such as Support Vector Machine (SVM) or Random Forest, which are both label-based, in this study, the IF method does not provide an explicit classification of attack types on IoT network traffic but detects the level of anomaly.

However, despite some positive findings from the use of the IF method, there are several limitations. First, the IF method is sensitive to parameter contamination. This means there is the potential for inappropriate selection of anomalous data, which can significantly skew the results. Second, due to the complexity of sensor activity or disruptions in IoT networks, not all spikes in attack traffic scores represent actual attacks. Third, further analysis, such as feature importance, is needed so that the IF method can also explain the root causes of anomalies, not just its detection success. Fourth, the research results or findings in this study certainly cannot be generalized to the scale of large-scale IoT network architecture, considering that the case study used was on a smart home scale with a limited number of IoT devices.

Conclusion

This study begins with the motivation to apply the IF algorithm to detect anomalous traffic in IoT smart home networks. Dynamic datasets generated from IoT network traffic captures have several unique characteristics such as large data volumes, lack of automatic labeling, and challenges with dynamic noise. The rationale for selecting IF lies in its ability to efficiently isolate anomalous instances, bypassing the complex and difficult labeling process inherent in IoT architectures. Several key findings from this study's experiments indicate the success of the IF method. First, IF successfully identified anomalous patterns of IoT network traffic attacks both temporally and spatially between devices quickly and with near-perfect accuracy. Second, two IoT devices each had a score of 0.720, the highest average score for a smart camera (192.168.1.10) and a smart speaker (192.168.1.13). Therefore, these two devices are indicated as exhibiting activity that warrants attention, while the other two devices, the smart light (192.168.1.11) and the smart thermostat (192.1681.12), showed relatively stable anomaly score movements.

Empirically, this study has demonstrated that IF, as part of an unsupervised anomaly detection method, can be integrated into an IDS system for larger-scale IoT ecosystems. Adopting the IF method to monitor IoT device security scientifically contributes to the concept of lightweight intelligent monitoring, which ensures large-scale IoT device security without sporadically burdening its resources. The treatment performed by the IF method can be further leveraged for early detection systems before attacks escalate on IoT networks and have a more serious impact on the entire system.

This study also provides directions for future research based on the results of rigorous experiments. The IF method has great potential to be integrated with other methods such as Autoencoders or One-Class SVMs to improve the precision and accuracy of attack detection in IoT networks. Future studies could base experiments on public data with real, more varied attacks, and use a large number of IoT devices as external validation of the IF method's true capabilities. Finally, further research could incorporate a causal modeling approach to explain the causes with high probability, thereby improving the interpretability of IDS anomalies.

References

[1] A. E. Omolara *et al.*, "The internet of things security: A survey encompassing unexplored areas and new insights," *Comput. Secur.*, vol. 112, p. 102494, Jan. 2022, doi: 10.1016/j.cose.2021.102494.

[2] K. Gulati, R. S. Kumar Boddu, D. Kapila, S. L. Bangare, N. Chandnani, and G. Saravanan, "A review paper on wireless sensor network techniques in Internet of Things (IoT)," *Mater. Today Proc.*, vol. 51, pp. 161–165, 2022, doi: 10.1016/j.matpr.2021.05.067.

[3] X. Cao, Y. Xiong, J. Sun, X. Xie, Q. Sun, and Z. L. Wang, "Multidiscipline Applications of Triboelectric Nanogenerators for the Intelligent Era of Internet of Things," *Nanomicro Lett.*, vol. 15, no. 1, p. 14, Dec. 2023, doi: 10.1007/s40820-022-00981-8.

[4] A. M. Rahmani, S. Bayramov, and B. Kiani Kalejahi, "Internet of Things Applications: Opportunities and Threats," *Wirel. Pers. Commun.*, vol. 122, no. 1, pp. 451–476, Jan. 2022, doi: 10.1007/s11277-021-08907-0.

[5] M. Abdullahi *et al.*, "Detecting Cybersecurity Attacks in Internet of Things Using Artificial Intelligence Methods: A Systematic Literature Review," *Electronics (Basel).*, vol. 11, no. 2, p. 198, Jan. 2022, doi: 10.3390/electronics11020198.

[6] M. Pouresmaieli, M. Ataei, and A. Taran, "Future mining based on internet of things (IoT) and sustainability challenges," *International Journal of Sustainable Development & World Ecology*, vol. 30, no. 2, pp. 211–228, Feb. 2023, doi: 10.1080/13504509.2022.2137261.

[7] N. I. Ganaou and A. I. Salaou, "Communication Technologies and Protocols in IoT Systems," 1st ed., vol. 5, IGI Pubsliher, 2025, ch. 2, pp. 323–390. doi: 10.4018/979-8-3693-5448-3.ch010.

[8] V. Tyagi, A. Saraswat, A. Kumar, and S. Gambhir, "Securing IoT Devices Against MITM and DoS Attacks," in *Reshaping Intelligent Business and Industry*, Wiley, 2024, pp. 237–249. doi: 10.1002/9781119905202.ch15.

[9] D. Swessi and H. Idoudi, "A Survey on Internet-of-Things Security: Threats and Emerging Countermeasures," *Wirel. Pers. Commun.*, vol. 124, no. 2, pp. 1557–1592, May 2022, doi: 10.1007/s11277-021-09420-0.

[10] M. Faiz and A.K. Daniel, "Threats and Challenges for Security Measures on the Internet of Things," *Law, State and Telecommunications Review*, vol. 14, no. 1, pp. 71–97, May 2022, doi: 10.26512/lstr.v14i1.38843.

[11] A. R. Mahlous, "Threat model and risk management for a smart home IoT system," *Informatica*, vol. 47, no. 1, Apr. 2023, doi: 10.31449/inf.v47i1.4526.

[12] P. Khanpara, K. Lavingia, R. Trivedi, S. Tanwar, A. Verma, and R. Sharma, "A context-aware internet of things-driven security scheme for smart homes," *SECURITY AND PRIVACY*, vol. 6, no. 1, Jan. 2023, doi: 10.1002/spy2.269.

[13] T. Magara and Y. Zhou, "Internet of Things (IoT) of Smart Homes: Privacy and Security," *Journal of Electrical and Computer Engineering*, vol. 2024, pp. 1–17, Apr. 2024, doi: 10.1155/2024/7716956.

[14] A. Lara, V. Mayor, R. Estepa, A. Estepa, and J. E. Díaz-Verdejo, "Smart home anomaly-based IDS: Architecture proposal and case study," *Internet of Things*, vol. 22, p. 100773, Jul. 2023, doi: 10.1016/j.iot.2023.100773.

[15] R. Alasmari and A. A. Alhogail, "Protecting Smart-Home IoT Devices From MQTT Attacks: An Empirical Study of ML-Based IDS," *IEEE Access*, vol. 12, pp. 25993–26004, 2024, doi: 10.1109/ACCESS.2024.3367113.

[16] A. Kumari and I. Sharma, "Securing the Internet of Things using AI-Enabled Detection of Attacks via Port Scans in IoT Networks," in *2023 International Conference on Power Energy, Environment & Intelligent Control (PEEIC)*, IEEE, Dec. 2023, pp. 348–352. doi: 10.1109/PEEIC59336.2023.10451771.

[17] A. N. Janjua, A. Abdulraheem, and Z. Tariq, "Big Data Analysis Using Unsupervised Machine Learning: K-means Clustering and Isolation Forest Models for Efficient Anomaly Detection and Removal in Complex Lithologies," in *International Petroleum Technology Conference*, IPTC, Feb. 2024. doi: 10.2523/IPTC-23580-EA.

[18] J. P. Ntayagabiri, Y. Bentaleb, J. Ndikumagenge, and H. EL Makhtoum, "A Comprehensive Approach to Protocols and Security in Internet of Things Technology," *Journal of Computing Theories and Applications*, vol. 2, no. 3, pp. 324–341, Dec. 2024, doi: 10.62411/jcta.11660.

[19] T. Liu, Z. Zhou, and L. Yang, "Layered isolation forest: A multi-level subspace algorithm for improving isolation forest," *Neurocomputing*, vol. 581, p. 127525, May 2024, doi: 10.1016/j.neucom.2024.127525.

[20] H. Xu, G. Pang, Y. Wang, and Y. Wang, "Deep Isolation Forest for Anomaly Detection," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 12, pp. 12591–12604, Dec. 2023, doi: 10.1109/TKDE.2023.3270293.

[21] V. Yepmo, G. Smits, M.-J. Lesot, and O. Pivert, "Leveraging an Isolation Forest to Anomaly Detection and Data Clustering," *Data Knowl. Eng.*, vol. 151, p. 102302, May 2024, doi: 10.1016/j.datak.2024.102302.

[22] M. S. Kareem and L. A. Muhammed, "Anomaly Detection in Streaming Data using Isolation Forest," in *2024 Seventh International Women in Data Science Conference at Prince Sultan University (WiDS PSU)*, IEEE, Mar. 2024, pp. 223–228. doi: 10.1109/WiDS-PSU61003.2024.00052.

[23] M. Agoramoorthy, A. Ali, D. Sujatha, M. Raj. T. F, and G. Ramesh, "An Analysis of Signature-Based Components in Hybrid Intrusion Detection Systems," in *2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS)*, IEEE, Dec. 2023, pp. 1–5. doi: 10.1109/ICCEBS58601.2023.10449209.

[24] U. Ahmed *et al.*, "Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering," *Sci. Rep.*, vol. 15, no. 1, p. 1726, Jan. 2025, doi: 10.1038/s41598-025-85866-7.

[25] L. Simon, A. Andreas, L. Leah, R. Ulrich, F. Ian, and S. Matthias, "Analyzing the Attack Surface and Threats of Industrial Internet of Things Devices," *Cryptography and Security*, vol. 14, no. 1, pp. 59–70, May 2024.

[26] C. Gan, J. Lin, D.-W. Huang, Q. Zhu, and L. Tian, "Advanced Persistent Threats and Their Defense Methods in Industrial Internet of Things: A Survey," *Mathematics*, vol. 11, no. 14, p. 3115, Jul. 2023, doi: 10.3390/math11143115.

[27] L. Zhang and L. Liu, "Data Anomaly Detection Based on Isolation Forest Algorithm," in *2022 International Conference on Computation, Big-Data and Engineering (ICCBE)*, IEEE, May 2022, pp. 87–89. doi: 10.1109/ICCBE56101.2022.9888169.

[28] O. AbuAlghanam, H. Alazzam, E. Alhenawi, M. Qatawneh, and O. Adwan, "Fusion-based anomaly detection system using modified isolation forest for internet of things," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 1, pp. 131–145, Jan. 2023, doi: 10.1007/s12652-022-04393-9.

[29] H. Xiang *et al.*, "Federated Learning-Based Anomaly Detection with Isolation Forest in the IoT-Edge Continuum," *ACM Transactions on Multimedia Computing, Communications, and Applications*, Nov. 2024, doi: 10.1145/3702995.

[30] H. Liu, J. Zhou, and H. Li, "Using Rough Sets to Improve the High-dimensional Data Anomaly Detection Method Based on Extended Isolation Forest," in *2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, IEEE, May 2023, pp. 231–236. doi: 10.1109/CSCWD57460.2023.10152795.

[31] Z. Azam, Md. M. Islam, and M. N. Huda, "Comparative Analysis of Intrusion Detection Systems and Machine Learning-Based Model Analysis Through Decision Tree," *IEEE Access*, vol. 11, pp. 80348–80391, 2023, doi: 10.1109/ACCESS.2023.3296444.

[32] N. Saran and N. Kesswani, "A comparative study of supervised Machine Learning classifiers for Intrusion Detection in Internet of Things," *Procedia Comput. Sci.*, vol. 218, pp. 2049–2057, 2023, doi: 10.1016/j.procs.2023.01.181.

[33] S. Qadir Mohammed and M. A. Hussein, "Performance Analysis of different Machine Learning Models for Intrusion Detection Systems," *Journal of Engineering*, vol. 28, no. 5, pp. 61–91, May 2022, doi: 10.31026/j.eng.2022.05.05.

[34] N. Alghanmi, R. Alotaibi, and S. M. Buhari, "Machine Learning Approaches for Anomaly Detection in IoT: An Overview and Future Research Directions," *Wirel. Pers. Commun.*, vol. 122, no. 3, pp. 2309–2324, Feb. 2022, doi: 10.1007/s11277-021-08994-z.

[35] V. M. Prasad and B. Bharathi, "A Survey on Security in Data Transmission Using Wireless Communication Methods for IoT Edge Devices," in *Smart Factories for Industry 5.0 Transformation*, Wiley, 2025, pp. 45–69. doi: 10.1002/9781394200467.ch3.

[36] E. Ortega, F. Su, R. Chattopadhyay, and K. Chakrabarty, "Discretized-Isolation Forest: Memory- and Compute-Efficient Unsupervised Anomaly Detection for Resource-Constrained Internet of Things Edge Devices," *IEEE Internet Things J.*, vol. 12, no. 2, pp. 1699–1717, Jan. 2025, doi: 10.1109/JIOT.2024.3468950.