

Recent Advances in Malware Detection Techniques for IoT Systems: A Systematic Review

Muhammad Sajjad Ur Rehman¹, Muhammad Junaid Arshad²

^{1,2}Department of Computer Science, University of Engineering and Technology, Lahore

¹msajjadm786@gmail.com

Abstract—The Internet of Things (IoT) has grown tremendously in the past decade, with billions of heterogeneous resource-constrained devices being deployed in fields such as smart homes, healthcare, industrial automation, and critical infrastructure. This massive growth has also resulted in a tremendous attack surface going largely unsupervised and unmonitored, making IoT ecosystems a desirable target for malware operators. Traditional security approaches, especially signature based Anti-Virus engines, are not designed for and suitable to the limited hardware and diverse architecture commonly found in IoT deployments and were never intended to detect or prevent new and unknown “zero day” threats. This paper aims to systematically review the IoT environment proposed malware detection techniques from 2015 to 2025. We share the literature search and screening process to obtain 38 primary studies, provide a taxonomy of detection paradigms, and perform a comparative analysis of signature-based, anomaly-based, classical machine learning, deep learning, and federated learning approaches. The review also analyzes publicly available benchmark datasets and presents typical performance results in the form of radar charts and bar charts, and discusses the main open challenges, such as limited resources, biases in the datasets, adversarial evasion, concept drift, and privacy, as well as providing some detailed lines for future research. The paper aims to be a comprehensive and evidence-based guidebook for researchers and practitioners working on malware detection for IoT deployments.

Index Terms—Internet of Things, IoT Security, Malware Detection, Intrusion Detection Systems, Machine Learning, Deep Learning, Federated Learning, Anomaly Detection, Botnet and Network Traffic Analysis

I. INTRODUCTION

THE IoT (Internet of Things) is one of the most game-changing technological advances of the 21st century.

IoT has revolutionized the way in which information is gathered, communicated, and acted upon by connecting physical gadgets such as household appliances, wearables, industrial controls, medical implants and other devices around the world to the internet. According to industry market analyses, the global installed base of IoT devices is expected to increase from less than 20 billion in the mid-2020s to over 40 billion in the early 2030s [1]. This monumental deployment is associated with a massive attack surface, and

the bad guys are getting more and more excited about targeting it. Most of the IoT devices have known and limited resources: low-power processors, limited memory, and reduced operating systems, realized on top of an embedded Linux or proprietary real-time kernel. Historically, manufacturers have focused on getting products to market quickly and have provided devices with default credentials, unpatched firmware and no way to update them on the device over-the-air with security fixes, as mentioned in [2]. These vulnerabilities make IoT devices particularly appealing to the creators of malware for building large botnets with little effort.

IoT malware infections can have serious consequences. In 2016, Mirai appeared on the scene, infecting over 600,000 devices by scanning the Internet for open Telnet ports and attempting to brute force default credentials [3]. Mirai was then used to start DDoS attacks against many internet services, such as Twitter, Netflix and Amazon. During the same time period BASHLITE aka Gafgyt showed us that even simple shell-based malware could be able to compromise large numbers of embedded systems based on Linux and weaponize them for DDoS attacks [4]. However, IoT malware is not confined to DDoS; payloads for ransomware, cryptomining, and data exfiltration have all been found on compromised IoT devices [40].

Traditional malware detection techniques are largely based on signature matching, which involves scanning a suspect file to match it to a database of known malicious signatures. Although this method is very low cost and accurate when it comes to known threats, it is essentially reactive – a malware is first seen, then analyzed and catalogued and then detected. This limitation is further complicated by hardware differences (ARM, MIPS, x86, PowerPC) and the sheer impracticality of installing heavy-weight AV agents in constrained hardware in IoT environments [7]. With the awareness of these limitations, the research community has investigated a wide range of alternative approaches. Anomaly-based and behavior-based detection techniques analyze network traffic, system calls, memory access patterns and report variations from a baseline, thus identifying new threats that have not been previously detected [9]. A variety of benchmarks have been presented with detection accuracy greater than 99% using machine learning classifiers with labeled traffic data [8].

The federated learning has been proposed to train shared detection models over multiple IoT deployments without the need of centralizing sensitive network data [24].

Yet, none of the existing reviews has so far presented a comprehensive overview of the different techniques ranging from classical statistical approaches to privacy-preserving distributed learning under a single roof, while also providing a methodological transparency, and explicitly taking into account the practical limitations of IoT hardware. The present paper is intended to fill this gap. In particular, our contribution is the following:

- We describe an explicit, reproducible literature search and screening methodology (Section III) used to identify the primary studies surveyed.
- We provide a structured taxonomy of malware detection techniques applicable to IoT systems, organized by detection paradigm and underlying algorithmic family (Section IV).
- We offer a systematic literature review of signature-based, anomaly-based, classical machine learning, deep learning, and federated learning approaches proposed between 2015 and 2025, including their datasets, features, and reported performance metrics (Section V).
- We compile and compare the most widely used IoT malware datasets (Section VI).
- We present a multi-dimensional comparative analysis using radar (spider) charts and grouped bar charts to visualize trade-offs across detection accuracy, resource efficiency, scalability, privacy, interpretability, and robustness (Section VII).
- We identify the principal open challenges (Section VIII) and outline concrete directions for future work (Section IX).

II. BACKGROUND

A. IoT Architecture and Security Layers

An IoT ecosystem is typically modeled as a layered stack consisting of a perception layer (sensors, actuators, embedded controllers that physically interact with the environment), a network layer (data transport via Wi-Fi, Zigbee, Z-Wave, Bluetooth Low Energy, LoRaWAN, or NB-IoT), and an application layer (domain-specific services such as smart metering, patient monitoring, or vehicle tracking) [39].

B. IoT Malware Families and Attack Vectors

Table I lists the most important IoT malware families found in the literature. However, Mirai and its variants continue to be prominent and dominate the majority of botnet infections that have been observed in various reports [3]. Like Mirai, BASHLITE has infected millions of devices and is unique because of its shell command injection [4]. Hajime uses a peer-to-peer command-and-control (C&C) architecture, unlike centralized servers, which would be easily targeted for takedown. In contrast to Mirai and BASHLITE, which are largely based on Telnet credential brute-forcing, the Reaper attack vector studied in depth in the deployment of honeypots,

including IoTTPOT [6] was based on direct exploitation of software vulnerabilities, which is a qualitative change in attacker sophistication [35].

TABLE I
NOTABLE IoT MALWARE FAMILIES AND THEIR CHARACTERISTICS

Family	Year	Primary Attack Vector / Capability
Mirai	2016	Telnet brute-force; large-scale DDoS
BASHLITE	2014	Shell command injection; DDoS
Hajime	2016	P2P C&C; credential brute-force
Reaper/IoTroop	2017	CVE exploitation; centralized C&C
Satori	2017	Mirai variant; router exploitation
HideNSeek	2018	First IoT malware to survive reboot
Mozi	2019	P2P DHT botnet; DDoS, data theft
BotenaGo	2021	Go-based loader; 30+ device exploits
Dark Nexus	2020	Code obfuscation; sandbox evasion

C. Why Traditional Defenses Fall Short

Figure 1 places these malware families on a timeline, with successive generations getting increasingly sophisticated propagation and evasion techniques, ranging from simple credential brute-forcing (BASHLITE and Mirai) to CVE exploitation (Reaper), reboot-persistent infection (HideNSeek), peer-to-peer command-and-control (Hajime and Mozi), and active anti-analysis (Dark Nexus).

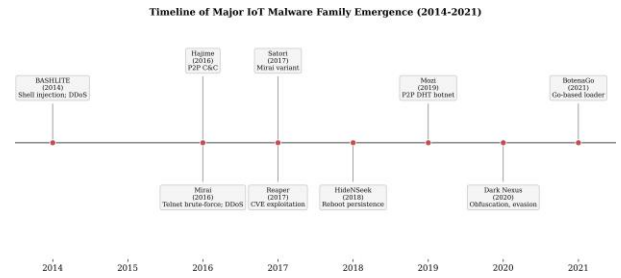


Fig. 1. Timeline of major IoT malware family emergence (2014–2021), corresponding to the families listed in Table I.

There are three key characteristics of IoT devices that render the traditional approach to malware defense impractical. Firstly, *resource constraints*: microcontrollers and system-on-chip (SoC) devices might have just a few megabytes of RAM and a single-core processor that operates below 1 GHz, and would not have any extra resources for a heavyweight security agent. Second, *heterogeneity*: deployments of IoT devices use a variety of processor instruction set architectures, and a binary-level signature for one of these will not apply to the same malware compiled for another. Third, *operational context*: many IoT devices run unattended for years, and cannot be easily rebooted for updates, thus rendering it extremely difficult to push database updates of signatures in timely fashion [2].

III. REVIEW METHODOLOGY

To enable systematic and reproducible literature search and screening, a structured protocol was applied which was motivated by PRISMA style systematic review guidelines, as shown in Fig. 2.

A. Search Strategy

IEEE Xplore, ScienceDirect, SpringerLink, MDPI, ACM Digital Library, USENIX and arXiv were searched with keywords combinations: “IoT malware detection,” “IoT botnet,” “intrusion detection system IoT,” “machine learning IoT security,” “deep learning malware,” and “federated learning IoT.” Only papers published between January 2015 and June 2025 were searched, except for a few seminal papers published before 2015 that are fundamental to the field (e.g., malware visualization techniques and IoT architecture surveys).

B. Inclusion and Exclusion Criteria

Studies were included if they (i) proposed or evaluated a malware or intrusion detection technique with explicit applicability to IoT or embedded network environments, (ii) reported quantitative evaluation results using standard metrics (accuracy, precision, recall, F1-score, or false positive rate), and (iii) were published in a peer-reviewed venue or a widely cited preprint with subsequent peer-reviewed publication. Studies were excluded if they focused exclusively on non-IoT platforms (e.g., desktop or mobile malware with no IoT relevance), lacked sufficient methodological detail to assess reproducibility, or were duplicate reports of the same underlying study.

C. Screening Process

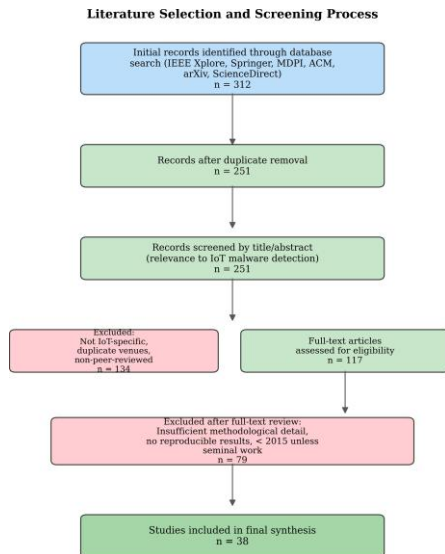


Fig. 2. Literature selection and screening process used in this review.

As shown in Fig. 2, an initial pool of 312 records was identified through database searches. After removing duplicates, 251 unique records remained. Title and abstract screening removed 134 records that were not IoT-specific, were duplicate conference/journal versions, or were not peer-reviewed, leaving 117 records for full-text assessment. A further 79 records were excluded after full-text review for insufficient methodological detail, lack of reproducible results, or being superseded by a more recent version of the same work. The final synthesis

includes 38 primary studies, which form the basis of the literature review in Section V.

IV. TAXONOMY OF DETECTION TECHNIQUES

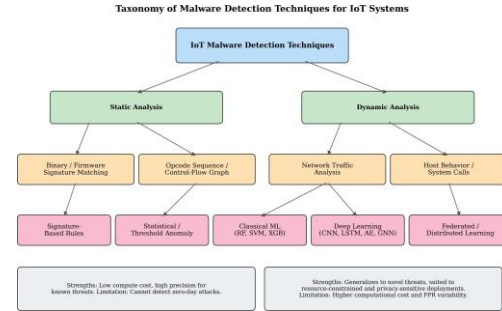


Fig. 3. Taxonomy of malware detection techniques for IoT systems.

Fig. 3 presents the taxonomy of malware detection techniques discussed in this review. At the top level we distinguish between *static* analysis, which examines a binary or firmware image without execution, and *dynamic* analysis, which monitors behavior during execution or over the network. Within dynamic analysis we distinguish *network-based* approaches – which analyze traffic flows without touching the device itself – from *host-based* approaches that require an agent running on the device. Cutting across both dimensions, detection logic can be signature-based, anomaly-based, or hybrid, and the underlying algorithmic engine may be a rule-based system, a classical machine learning classifier, a deep neural network, or a federated/distributed variant of any of the above.

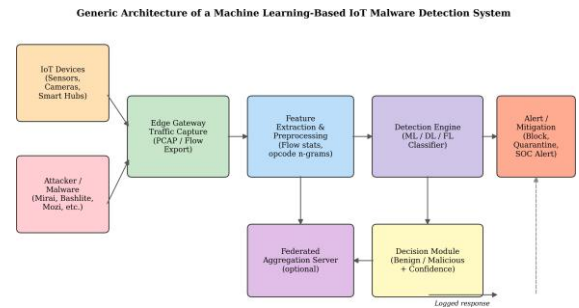


Fig. 4. Generic architecture of a machine learning-based IoT malware detection system, showing the data path from IoT devices through feature extraction to the detection engine and downstream response actions.

The end-to-end architecture of a typical machine learning-based IoT intrusion/malware detection deployment is shown in Fig. 4. Traffic generated by IoT devices (and potentially compromised devices running malware) is captured at an edge gateway, converted into flow-level or packet-level features, and passed to a detection engine. The detection engine – which may be a classical ML classifier, a deep neural network, or a federated model coordinated by an aggregation server – produces a benign/malicious decision with an associated confidence score, which is then used to trigger an alert, quarantine action, or logged response.

V. LITERATURE REVIEW

This section presents the systematic review of the 38 primary studies, organized according to the taxonomy introduced in Section IV.

A. Signature-Based Detection

Signature-based detection was the dominant paradigm in endpoint security throughout the 1990s and 2000s. A unique binary pattern or hash extracted from a known malicious sample is stored in a database; incoming files or network packets are scanned and matched against this database [7]. The technique is computationally cheap and highly precise for known threats, which explains why it remains in use as one layer within multi-layer security products.

In the IoT world, some modifications have been suggested. Abbas and Srikanthan proposed a low-complexity signature matching engine for embedded processors, which includes a pre-screening procedure of the signatures based on the Bloom filter to decrease the computational burden of signature matching [34]. Tam et al. examined firmware images and discovered that a significant percentage of devices were delivered with certain known vulnerable software components that can be detected by CVE-based signatures [33].

B. Anomaly-Based and Behavior-Based Detection

With anomaly detection, the challenge is not to find known malicious activity, but rather to understand the “normal” behavior of the device and detect activity that falls outside of it. In the case of IoT devices, they are designed to only perform a small and known set of operations: a smart thermostat, for instance, should never make connections to random IP addresses on the Internet or start shell processes [9].

1) *Statistical and Threshold-Based Approaches*: The first research in the area of anomaly detection was to establish statistical profiles of network traffic volume, protocol distribution and inter-packet timing. Meidan *et al.* created the N-BaIoT dataset and showed that using a shallow autoencoder with traffic features gathered from per-device benign traffic, the detection of Mirai and BASHLITE infections could be achieved with almost perfect accuracy in terms of deviations in statistical features of the traffic [8]. Doshi et al. demonstrated that deep packet inspection can be avoided to detect DDoS traffic from compromised IoT devices using a simple random forest, which is trained on flow level features like packet rate, byte rate, and protocol flags [10].

2) *Network Traffic Profiling*: Much research has been directed towards passively characterizing IoT device activities at the network level. This is desirable because no changes are necessary to the device itself, the detection logic is performed on the gateway, router, or network tap. Sivanathan *et al.* created a multi-stage machine learning classification approach which can classify IoT device types from their network traffic characteristics with more than 99% accuracy, paving the way for future device-level visibility for downstream anomaly detectors to detect unauthorized and/or compromised devices [13]. The Bot-IoT dataset was created by Koroniotis *et al.* and

they tested various network-level anomaly detectors, including flow-based features, which were found to be the most stable and generalizable [12]. During the scanning and propagation phase, before the attack payload is delivered, Kumar and Lim proposed the distributed modular system called EDIMA (Early Detection of IoT Malware Network Activity) [41].

3) *System Call Monitoring*: On devices running a Linux kernel, system call sequences provide a rich behavioral signal. Shobana and Poonkuzhali collected sequences of system calls from IoT devices during benign and malicious operation and used deep learning to classify them, achieving detection rates above 97% on their experimental testbed [14]. The primary limitation of this approach is that it requires the ability to instrument or modify the target device, which is often infeasible in practice.

C. Machine Learning-Based Detection

The application of classical machine learning classifiers to IoT malware detection has been the dominant research strand of the period 2018–2022. The general workflow consists of three stages: feature extraction from network traffic or binary data, feature selection or dimensionality reduction, and classification using one of several standard algorithms.

1) *Feature Engineering*: Features can be extracted at multiple levels of abstraction. At the packet level, individual fields from IP, TCP, UDP, and application-layer headers provide fine-grained information. At the flow level, aggregated statistics over all packets belonging to a five-tuple (source IP, destination IP, source port, destination port, protocol) capture session-level behavior while discarding the content of individual packets. Typical flow features include packet count, byte count, mean and standard deviation of inter-arrival times, flag distributions, and flow duration [12]. At the firmware level, byte n-gram frequency histograms and opcode sequences extracted from disassembled binaries have been used as features for malware classification across architectures [15].

2) *Classical Classifiers*: Table II summarizes the performance of representative ML-based IoT malware detection systems reported in the literature. Random forests have been the most consistently used classifier, largely because they are robust to feature noise and require minimal hyperparameter tuning. Another effective machine for relatively small sets of features is the support vector machine (SVM) with radial basis function (RBF) kernels. Decision trees also are sometimes used because they are easier to explain, particularly when the deployment is such that an operator would want to know the reason for a given device being flagged. Yousaf *et al.* used a hybrid approach of random forest and SVM classifiers to classify Mirai and BASHLITE traffic in the N-BaIoT dataset, achieving accuracy of more than 99% [37].

3) *Ensemble Methods*: In the field of ensemble learning, combination of multiple base classifiers has always been the most successful approach. Because of their relatively fast inference times and high accuracy, gradient boosted trees (XGBoost and LightGBM, in particular) have become popular for deployment on edge gateways, as opposed to cloud backends

TABLE II
PERFORMANCE COMPARISON OF REPRESENTATIVE MACHINE LEARNING- AND DEEP LEARNING-BASED IoT MALWARE DETECTION SYSTEMS

Reference	Algorithm	Dataset	Accuracy	Precision	Recall	F1-Score	Year
Meidan et al. [8]	Deep Autoencoder	N-BaIoT	99.3%	99.1%	99.5%	99.3%	2018
Doshi et al. [10]	Random Forest	Custom (DDoS)	99.0%	98.7%	99.2%	98.9%	2018
Koroniotis et al. [12]	DT / NB / RF	Bot-IoT	98.2%	97.9%	98.6%	98.2%	2019
Lo et al. [21]	GNN (E-GraphSAGE)	NF-Bot-IoT	—	—	—	97.0%	2022
Yousaf et al. [37]	RF + SVM	N-BaIoT	99.1%	98.9%	99.3%	99.1%	2024
Riaz et al. [20]	LSTM	Custom IoT	97.8%	97.2%	98.1%	97.6%	2022
Lee et al. [27]	RF (opcode features)	Custom IoT binaries	98.1%	97.8%	98.4%	98.1%	2023
Rey et al. [24]	Federated MLP	N-BaIoT	97.5%	97.1%	97.9%	97.5%	2022
Nobakht et al. [26]	FL + Encryption	N-BaIoT	97.2%	96.8%	97.6%	97.2%	2024

Note: “—” indicates that the corresponding metric was not separately reported in the original publication for the configuration shown.

[35]. Bagging ensembles decrease variance and are suitable for the class imbalance that is usual in IoT traffic datasets, where the ratio of benign traffic to malicious samples is large.

D. Deep Learning-Based Detection

Since 2020, deep learning techniques have largely replaced traditional machine learning techniques in the IoT malware detection field of research due to the availability of large labeled IoT datasets and the fact that neural networks can learn useful feature representations without human intervention.

1) *Convolutional Neural Networks*: CNNs were first used for malware detection using the approach of representing each binary executable as a gray scale image, in which each byte of the binary is mapped to a pixel value, and using a standard image classifier to classify the image [16]. The visualization technique can be easily extended to multi-class classification of IoT malware families (and naturally IoT binaries) [17]. For network traffic, one-dimensional CNNs applied directly to raw packet bytes or flow feature vectors have achieved competitive results without requiring manual feature engineering, and several studies evaluated on the CIC IoT 2023 dataset report accuracies above 98% across multiple attack categories [32].

2) *Recurrent and Sequence-Based Models*: Recurrent neural networks, and in particular long short-term memory (LSTM) networks, are naturally suited to temporal sequences of network packets or system calls. Riaz *et al.* used an LSTM-based architecture to capture temporal patterns in IoT device network behavior, achieving detection rates above 97% while maintaining a false positive rate below 1% [20]. Bidirectional LSTMs and gated recurrent units (GRUs) have been explored as lower-complexity alternatives that reduce training time while retaining the temporal modeling capacity of standard LSTMs.

3) *Autoencoders and Generative Models*: Unsupervised autoencoders are particularly valuable for IoT security because they require only benign training data – a much easier data collection problem than assembling a labeled dataset of malicious samples. The autoencoder learns a compressed representation of normal behavior; at inference time, inputs that reconstruct poorly (i.e., whose reconstruction error exceeds a threshold) are flagged as anomalous. Generative adversarial networks

(GANs) have also been explored as a way to synthesize additional malicious samples to address the severe class imbalance present in most IoT traffic datasets.

4) *Graph Neural Networks*: An emerging thread of research represents IoT device communication patterns as graphs – devices are nodes, and observed communication flows are directed edges – and applies graph neural networks (GNNs) to detect anomalous subgraphs consistent with botnet command-and-control traffic. Lo *et al.* proposed E-GraphSAGE, which extends the GraphSAGE architecture to incorporate edge (flow) features for binary attack/benign classification, reporting an F1-score of approximately 0.97 on the NetFlow-converted Bot-IoT dataset, comparable to leading non-graph baselines while additionally capturing topological relationships between devices [21].

5) *Transfer Learning and Cross-Architecture Detection*: A persistent challenge in IoT malware analysis is that the same malware family is compiled for multiple CPU architectures, and a model trained on ARM binaries may fail to detect the same malware compiled for MIPS. Vasan *et al.* proposed a cross-architecture detection framework using a neural network ensemble with advanced transfer learning, demonstrating that image-based representations of malware binaries preserve enough structural information to support cross-architecture classification [18]. More recent work has explored large pre-trained models fine-tuned on IoT-specific malware corpora as a foundation for cross-platform detection [22].

E. Federated Learning for IoT Malware Detection

Federated learning (FL) [23] was proposed as a method for training machine learning models on data distributed across many edge devices without requiring that data to be transmitted to a central server. Each participant trains a local model on its own data and uploads only the model weights (or gradients) to an aggregation server, which combines them typically using a weighted average to produce an updated global model that is then pushed back to the participants.

1) *FedAvg and Its Variants*: Rey *et al.* [24] were among the first to systematically explore FL for the detection of IoT malware with an N-BaIoT dataset split among various simulated participants. They showed that a federated model

can have similar detection accuracy to a centrally trained model, without revealing any raw traffic data. Subsequent work has addressed the challenge of non-independent and identically distributed (non-IID) data – a common situation where different IoT deployments have very different traffic profiles – using techniques such as FedProx, FedNova, and personalized FL variants [28].

Fig. 5 illustrates the qualitative convergence behavior reported across these studies: centrally trained models typically reach their asymptotic accuracy fastest, federated models trained on IID-partitioned data converge somewhat more slowly but reach within one to two percentage points of the centralized upper bound, while federated models trained on non-IID partitions converge more slowly still and tend to plateau at a noticeably lower accuracy, motivating the personalized and adaptive aggregation strategies summarized in Table III.

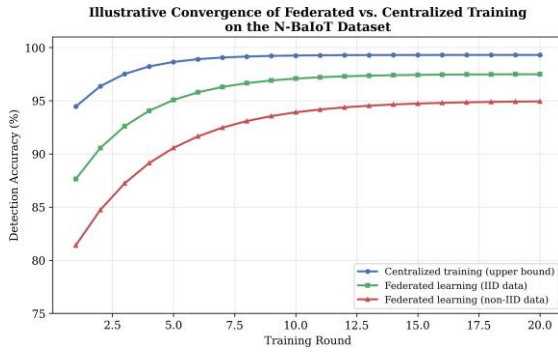


Fig. 5. Illustrative convergence behavior of centralized versus federated training under IID and non-IID data partitions, based on trends reported across the federated learning studies reviewed in Section V-E.

TABLE III
FEDERATED AGGREGATION STRATEGIES DISCUSSED IN THE REVIEWED LITERATURE

Strategy	Primary Goal	Reported Trade-off
FedAvg [23]	Baseline weighted-average aggregation	Simple but degrades under non-IID data
FedProx [28]	Tolerate client drift via proximal term	Improved stability; added hyperparameter tuning
FedNova [28]	Normalize local update magnitudes	Better convergence; higher communication cost
Personalized FL [28]	Per-client model adaptation	Higher per-client accuracy; reduced global generalization
Secure aggregation [25], [26]	Protect update confidentiality	Strong privacy; added computational overhead

2) *Privacy and Security of Federated Models*: The aggregation step in FL is itself a potential attack surface. Gradient inversion attacks can, in theory, reconstruct training data from model gradients, while poisoning attacks can corrupt the global model by submitting adversarially crafted weight updates from a malicious participant. Mothukuri *et al.* analyzed these attack vectors in the context of federated IoT anomaly detection and proposed differential privacy mechanisms and secure aggregation protocols as defenses [25]. Nobakht *et al.* introduced

a secure IoT malware detection framework that integrates FL with homomorphic encryption to prevent the aggregation server from observing individual participants' updates [26]. More recently, Namakshenas *et al.* proposed a federated threat-detection model for consumer IoT that incorporates quantum-resistant cryptographic primitives into the aggregation protocol, explicitly addressing the threat that future quantum-capable adversaries could compromise the confidentiality of model updates exchanged during federated training [19].

VI. BENCHMARK DATASETS

The availability of high-quality, labeled datasets has been a decisive factor in the progress of IoT malware detection research. Table IV summarizes the most widely used datasets identified during the review.

N-BaIoT [8] remains one of the most cited datasets in this field. It was constructed by deploying nine real consumer IoT devices in a controlled testbed, infecting them with Mirai and BASHLITE variants, and capturing per-device flow statistics. Bot-IoT [12] is larger and more diverse, encompassing not only botnet traffic but also network reconnaissance and data exfiltration scenarios. IoT-23 [29] is notable for its breadth of malware families and for its realistic mix of malicious and benign scenarios.

Fig. 6 compares the six datasets by the number of distinct traffic classes (benign plus attack categories) they provide. IoT-23 offers by far the largest number of distinct classes, reflecting its 20 separate malware capture scenarios, while MedBioT and Bot-IoT offer a smaller number of broader attack categories more suitable for binary or coarse-grained multi-class classification tasks.

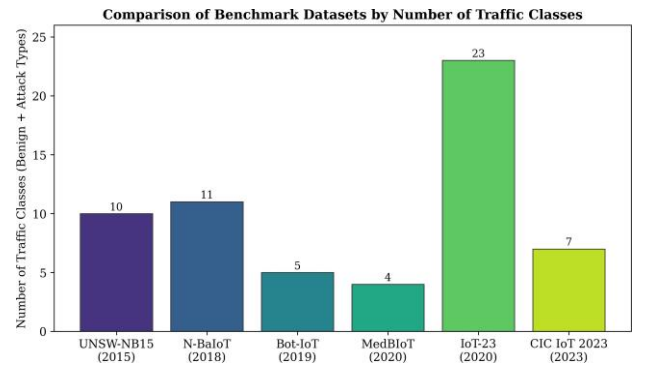


Fig. 6. Comparison of benchmark datasets by number of distinct traffic classes (benign plus attack categories).

Another criticism of all the datasets is that they are a snapshot of the malware's behavior at a single point in time; as IoT malware families constantly evolve, classifiers trained on these datasets may struggle to adapt to new variants in the future [38]. Dataset imbalance – where benign traffic vastly outnumbers malicious traffic – is a further challenge that researchers have addressed using oversampling (SMOTE), undersampling, and cost-sensitive learning.

TABLE IV
BENCHMARK DATASETS USED IN IoT MALWARE DETECTION RESEARCH

Dataset	Year	Traffic Type	Classes	Key Characteristics
N-BaIoT [8]	2018	Network (flow statistics)	Benign + 10 attack	Per-device statistical features collected from 9 real consumer IoT devices infected with Mirai/BASHLITE
Bot-IoT [12]	2019	Network (PCAP / flows)	5	Large-scale; includes DDoS, DoS, data theft, and reconnaissance scenarios
MedBioT [30]	2020	Network (PCAP)	Benign + 3 botnet	Medium-sized real IoT testbed; Mirai, Torii, Bashlite
UNSW-NB15 [31]	2015	Network (mixed)	10	Widely used general IDS benchmark with partial IoT-relevant traffic
IoT-23 [29]	2020	Network (PCAP / flows)	23	20 malicious + 3 benign scenarios; widely used for botnet labeling research
CIC IoT 2023 [32]	2023	Network (flows)	7	33 attacks across 105 IoT devices, grouped into 7 categories (DDoS, DoS, recon, web-based, brute force, spoofing, Mirai); 47 flow-level features

VII. COMPARATIVE ANALYSIS

This section synthesizes the findings of Section V through a multi-dimensional comparative analysis, visualized using radar (spider) charts and bar charts.

A. Multi-Criteria Comparison of Detection Paradigms

Fig. 7 presents a radar chart comparing the five major detection paradigms – signature-based, anomaly-based, classical ML, deep learning, and federated learning – across seven evaluation criteria: detection accuracy, zero-day capability, resource efficiency, scalability, privacy preservation, interpretability, and adversarial robustness. Each criterion is scored on a 1–10 scale based on qualitative assessment of the reviewed literature.

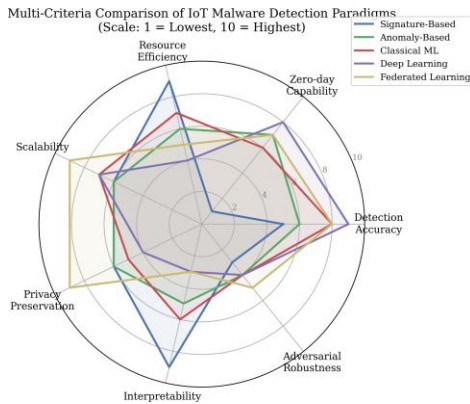


Fig. 7. Multi-criteria radar (spider) chart comparing the five major IoT malware detection paradigms across seven evaluation dimensions.

The radar chart highlights several important trade-offs. Signature-based detection is the most resource-efficient and most interpretable approach but is least capable of defending against zero-day attacks or adversarial attacks, confirming its role as a complementary rather than standalone defense. Deep learning achieves the highest detection accuracy and strong zero-day capability but scores poorly on resource efficiency and interpretability, reflecting the computational cost and “black-box” nature of neural networks. Federated learning achieves the best balance of scalability and privacy preservation, while classical ML occupies a balanced middle

ground that explains its continued popularity in resource-constrained deployments.

B. Performance Profile of Representative Models

Fig. 8 presents a second radar chart comparing four representative models – a random forest evaluated on N-BaIoT, a random forest using opcode-category features evaluated on a custom IoT malware binary corpus, an LSTM evaluated on a custom IoT dataset, and a federated MLP evaluated on N-BaIoT – across six standard evaluation metrics: accuracy, precision, recall, F1-score, AUC, and true positive rate (TPR).

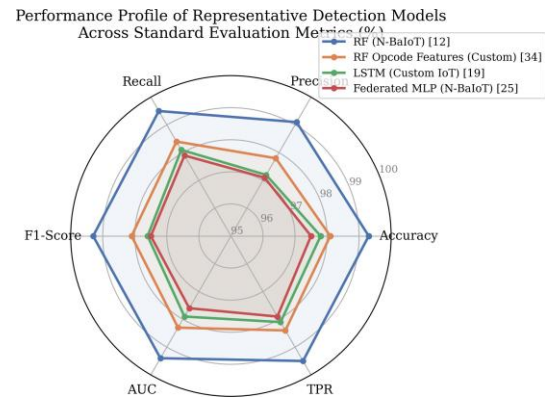


Fig. 8. Radar chart comparing the performance profile of four representative detection models across six evaluation metrics (values in percent).

All four models achieve metrics above 97% across the board, illustrating that on curated benchmark datasets, the absolute performance differences between algorithmic families are relatively small. The random forest model evaluated on N-BaIoT achieves the most balanced and consistently high profile, while the federated MLP – despite operating under the additional constraint of not centralizing training data – remains within approximately two percentage points of the centrally trained models across all metrics.

C. Accuracy and False Positive Rate Trade-offs

Fig. 9 presents a grouped bar chart comparing the average detection accuracy and average false positive rate (FPR) reported across the studies reviewed for each algorithmic family.

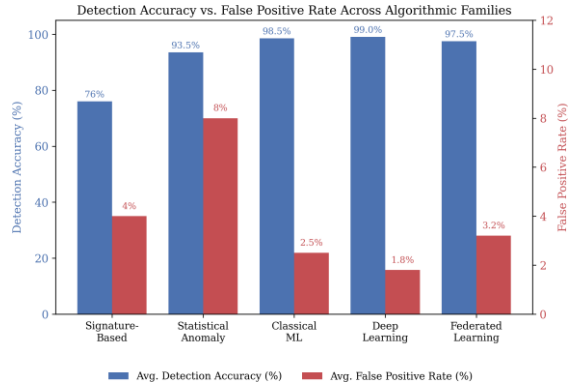


Fig. 9. Average detection accuracy and false positive rate across algorithmic families, aggregated from the studies reviewed.

Deep learning approaches achieve both the highest average accuracy (99.0%) and the lowest average FPR (1.8%), but this advantage must be weighed against the resource efficiency and interpretability disadvantages identified in Fig. 7. Signature-based detection exhibits the lowest accuracy (76%) and highest FPR (4%), primarily because the figure aggregates across both known and unknown threats.

D. Research Trends Over Time

Fig. 10 presents a stacked bar chart illustrating the relative research emphasis on each algorithmic family over five two-year periods from 2016 to 2025, based on the publication years of the studies identified during the literature search.

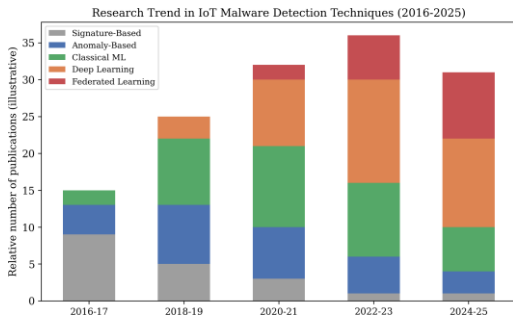


Fig. 10. Illustrative research trend showing the relative emphasis on each detection paradigm over time, based on the distribution of publication years in the reviewed literature.

The trend confirms the narrative developed throughout Section V: signature-based approaches dominated early IoT security research (2016–2017), anomaly-based and classical ML approaches grew steadily through 2018–2021, deep learning became the dominant paradigm from 2020 onward, and federated learning has shown the most rapid relative growth in the most recent period (2022–2025), reflecting the field’s increasing emphasis on privacy and distributed deployment.

VIII. OPEN CHALLENGES

TABLE V
SUMMARY OF OPEN CHALLENGES IN IoT MALWARE DETECTION

Challenge	Severity	Research Maturity	Primary Mitigation Direction
Resource constraints	High	Emerging	TinyML, quantization, pruning
Dataset quality	High	Moderate	Continuously updated, multi-vendor datasets
Adversarial evasion	Medium	Early	Adversarial training, certified defenses
Zero-day threats	High	Moderate	Anomaly detection, hybrid models
Concept drift	Medium	Early	Continual / on-line learning
Privacy in collaboration	Medium	Moderate	Federated learning, secure aggregation

Table V summarizes the six open challenges discussed in this section, together with a qualitative assessment of their current severity and research maturity.

A. Resource-Constrained Deployment

Running a detection engine directly on an IoT device is highly desirable because it eliminates the network latency and privacy exposure of cloud-based analysis. However, even the most carefully optimized neural network inference remains beyond the memory and compute budget of many constrained devices. TinyML, the field concerned with deploying machine learning on microcontrollers is an active area of research that holds promise for on-device IoT malware detection.

B. Dataset Quality and Diversity

As discussed in Section VI, existing benchmark datasets have several limitations: they are often collected in controlled laboratory environments that do not fully reflect the traffic variability of real-world deployments, they may not include recent malware families, and they are typically collected from a small number of device models. Constructing a truly representative, continuously updated, and openly available IoT malware dataset remains an important unsolved problem [38].

C. Adversarial Evasion

Adversarial machine learning techniques can be used to craft network traffic or binary payloads that are correctly classified as benign by a learned detection model while still executing malicious functionality. Attacks such as the fast gradient sign method (FGSM) and more sophisticated iterative methods have been applied to IDS and malware detection models, substantially reducing detection rates [36]. Making IoT malware detection models robust to adversarial perturbations is an active and largely open research problem.

D. Zero-Day Threats

A zero-day exploit by definition, or a truly new family of malware, will not be present in any training set. A key open question is the development of detection systems that can both detect new threats and ensure a low false positive rate in the diverse and noisy environment of real IoT deployments.

E. Concept Drift

IoT device behavior changes with the passage of time as new firmware versions, user behaviors and network topology evolve. When the distribution of normal traffic deviates from the distribution used to train a detection model, this phenomenon is known as concept drift. Continuous learning approaches that enable the model to update incrementally when new data arrive represent an important but under-explored direction [24].

F. Privacy in Collaborative Detection

Many of the most effective detection approaches benefit from large and diverse datasets, which naturally suggests collaborative data sharing among organizations. However, IoT network traffic can be sensitive – it may reveal occupancy patterns in smart homes or production schedules in industrial settings. Federated learning addresses some of these concerns, but as noted in Section V-E, the FL protocol itself introduces new attack vectors. Achieving robust privacy guarantees without unacceptable accuracy costs remains a key open challenge [25].

IX. FUTURE RESEARCH DIRECTIONS

Based on the gaps and limitations identified in this review, we identify the following as particularly promising directions for future research.

Lightweight and on-device detection: Designing neural network architectures optimized for the memory and compute constraints of IoT microcontrollers using TinyML techniques, binary neural networks, or hardware-aware neural architecture search would enable detection to be pushed to the network edge, reducing latency and exposure.

Cross-architecture generalization: Detection models that generalize across CPU architectures without retraining are highly valuable given the extreme diversity of IoT hardware. Transfer learning with architecture-agnostic intermediate representations is one promising avenue.

Continual and online learning: Detection systems capable of incrementally updating their models in response to new traffic observations without catastrophic forgetting of previously learned patterns would be better suited to the dynamic nature of real IoT environments.

Adversarially robust models: Incorporating adversarial training, certified defenses, or ensemble diversity strategies into IoT malware detection models would improve resilience to evasion attacks, which are likely to become more prevalent as ML-based defenses are deployed more widely.

Explainable detection. Developing explainability methods appropriate for IoT detection architectures such as attention

visualization for sequence models or SHAP values for gradient boosted classifiers will improve operational trust and facilitate faster incident response.

Standardized evaluation benchmarks. The field would benefit greatly from a standardized evaluation protocol including a common set of datasets, evaluation metrics, and experimental conditions that would allow fair and reproducible comparison across methods.

X. CONCLUSION

This paper has presented a systematic review of malware detection techniques for IoT systems, spanning the period from the emergence of the Mirai botnet in 2016 through to the most recent advances in federated and privacy-preserving detection as of 2025. We described an explicit literature search and screening methodology that identified 38 primary studies, introduced a taxonomy of detection approaches, and systematically reviewed signature-based, anomaly-based, machine learning, deep learning, and federated learning methods. We compiled and characterized the major benchmark datasets used in the field and conducted a multi-dimensional comparative analysis – using radar charts and bar charts – of reported detection performance across approaches.

This review has three main findings. First, there is no single detection paradigm that is always best: signature-based approaches can be accurate for known threats but ineffective against new variants; anomaly detection may generalize well but has a high false positive rate; supervised ML and DL can be very accurate on benchmark data but require large labeled training datasets and may not work equally well across all device types or network environments. Second, deploying detection logic directly on resource-constrained IoT hardware – rather than on a cloud backend or gateway – remains a major engineering challenge requiring continued attention from both the machine learning and embedded systems communities.

Finally, federated learning represents a genuinely promising direction for scalable, privacy-preserving collaborative detection, but substantial research is still needed to address its vulnerability to poisoning and inference attacks.

We hope this review is helpful to researchers and practitioners developing security architectures for IoT deployments, all while being methodologically transparent.

REFERENCES

- [1] Transforma Insights / Statista, “Number of Internet of Things (IoT) connections worldwide from 2022 to 2023, with forecasts from 2024 to 2034,” Statista, 2025. [Online]. Available: <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>
- [2] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, “Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on internet-scale IoT exploitations,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2702–2733, 2019.

- [3] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, "Understanding the Mirai botnet," in *Proc. 26th USENIX Security Symposium*, Vancouver, Canada, Aug. 2017, pp. 1093–1110.
- [4] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, Jul. 2017.
- [5] S. Edwards and I. Profetis, "Hajime: Analysis of a decentralized internet worm for IoT devices," *Rapidity Networks*, Tech. Rep., Oct. 2016.
- [6] Y. M. P. Pa, S. Suzuki, K. Yoshioka, T. Matsumoto, T. Kasama, and C. Rossow, "IoTPOT: A novel honeypot for revealing current IoT threats," *Journal of Information Processing*, vol. 24, no. 3, pp. 522–533, 2016.
- [7] O. Aslan and R. Samet, "A comprehensive review on malware detection approaches," *IEEE Access*, vol. 8, pp. 6249–6271, 2020.
- [8] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, Jul.–Sep. 2018.
- [9] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," in *Proc. Network and Distributed Systems Security (NDSS) Symposium*, San Diego, CA, Feb. 2018.
- [10] R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning DDoS detection for consumer internet of things devices," in *Proc. IEEE Security and Privacy Workshops (SPW)*, San Francisco, CA, May 2018, pp. 29–35.
- [11] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and A.-R. Sadeghi, "D²IoT: A federated self-learning anomaly detection system for IoT," in *Proc. IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, Dallas, TX, Jul. 2019, pp. 756–767.
- [12] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Generation Computer Systems*, vol. 100, pp. 779–796, Nov. 2019.
- [13] A. Sivanathan, H. H. Gharakheili, F. Loi, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman, "Classifying IoT devices in smart environments using network traffic characteristics," *IEEE Transactions on Mobile Computing*, vol. 18, no. 8, pp. 1745–1759, Aug. 2019.
- [14] M. Shobana and S. Poonkuzhali, "A novel approach to detect IoT malware by system calls using deep learning techniques," in *Proc. International Conference on Innovative Trends in Information Technology (ICITIIT)*, Kottayam, India, Feb. 2020, pp. 1–5.
- [15] E. Cozzi, M. Graziano, Y. Fratantonio, and D. Balzarotti, "Understanding Linux malware," in *Proc. IEEE Symposium on Security and Privacy (S&P)*, San Francisco, CA, May 2018, pp. 161–175.
- [16] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: Visualization and automatic classification," in *Proc. 8th International Symposium on Visualization for Cyber Security (VizSec)*, Pittsburgh, PA, Jul. 2011, pp. 1–7.
- [17] O. Sharma, A. Sharma, and A. Kalia, "Windows and IoT malware visualization and classification with deep CNN and Xception CNN using Markov images," *Journal of Intelligent Information Systems*, vol. 60, no. 2, pp. 349–375, Apr. 2023.
- [18] D. Vasan, M. Alazab, S. Venkatraman, J. Akram, and Z. Qin, "MTHAEL: Cross-architecture IoT malware detection based on neural network advanced ensemble learning," *IEEE Transactions on Computers*, vol. 69, no. 11, pp. 1654–1667, Nov. 2020.
- [19] D. Namakshenas, A. Yazdinejad, A. Dehghantanha, and G. Srivastava, "Federated quantum-based privacy-preserving threat detection model for consumer Internet of Things," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 3, pp. 5829–5838, 2024.
- [20] S. Riaz, S. Latif, S. U. R. Malik, C. Shorten, O. Rana, and R. Raza, "Malware detection in IoT systems using machine learning techniques," *Sensors*, vol. 22, no. 23, p. 9413, Dec. 2022.
- [21] W. W. Lo, S. Layeghy, M. Sarhan, M. Gallagher, and M. Portmann, "E-GraphSAGE: A graph neural network based intrusion detection system for IoT," in *Proc. IEEE/IFIP Network Operations and Management Symposium (NOMS)*, Budapest, Hungary, Apr. 2022, pp. 1–9.
- [22] R. Darwish, M. Abdelsalam, and S. Khorsandroo, "Deep learning based XIoT malware analysis: A comprehensive survey, taxonomy, and research challenges," *Journal of Network and Computer Applications*, vol. 232, p. 104258, 2025.
- [23] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, Fort Lauderdale, FL, Apr. 2017, pp. 1273–1282.
- [24] V. Rey, P. M. S. Sa'nchez, A. H. Celdra'n, and G. Bovet, "Federated learning for malware detection in IoT devices," *Computer Networks*, vol. 204, p. 108693, Feb. 2022.
- [25] V. Mothukuri, P. Khare, R. M. Parizi, S. Pouriyeh, A. Dehghantanha, and G. Srivastava, "Federated-learning-based anomaly detection for IoT security attacks," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2545–2554, Feb. 2022.
- [26] M. Nobakht, R. Javidan, and A. Pourebrahimi, "SIM-Fed: Secure IoT malware detection model with federated learning," *Computers and Electrical Engineering*, vol. 116, p. 109139, Jun. 2024.
- [27] H. Lee, S. Kim, D. Baek, D. Kim, and D. Hwang, "Robust IoT malware detection and classification using opcode category features on machine learning," *IEEE Access*, vol. 11, pp. 18855–18867, 2023.
- [28] M. Venkatasubramanian, A. H. Lashkari, and S. Hakak, "IoT malware analysis using federated learning: A comprehensive survey," *IEEE Access*, vol. 11, pp. 57465–57493, 2023.
- [29] S. Garcia, M. Parmisano, and M. J. Erquiaga, "IoT-23: A labeled dataset with malicious and benign IoT network traffic," *Zenodo*, Jan. 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.4743746>
- [30] J. Guerra-Manzanares, J. Medina-Galindo, H. Bahsi, and S. No'mm, "MedBIoT: Generation of an IoT botnet dataset in a medium-sized IoT network," in *Proc. 6th International Conference on Information Systems Security and Privacy (ICISSP)*, Valletta, Malta, Feb. 2020, pp. 207–218.
- [31] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, Nov. 2015, pp. 1–6.
- [32] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, p. 5941, Jun. 2023.
- [33] K. Tam, A. Feizollah, N. B. Anuar, R. Salleh, and L. Cavallaro, "The evolution of Android malware and Android analysis techniques," *ACM Computing Surveys*, vol. 49, no. 4, pp. 1–41, Feb. 2017.

- [34] M. F. B. Abbas and T. Srikanthan, "Low-complexity signature-based malware detection for IoT devices," in *Proc. International Conference on Applications and Techniques in Information Security (ATIS)*, Auckland, New Zealand, Jul. 2017, pp. 181–189.
- [35] S. Kumar, K. Shankar, and A. K. Luhach, "Cybersecurity threats and trends in Internet of Things environments," *IEEE Access*, vol. 10, pp. 95578–95591, 2022.
- [36] A. Abusnaina, A. Khormali, H. Alasmay, J. Park, A. Anwar, and A. Mohaisen, "Adversarial learning attacks on graph-based IoT malware detection systems," in *Proc. IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, Washington, DC, Jul. 2021, pp. 1–10.
- [37] F. Yousaf, M. Arslan, A. A. Khan, A. Tanzil, A. Batool, and M. Asad, "Machine learning-based detection of Mirai and Bashlite botnets in IoT networks," *Journal of Computing & Biomedical Informatics*, vol. 7, no. 1, pp. 678–689, 2024.
- [38] T. Lei, J. Xue, Y. Wang, T. Baker, and Z. Niu, "An empirical study of problems and evaluation of IoT malware classification label sources," *Journal of King Saud University – Computer and Information Sciences*, vol. 36, no. 1, p. 101898, Jan. 2024.
- [39] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, proto-cols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [40] Zscaler ThreatLabz, "Zscaler ThreatLabz 2023 Enterprise IoT and OT Threat Report," Zscaler, San Jose, CA, USA, Tech. Rep., 2023. [Online]. Available: <https://www.zscaler.com/resources/industry-reports/threatlabz-enterprise-iot-ot-threat-report.pdf>
- [41] A. Kumar and T. J. Lim, "EDIMA: Early detection of IoT malware network activity using machine learning techniques," *Computers & Security*, vol. 116, p. 102675, May 2022.