A survey on IoT profiling, fingerprinting, and identification

Safi, Miraqa; Dadkhah, Sajjad; Shoeleh, Farzaneh; Mahdikhani, Hassan; Molyneaux, Heather; Ghorbani, Ali A.

Publisher's version / Version de l'éditeur:
https://doi.org/10.1145/3539736
ACM Transactions on Internet of Things, 2022-05-31

A Survey on IoT Profiling, Fingerprinting, and Identification

MIRAQA SAFI, SAJJAD DADKHAH, FARZANEH SHOELEH, and HASSAN MAHDIKHANI, Canadian Institute for Cybersecurity (CIC), Faculty of Computer Science, University of New Brunswick (UNB), Canada
HEATHER MOLYNEAUX, National Research Council Canada, Fredericton,Canada, Canada
ALI A. GHORBANI, Canadian Institute for Cybersecurity (CIC), Faculty of Computer Science, University of New Brunswick (UNB), Canada

The proliferation of heterogeneous Internet of things (IoT) devices connected to the Internet produces several operational and security challenges, such as monitoring, detecting, and recognizing millions of interconnected IoT devices. Network and system administrators must correctly identify which devices are functional, need security updates, or are vulnerable to specific attacks. IoT profiling is an emerging technique to identify and validate the connected devices' specific behaviour and isolate the suspected and vulnerable devices within the network for further monitoring. This paper provides a comprehensive review of various IoT device profiling methods and provides a clear taxonomy for IoT profiling techniques based on different security perspectives. We first investigate several current IoT device profiling techniques and their applications. Next, we analyzed various IoT device vulnerabilities, outlined multiple features, and provided detailed information to implement profiling algorithms' risk assessment/mitigation stage. By reviewing approaches for profiling IoT devices, we identify various state-of-the-art methods that organizations of different domains can implement to satisfy profiling needs. Furthermore, this paper also discusses several machine learning and deep learning algorithms utilized for IoT device profiling. Finally, we discuss challenges and future research possibilities in this domain.

CCS Concepts: • **Security and privacy** → **Intrusion/anomaly detection and malware mitigation**; • **Computing methodologies** → *Machine learning*.

Additional Key Words and Phrases: IoT Security, IoT profiling, IoT Fingerprinting, IoT Device Type Identification, Machine Learning.

1 INTRODUCTION

The number of smart devices connected through different Internet of things (IoT) methods is rapidly increasing. According to [39], consumer smart homes may deploy as many as 500 devices in their networks by the year 2022. A survey by [44] shows that there will be 25 billion connected IoT devices by 2025, which will contribute $1.1 trillion in industry revenue. As more devices are added to the IoT ecosystem, identifying and regulating millions of interconnected devices becomes increasingly challenging. As security issues arise in this heterogeneous environment, it is essential for organizations, network administrators, and consumers to have tools to detect and diagnose abnormal behaviour.

Authors' addresses: Miraqa Safi, mir@unb.ca; Sajjad Dadkhah, sdadkhah@unb.ca; Farzaneh Shoeleh, rzaneh.shoeleh@unb.ca; Hassan Mahdikhani, hmahdikh@unb.ca, Canadian Institute for Cybersecurity (CIC), Faculty of Computer Science, University of New Brunswick (UNB), Fredericton, Canada; Heather Molyneaux, Heather.Molyneaux@nrc-cnrc.gc.ca, National Research Council Canada, Fredericton,Canada, Fredericton, Canada; Ali A. Ghorbani, ghorbani@unb.ca, Canadian Institute for Cybersecurity (CIC), Faculty of Computer Science, University of New Brunswick (UNB), Fredericton, Canada.

2577-6207/2022/5-ART $15.00
https://doi.org/10.1145/3539736

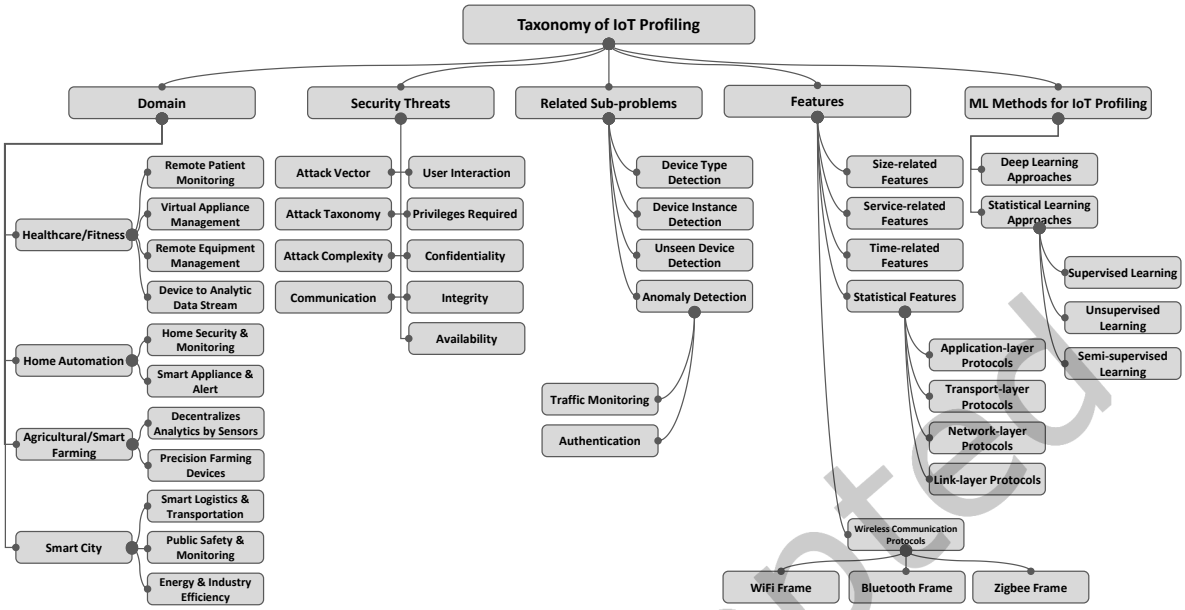

Fig. 1. The proposed taxonomy for IoT profiling techniques.

Profiling tools are a requirement in the evolution of the IoT ecosystem. Their significance must not be downplayed, as IoT devices can be the point of access an attacker needs to initiate a large scale attack [45]. For example, consider the Mirai botnet attack, which targeted a simple weakness in specific IoT devices. By exploiting a single vulnerable device in the network, attackers were able to infect over 60,000 IoT devices. The day after Mirai released, 15,000 cyber-attacks were launched, disrupting telecommunication, game servers, and several high-profile websites.

The nature of IoT devices makes them an easy target for infiltration [102]. The network-level security mechanism is essential for analyzing the traffic pattern and identifying attacks in heterogeneous IoT networks. Due to resource constraints, IoT devices use limited cryptographic capabilities for securing the data exchange. As a result, IP addresses, MAC addresses, and port numbers can be spoofed easily. Therefore, an IoT profiling technique is an essential approach to identify and monitor the connected devices' specific behaviour within the network. Utilization of profiling can prevent malicious network activity by promptly detecting and isolating suspected compromised devices for further observation.

IoT device identification is challenging because of the variance in device types, control sequences, and transmission protocols. A generalized procedure is not enough to accommodate the existing or future amount of IoT devices and their variations. Existing monitoring systems also do not have sufficient implementations to detect and recognize the existing set of devices that can be connected to a network. Additionally, IoT devices have limited storage and computation power, resulting in utilization of weak digital certificates and insecure cryptographic keys, or even no encryption at all. Due to their limited design, IoT devices are vulnerable to a wide range of cyber attacks, including but not limited to crucial leakage, broken authentication, and spoofing.

1.1 Contributions

Figure 1 shows the main contribution of this paper. This paper investigates several IoT device profiling techniques in different domains, such as Healthcare, Smart Home, Agriculture, Smart Farming, and Smart Cities. The following describes the different contributions of this paper:

- A thorough review of various IoT device profiling methods, such as IoT device type identification, individual device identification, unseen IoT device identification, and anomaly detection schemes in IoT environments.
- A clear taxonomy for IoT profiling based on several different security issues in this field.
- A detailed list of IoT devices' common vulnerabilities in multiple domains, along with corresponding security parameters. The proposed list of vulnerabilities can profoundly contribute to strengthening the risk assessment and mitigation of different organizations' networks.
- Common feature vectors used by various approaches to profile IoT devices such as device type identification, device instance identification, and unseen device identification, along with the efficiency and practicality of the features.
- Discussion of overhead related to feature selection and feature engineering in terms of practicality (i.e. the contribution of the feature to profiling) and efficiency (i.e. the complexity of extracting the feature). This part of the paper can assist researchers in selecting the most efficient collection of features for their profiling algorithms. By determining a set of features with the highest practicality and efficiency, it is possible to achieve better performance both in terms of accuracy and scalability for a proposed approach.
- Lastly, different machine learning and deep learning algorithms used for IoT device profiling are analyzed and compared. In addition, we investigate best performing machine learning classifiers employed by state-of-the-art IoT profiling.

Table 1 compares our work to similar works from 2014 to 2020 concerning different aspects such as Service Profiling, Smart Home, Smart City, Agriculture, Healthcare, Machine learning, security, and Privacy. After careful comparison o, as illustrated in the table, we have found that the distinguishing feature of our work is that we consider a wide range of industries, factors in security and privacy, but most importantly, address profiling techniques to identify key characteristics of IoT network traffic as a means of identifying malicious actors, as well as compromised or vulnerable devices.

2 PROBLEM DEFINITION

IoT devices are extremely diverse in build standard. Even for devices of the same type, there can be variations in control sequences, transmission protocols, and encryption standards. Environments that utilize IoT devices often have a multitude of devices deployed, varying in type and manufacturer. The inability for an administrator to easily identify devices and healthy behaviour can be detrimental for network security and operation.

The significance of the problem increases as more devices are added to the network. Due to the lack of uniformity, network administrators have an increasingly time-consuming problem of configuring various rules and permitted behaviours for deployed IoT devices. Thus, profiling tools for monitoring, detecting, and identifying thousands of interconnected IoT devices is essential for differentiating standard behaviour from anomalous behaviour. Profiling will significantly improve modularity of connected IoT devices and help map particular nodes to specific sectors. For instance, network administrators may wish to limit specific device activity to particular periods of the day. Profiling and identifying the device enables network administrators to see if devices are operating with intended privileges, or locate vulnerable devices and trigger security updates or patches. If a specific device is violating a baseline profile, it should be quarantined for further monitoring. The network traffic in the IoT environment is heterogeneous; making it difficult to create fixed profiles to identify each device.

One of the most significant challenges to consider in IoT profiling is selecting a small, yet effective set of features. Several works published regarding IoT profiling utilize features related to device network traffic. Then,

Table 1. Comparison of Similar Surveys to this Work

Survey	Device Profiling	Smart Home	Smart City	Agriculture	Healthcare	Machine Learning	Security	Privacy
[121]		✓	✓		✓		✓	✓
[142]					✓		✓	
[144]		introduced	introduced		introduced		✓	✓
[134]						✓		✓
[69]		✓			✓		✓	✓
[107]		✓					✓	✓
[50]					✓		✓	✓
[37]				✓			✓	
[9]		✓						
[32]			✓				✓	
[42]			✓		✓		✓	✓
[64]		✓	✓			✓	✓	✓
[65]		✓					✓	✓
[72]			✓				✓	✓
[79]			✓				✓	✓
[109]					✓		✓	
[70]		✓	✓		✓		✓	✓
[11]					✓		✓	✓
[38]		✓						
[92]				✓				
[61]				✓				
[53]			✓				✓	✓
[48]							✓	✓
[90]	✓						✓	✓
[146]	✓	introduced			introduced	✓	✓	✓
Our Work	✓	✓	✓	✓	✓	✓	✓	✓

Index: [✓] = Discussed Thoroughly, [introduced] = Briefly Discussed or Mentioned, [] = Not Discussed or Mentioned.

they perform fundamental operations on the feature set, train a machine learning model using the set, and create a profile for the device based on the results [25, 49, 91, 95, 117]. Profiling IoT devices based on their network behaviour has been an area of research for a long time. Examples of behaviour features that can be used in IoT profiling are DNS server, set of protocols, request-response pattern, packet size, and frequency of the message. The traffic features of an IoT device can be classified into the following types:

- Size-related features: IP Packet size, Payload length, TCP Window size, Flow size, Traffic volume, Traffic rate.
- Service-Related features: IP address, Protocol number, Port number, DNS, and NTP query.
- Time-related features: Flow duration, active, and sleep duration.
- Statistical Features: Flow size, Minimum, Maximum, etc.
- Network communication protocols-long Range (Cellular, LoRa (LoRaWAN, Ingenu, WiMAX))
- Network communication protocols-Short Range (Bluetooth Smart (BLE), Zigbee, Wi-Fi, NFC, EnOcean, Wireless HART, Z-Wave, 6LoWPAN)

As Figure 2 shows, the prominent architecture of the IoT profiling system includes:

- Data acquisition: The network traffic and other necessary information of the IoT devices are captured. This raw traffic generated by IoT devices could be imbalanced due to the nature of the IoT devices.
- Feature extraction: A list of features is extracted from the captured data. The features are selected in a way that could identify most of the devices with better accuracy. This feature engineering is an important stage in IoT profiling and should be selected carefully.

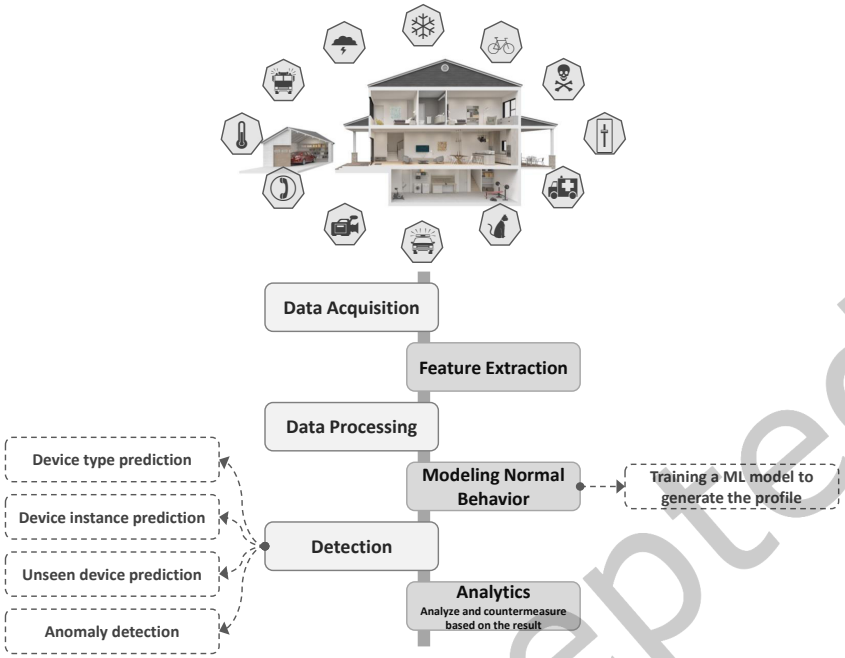

Fig. 2. General Architecture of IoT Profiling.

- Data pre-processing: The features are pre-processed and those features which could differentiate better between the devices are selected.
- Modeling normal behaviour: The expected behaviour of the ideal IoT device creates a signature for the normal behaviour of the IoT devices using a machine learning algorithm. Selecting the best machine learning algorithm in this stage is also an important step toward profiling the devices. The moment a device violates this normal profile it could be quarantined for further monitoring.
- Detection: In this stage, the type of device, an instance of the device, anomalous device, and unseen devices that did not exist in the network while training the model are identified. Lastly, the information is analyzed to deduce the proper actions to take against the identified device. However, modelling and creating a proper profile for each device requires in-depth research about the IoT device's nature before starting any ML training algorithm. The IoT devices' baseline profiles connected to the network must be securely stored for future profile comparison. Finally, actions such as discarding the device from the network or quarantining the device can be taken.

As mentioned before, with the fast growth of smart devices, security issues are arising in the heterogeneous IoT environment. Remotely monitoring, detecting, and recognizing millions of interconnected smart IoT devices is essential for operational and security motivations to detect anomalous behavior. However, several researchers attempted to conduct different studies in behavioral analyses or profiling IoT devices throughout recent years. Figure 3 illustrates the evolvement of the IoT profiling method from 2017 to 2021.

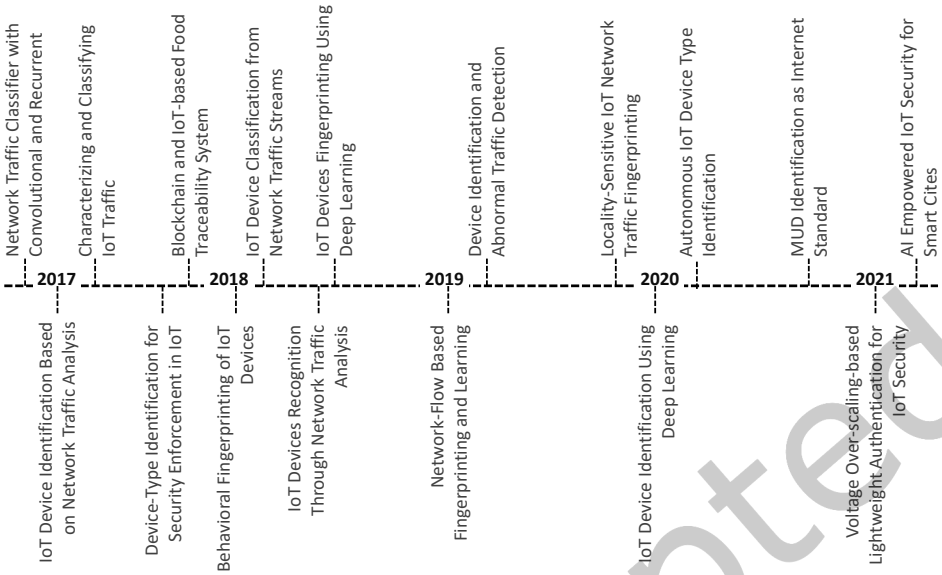

Fig. 3. IoT Profiling Evolvement.

3 DOMAINS OF IOT SECURITY PROFILING

Security Profiling of IoT devices is rapidly emerging and spans several industries and services. The survey provided by the authors of [142] detail the application of IoT in several industries. We will further expand by giving information on how profiling techniques can improve IoT performance and security in various domains. A poor security profiling mechanism can degrade performance. It can be even worse when it comes to IoT devices. The patient lives in IoT-based healthcare, the safety of home appliances and security of elements in smart farming and the smart city could be in danger exploited due to malicious activities. Some most common domains of IoT profiling, as shown in Figure 4 are:

- Healthcare/Fitness IoT (Remote Temperature Monitoring Drug effectiveness tracking, Sleep monitoring, Medication refill reminder, Nutrition Control, Contact-less Surgery, Chronic Care Management, Remote Treatment Solutions, Blood glucose and diabetes monitor, Assisted living, Asthma Inhaler System)
- Home Automation (Home security and monitoring systems,Smart Lighting,Smart kitchen,Smart sound system)
- Agriculture/Smart Farming (Climate Monitoring System, Greenhouse Automation System, Crop Management, Autonomous irrigation, Autonomous Tractor, Smart Harvesting)
- Smart City (Law Enforcement, Disaster Management, Security and Surveillance, Oil Exploration, Smart Transport, Energy Management, Smart Mining)

3.1 IoT Security in Healthcare/Fitness

IoT devices in healthcare have an essential role in the advancement of healthcare industries [70]. With the massive amounts of data transmission to and from these devices, it is challenging to create a secure and effective profiling framework for healthcare IoT devices. The consequences of an insecure healthcare IoT network are highly significant. Attacks on data privacy and integrity could have significantly detrimental results for both

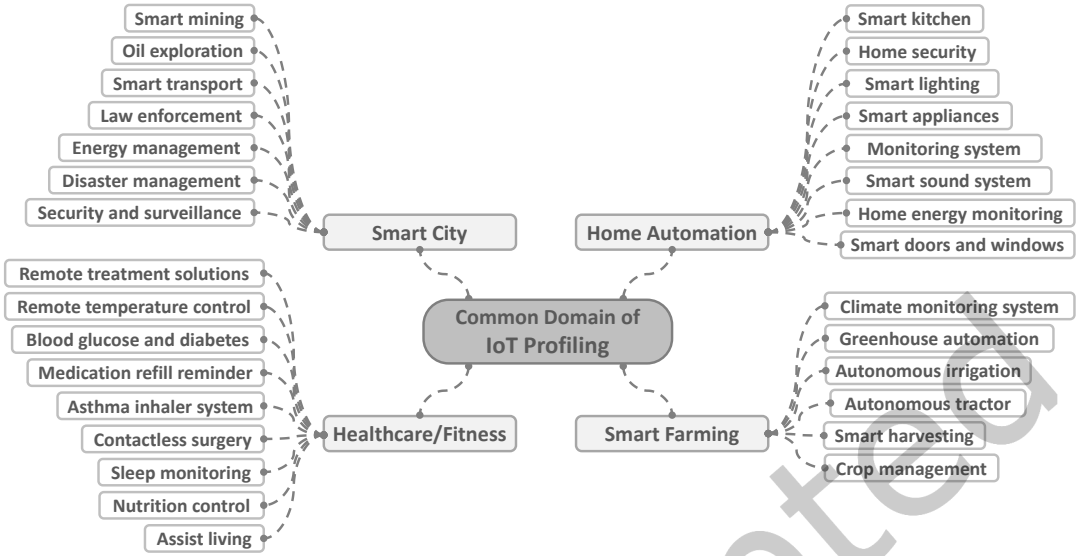

Fig. 4. Common domains of IoT Profiling.

the organization and the patients. In addition, delays in IoT device operation or abnormal behaviour from compromised devices could be a threat to a patient's life. Therefore, it is essential to have a secure and efficient profiling framework for healthcare IoT.

As many smart devices are being used in the health domain, dissecting the privacy and security issues associated with those devices is an area of interest for researchers. Several researchers have surveyed the privacy and security issues of IoT devices in healthcare sectors [6, 52, 98, 103, 106, 109, 119, 129]. Most of these have provided an overview of privacy and security issues and solutions in different health industry areas.

Security demands such as integrity, availability, authorization, authentication, trust management, and privacy for each layer of healthcare IoT devices to recognize the network's malignant hubs are explained by [67]. In [85] they have referred to privacy, confidentiality, authentication, access control, and trust management of healthcare-IoT. A reliable architecture based on international healthcare cybersecurity regulations and standards to protect patient's health information and privacy is proposed in [127]. It enhances the reliability of IoT devices in healthcare by lessening the vulnerabilities. [17] introduces a cloud-based model, stating that it is the best option for IoT implementation in healthcare. Cloud services provide virtually unlimited storage space and allow accessibility for both patients and doctors. Figure 5 shows the importance of IoT devices in the Healthcare environment. As Figure 5 demonstrated, it is expected that the demands of IoT devices in the healthcare market will be at least twice by the year 2025.

3.2 IoT Security in Home Automation

Even at the consumer level, many IoT devices are being connected to create smart home environments. There is a significant increase in IoT devices used in the smart home environment. To facilitate access, provide equipment, and allow users to control remotely and manage their home environment, such as remote working smartwatches, intelligent lights, baby monitors, they analyze their data in the cloud [102]. A 2019 iProperty management study found that over 26 million IoT devices run in the U.S. [83]. Another report says 120 IoT devices connect to the internet per second globally [84].

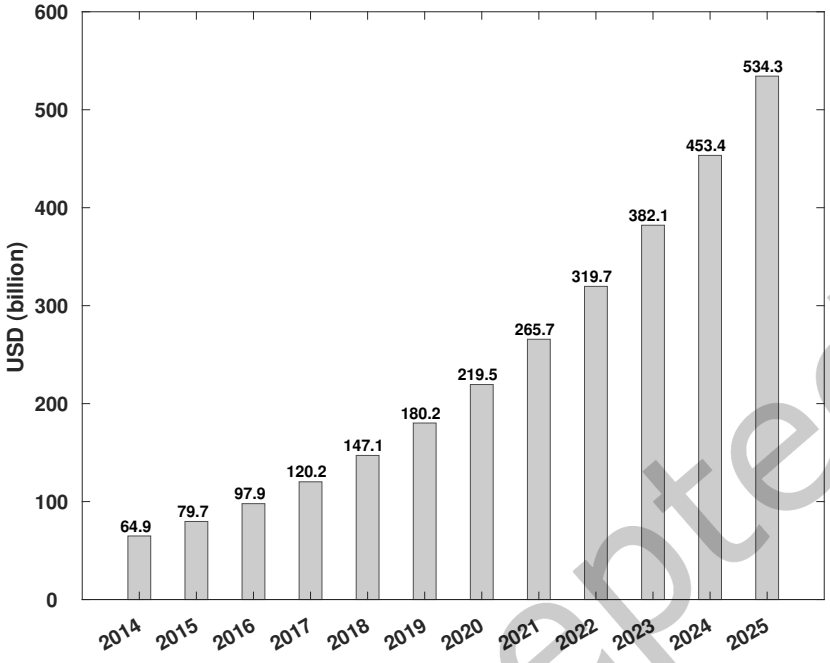

Fig. 5. IoT in the healthcare market. [?]

Deployment of consumer-grade IoT devices in smart home environments is also a concern for security and privacy. In early 2014, attackers hacked over 100,000 IoT devices and conducted a large-scale attack [22]. Manufacturers of these devices could also be collecting confidential user data for analytics or consumer profiling. Therefore, IoT profiling is a necessary consideration even at the consumer level.

Message integrity and security model authentication based on hash functions and encryption algorithms in the smart home using smartphone messages is encrypted with AES-256, RC4-based hash function, and key exchange ephemeral Diffie-Hellman [86]. A central hub is used for message control, and messages are transmitted by phone to the central transmitting hub. The authors propose this three-layer strategy as an energy-efficient means of authentication, key exchange, and protecting message integrity between the smartphone and the central hub. As Diffie-Hellman consumes significant energy, it is used only once for the user's initial authentication. AES256 is used to complete the cipher, as it is energy and resource-efficient. The RC4 hash function is used to preserve message integrity compared to MD-based and SHA-based hash functions.

IoT devices' attack vectors are growing, and maintaining home safety and protection is getting more difficult. [125] proposes a three-party software solution that allows users to configure network-level security for their home devices. Their solution, the security management provider (SMP), utilizes API calls to restrict access control or block invasive activities on IoT devices. Improved privacy protection criteria for the intelligent home environment and a security attack investigation focused on historical evidence were suggested by [10][74]. They anticipated security threats and projected the number of potential threats. The architecture includes; access control, management, cryptography, system integrity, authentication, logging, and digital signature.

Things are categorized based on the spatial and temporal characteristics of the underlying device-to-device, and owner-to-cloud interface [107]. In the smart home ecosystem for device integrity, and improved authentication

mechanism called the self-signing technique is proposed by [65]. The self-signing and access control prevent data from leakage, modification, and code fabrication. A lightweight lattice-based framework was proposed for the smart home to maintain privacy and track authentication [4].

In [140], authors used Android application and the GSM system to create a smart home protection system. It remotely controls smart home appliances like refrigerators, AC lighting, and TV and creates an alert in case of intrusion or security problems. [14] reported that even with encryption mechanisms in place, a passive adversary is still able to infer device activity from generated network traffic. The authors propose a privacy-preserving traffic shaping scheme to mask the channel operation. If the formed traffic rate is lower than the system's traffic, packets are queued. Otherwise, dummy packets are inserted to cover the difference in traffic rate. They report that the suggested traffic shaping scheme provided is accessible to users with good internet connections and that their scheme can be tuned to user needs. In [9], they have presented an overview of how IoT is blended with artificial intelligence to connect and monitor devices within a smart home environment remotely.

The most relevant limitation of IoT devices and their solutions, classification of IoT attacks, mechanisms for authentication and access control, analyzing the security issues in different layers, and presenting a layered architecture of IoT-based smart home is discussed by [75][144][38]. IoT devices' security risks and privacy issues and the risk element of smart home IoT devices that take into account four types of attacks: software, network, physical, and encryption are summarized [69][33][13]. This classification helps in a more comprehensive view of how to inspect vulnerabilities.

During a joint workshop session with nine participants, the risk analysis of a smart home automation system was carried out [54] [55]. The talk was organized using an open information security risk evaluation survey for reasoning, identity, analysis, and assessment of different threats. Several common threats such as eavesdropping, impersonation, DoS, ransomware that could be used on a smart home IoT environment by a cybercriminal was evaluated in [40, 68, 73].

3.3 IoT Security in Agriculture/Smart Farming

Lately, smart technologies have become an indispensable part of every agriculture strategy since they provide most of the world's food and fabric. Smart devices can help manage the farming system's performance, maintenance, and cost tracking. IoT is groundbreaking in smart farming and agriculture, where farmers can remotely track their crops and machinery. Smart sensors are used to calculate humidity, temperature, water level, soil nutrition levels, soil PH, etc. However, the security of IoT devices and smart sensors is vital in smart farming and agriculture to prevent the farmer's risks and financial loss and disrupt the country's economy.

In smart farming, IoT introduces a broad exposure to cybersecurity risks and vulnerabilities. The number of IoT devices deployed in intelligent agriculture is estimated by BI Intelligence to grow from 30 million in 2015 to 75 million by 2020 and is projected to produce 4.1 million data points per day in 2050 [93]. The security and privacy of smart farming, potential cyber-attacks, vulnerabilities, risks, and the financial impact of using smart devices in agriculture from a business perspective are discussed in detail by [46][51][87][18][57].

High gain and low maintenance agriculture systems can be built using eco-friendly and energy-efficient smart sensors. The application of IoT in agriculture to help improve crop management, resource management, cost-effectiveness, quality and quantity improvement, crop tracking, etc., are addressed by [31][97][126]. To recognize environmental and human-made factors hindering plant growth, they have merged IoT and image processing [60].

In order to understand various technologies and develop a smart farming model with a wireless network, the authors of [92] surveyed several implementations of agricultural IoT using cloud computing. An agricultural use of wireless sensor networks for crop field tracking has been suggested [81, 92**?**]. Their systems consist of sensors that measure temperature, humidity, and image sensing by taking crop images for information.

In [62], they recommended an IoT-based greenhouse monitoring system with a cloud to track multiple environmental parameters. Every 30 seconds, data from the temperature sensor, light sensor, soil humidity sensor, and relative humidity sensor are continuously sent to the cloud. A framework is put in place to understand the vulnerabilities in emerging technologies, especially in the smart farming environment by [138]. To develop an integration framework for smart farming, they used support vector machine (SVM) and Artificial Neural Networks in [63]. Through developing a system to ensure and track the food processing cycle, the authors in [78][**?**][108] concentrated on the application of blockchain technologies for food safety. For example, [78] proposes an ERP system applied to IoT, with data resting on a blockchain. They provide a use case, stating that a user can scan a barcode using their smartphone, then track the transaction history for their food product, allowing them to verify food safety.

Many facets of IoT in agriculture are defined and clarified by smart farming based on IoT [37] [20] [29] [116]. It highlights the security concerns and the complexities of automated virtual farms using wireless sensor networks. To concentrate on the physical layer of smart farming, a low-cost IoT based security monitoring system was proposed and implemented by [116].

3.4 IoT Security in Smart City

The development of smart cities is faced with many issues, including socioeconomic, political, and technical issues, but most importantly, security and privacy issues [100]. Security threats, vulnerabilities, and solutions that are proposed from 2010 to 2015 for the smart city environment are provided by [53]. They offered a comprehensive, categorized, and detailed overview based on security problems in the existing smart city designs. In developing a smart city, the safety and security issues of machine to machine (M2M) standard solutions are addressed by [19].

In [35], they addressed cybersecurity issues and concentrated on two main challenges: privacy and security. To explain the contact between the IoT, people, and servers. The role of intelligent software in developing a smarter city and security limitations is discussed by [115]. A distributive framework for ensuring trust, privacy, and security of information transfer in IoT is proposed by [21]. [137] addresses the issue of safety problems in urban growth. They proposed an encryption method to cope with data integrity and privacy concerns. Smart grids are an essential part of a smart city as they provide efficient energy supply chain networks and information management [30]. [80] discussed the problems of network security in the Smart Grid. The link between anonymity and security was addressed and showed that they could be matched in smart grids[43]. An automatic system is suggested that can be managed for more immeasurable smart city protection through distributed cloud concepts by [128].

The study of privacy problems such as query privacy, the privacy of identity, privacy of footprint, the owner's privacy, and privacy of location for an IoT device in the smart city is presented by [89]. The IoT architecture, protocols, and security issues, explicitly considering the smart city IoT application, are discussed in [32]. The comprehensive surveys provided in [121] and [72] also concentrate on the privacy and security of IoT in the smart city domain. IoT implementation, configuration, execution, and management using an application execution platform (AEP) in the smart house, smart city, and the smart car is presented by [71]. A description of the IoT device architecture and fog/edge computing integration in IoT applications is presented by [79]. They have also discussed privacy and security issues. The role of edge computing and critical requirements in the smart city is discussed by [64].

Data management approaches of smart IoT devices used for granularity, reusability, consistency, and interoperability are surveyed by [42]. Characteristic, deployment, architecture, and challenges of IoT in the smart city using big data is discussed by [77] [8] [122].

Table 2. Papers Discussed in Section 3, Classified by Industries Provided

Category	References
Healthcare	[70], [47], [6], [52], [106], [103], [119], [109], [98], [129], [67], [11], [85], [127], [17], [59], [59], [50]
Smart Home	[102], [83], [84], [22], [86], [125], [10], [86], [125], [10], [74], [107], [65], [4], [140], [14], [9], [41], [96], [75], [144], [38], [69], [33], [13], [54], [55], [40], [73], [68]
Agriculture	[93], [46], [51], [87], [18], [57], [31], [126], [60], [92], [?], [81], [62], [138], [63], [78], [?], [108], [61], [110], [23], [37], [20], [29], [116]
Smart City	[100], [53], [19], [35], [115], [21], [137], [30], [80], [43], [128], [89], [32], [121], [72], [71], [79], [64], [42], [77], [8], [122]

4 SECURITY IN IOT ENVIRONMENT

Due to rapid technological advancements, multiple devices in the IoT environment, such as mobile devices, embedded systems, sensors, and actuators(considered smart devices), can receive vast amounts of information via data trading and interconnection. It is essential to preserve individual privacy and secure, shared data in this context. Therefore, privacy and security have drawn significant attention and research interest in recent decades. Several security solutions have recently been discussed for the IoT environment. We address different security solutions in this domain in the context of different security analyses and security threats in IoT profiling with the following subsections.

4.1 Possible Security Analysis in IoT Profiling

IoT Profiling is an emerging and rapidly evolving field. In the most simplistic form, devices can be identified by their IP and MAC addresses. However, IP addresses can change frequently and MAC addresses can be spoofed. Moving towards a more reliable measure of identifying devices, researchers began utilizing feature selection with machine learning models to model device network behaviour, as seen by works referenced in Table 6. Now, with MUD becoming internet standard, manufacturers are able to provide specification for device information and intended communication patterns, allowing consumers to easily identify devices in their network by checking MUD components [**?**]. [**?**] further explains these components as the following, a URL provided on device connection to the network, a file which the URL points to, and a process that retrieves the file. The file defines the minimum communication access the device needs for normal access.

Securing IoT device communications is vital to create a robust network. There have been frequent prior cases of IoT devices passing user-sensitive information in clear-text, or manufacturers collecting more information than they are obligated to [148]. As TLS is internet standard, it has been used by many manufacturers as the means of securing communication between device and cloud server. However, as [148] states, this has a converse effect on security, making it harder for users to detect data leaks, as the information cannot be inspected by entities other than the communication endpoints. Further, [148] proposes inspection-friendly TLS (IF-TLS), which connects the device to a manager, that routes device traffic to middle-boxes, who can inspect the data in transit.

In contrast, some end-to-end countermeasures are given by [133]. These countermeasures are IP security (IPSEC), TLS encryption, generic routing encapsulation (GRE), and Tunneling. While the proposed countermeasures are end-to-end, the author's proposal of GRE encapsulates data payloads with checksum, key, and Sequence Security Flags, without encrypting the data. This allows parties not being directly communicated with to evaluate the traffic in transit. While this means the data is potentially visible to attackers, utilizing tunneling to create a virtual connection between two endpoints will allow the user to evaluate the data while protecting it from man-in-the middle attacks. [146] proposes the utilization of Deep Learning as a means of analyzing the security of IoT, as well as profiling device behaviour. They state that current methodologies of identifying devices using static identifiers is a security risk, as attackers can spoof these. By using Deep Learning, they claim that training a model using system logs and web traffic will allow the Deep Learning model to identify subtleties in legitimate and falsified requests. The profiles created by the Deep Learning model can also be utilized to determine the type

of the device based on the network traffic. Figure 6 illustrates some of the well-known vulnerabilities utilized in the IoT environment for different attacks.

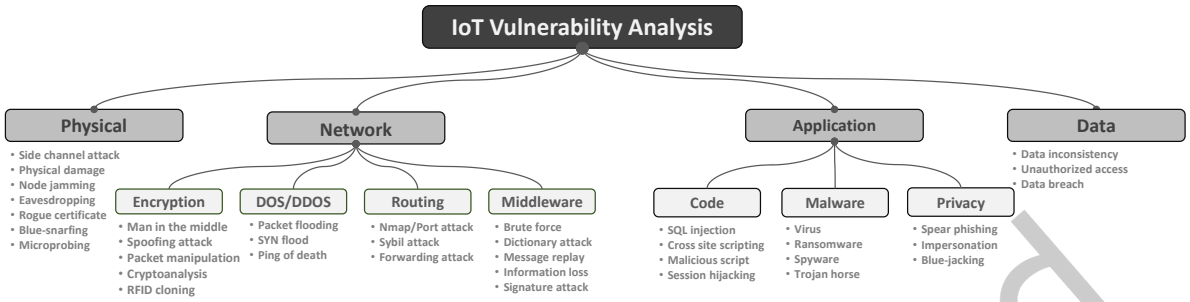

Fig. 6. The Security Vulnerability in IoT Profiling.

4.2 Security Threats in IoT Profiling

Vulnerability in an IoT device could replicate, as most IoT devices' software is developed using code clones and third-party libraries [147]. Since these clones and libraries play an essential role in developing the embedded software, different devices can have the same vulnerabilities. Figure 7 illustrates the security threats that can have a high impact on IoT profiling systems.

Li et al. [76] indicated that 22.3% of the Linux kernel's code of the IoT devices had been implemented previously in common libraries or other similar devices. This could unintentionally introduce the same vulnerabilities to millions of devices. Table 3 shows common vulnerabilities available in the IoT devices along with their main properties, which are listed below. In addition, Table 10 presents a detailed description of each vulnerability.

- CVSS Base Score: Shows the severity of the vulnerability.
- Type: Refers to the type of vulnerability.
  - Software (S): Vulnerable component is present in the software. Most of the IoT devices have software vulnerability.
  - Hardware (H): The vulnerable component is present due to the hardware vulnerability.
- Attack Vector: The attack vector's value increases if the attacker is physically and logically exploiting a particular vulnerability.
  - Network(N): It shows that the vulnerability is related to the network stack and the attacker can use the network layer (e.g. router) to exploit the vulnerability remotely.
  - Adjacent Network(A): It indicates that the vulnerability is bound to the network stack and limited to the same physical or logical network. It can not be performed across the boundary of the third layer of the OSI model, i.e., a router.
  - Local(L): In some cases, the vulnerability is not restricted to the network stack, and the attack can happen via reading/writing/executing capabilities. So, the attacker should be logged in or rely on user interaction to exploit the vulnerability.
  - Physical(P): This type of vulnerability requires physically touch or manipulate the unsafe components, e.g. attaching a peripheral device to the system.
- Attack Complexity: It shows that the conditions beyond the attackers' control require exploiting the vulnerability. e.g., gathering more information about the target or knowing specific system configurations.
  - Low(L): An attacker can expect repeated success against the vulnerability and there is no need for specialized access conditions.

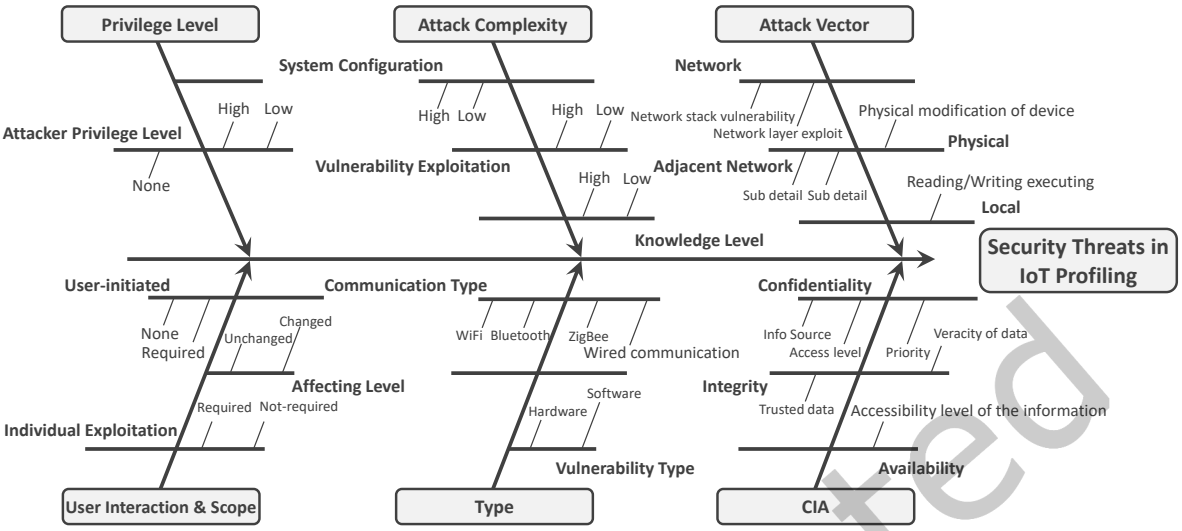

Fig. 7. The Security Threats in IoT Profiling.

  - High(H): It requires the attacker to spend some measurable amount of effort into exploiting the vulnerable component. Successful exploitation requires conditions beyond the attacker's control.
- Privileges Required: It is related to the level of privilege an attacker should have before exploiting the vulnerability.
  - None(N): The attacker needs no access to files or settings to accomplish an attack (unauthorized attacker).
  - Low(L): The attacker should have some basic user privileges to impact only non-sensitive resources, for example, typically affect the user's files and settings (authorized attacker).
  - High(H): The attacker needs privileges for significant control over the vulnerability and could affect component wide settings and files (authorized attacker).
- User Interaction: Exploiting some vulnerabilities requires the participation of users. This property shows a separate user-initiated process should participate in exploiting a vulnerability, or the attacker can exploit it individually.
  - None(N): No user interaction is needed to exploit the vulnerability.
  - Required(R): Requires the user to take any action before exploiting vulnerability, e.g. convincing the user to click a link in an email.
- Scope: It shows whether a vulnerability in one software can affect other resources beyond its privileges and means.
  - Unchanged(U): In this case, the vulnerable component and impact component are the same. It means that an exploited vulnerability can only impact on the resources managed by the same authority.
  - Changed(C): some vulnerabilities can impact other resources which are beyond the authorization privileges. So, the vulnerable components and impact components are different.
- Confidentiality: It refers to the impact on information sources' confidentiality, which means that data and resources are protected from unauthorized access. Thus, only authorized users can access and disclose information.
  - None (N): There is not any loss of confidentiality by exploiting the vulnerability.

  - Low (L): The attacker can access to some restricted information but without any control over what information is. The disclosed information does not cause a severe or direct loss.
  - High (H): It shows a total loss of confidentiality, where the attacker can access to some restricted information which may causes a direct and serious impact.
- Integrity: It refers to the trustworthiness and veracity of the data, the impact on information sources' integrity.
  - None (N): There is no loss of integrity within the vulnerability.
  - Low (L): The attacker can modify data, but it is impossible to have control over the consequence of modification. The modification does not have a severe and direct impact on the impacted components.
  - High (H): The attacker can modify any/all protected files; means complete loss of protection. These changes would present direct, severe consequences to the impacted components.
- Availability: It indicates the impacts on the accessibility of information resources.
  - None (N): There is not any impact on availability within the vulnerability.
  - Low (L): The attacker is not able to completely deny services to authorized users. There are no direct, severe consequences for the impacted component.
  - High (H): It shows a total loss of availability, which means the attacker can entirely deny access to the impacted components' resources. The loss of availability presents severe and direct consequences to the affected components.
- Communication: Presents what type of communication is used by the device, it could be Wi-Fi, Bluetooth, Zigbee, or wired communication.

Table 3 represents a list of common vulnerabilities related to IoT devices, which are found and labelled based on the information from common vulnerability exposure (CVE), national vulnerability database (NVD), common weakness enumeration, and blog posts about the IoT. If a specific vulnerability is not present in CVE, NVD, CWE, its properties are specified with the most suitable label. As an example, consider one of the Amazon Alexa's vulnerabilities, the one with *V-4* id, its type is hardware (H), and its attack vector is labelled as local (L) because the attacker needs to be in local proximity to the device. The attack complexity is labelled high (H) because the attacker should spend some measurable effort exploiting the vulnerable component. The privilege required is labelled none (N) because the attacker can attack without access to settings or files. User interaction is labelled none (N) because the attacker can exploit this vulnerability without user interaction. Since the vulnerability can impact resources beyond the authorization privileges, e.g., turning on the light or opening the door, its scope is changed (C). Confidentiality is none (N) because There is no loss of confidentiality within the vulnerability. Integrity is labelled high (H) because of a complete loss of protection. Since there is no impact on availability, this property is labelled as none (N). Finally, having the above approximation, the base score for this vulnerability is calculated using the CVSS base score calculator [**?**].

5 RELATED PROBLEMS FOR IOT SECURITY PROFILING

The existence of an increasing number of IoT devices creates a challenge for network administrators and industries in identifying and monitoring the heterogeneous IoT environment at the same time. It creates a point of intrusion for hackers.

IT administrators should be able to quickly identify what IoT devices are connected or trying to connect to their network. Hospitals, homes, cities, farms, and campuses would be equipped with vast numbers of IoT devices. It is essential for the operators of such an environment to monitor, control, and maintain IoT devices based on their network behaviour and to have visibility of all IoT assets; whether they are functioning or not, to ensure they are safe from cyber threats.

Table 3. All list of possible vulnerabilities of different IoT devices along with their properties.

Device	IDs	Citation	Type	Vec.	Comp	Pri.	UserInt.	Scope	Conf.	Int.	Avail.	Comm.
Amazon Echo	V-1	5.3 MEDIUM	S	A	H	N	N	U	N	H	N	Wi-Fi
	V-2	3.3 LOW	S	L	L	N	R	U	L	N	N	Wi-Fi
	V-3	6.8 MEDIUM	S	A	H	N	N	U	H	H	N	Wi-Fi
Amazon Alexa	V-4	5.9 MEDIUM	H	L	H	N	N	C	N	H	N	Wi-Fi
	V-5	8.9 HIGH	H	N	H	N	N	C	L	H	H	Wi-Fi
	V-6	6.8 MEDIUM	S	N	H	N	R	U	H	H	N	Wi-Fi
	V-7	5.9 MEDIUM	S	N	H	N	R	U	H	L	N	Wi-Fi
Apple Siri	V-4	5.9 MEDIUM	H	L	H	N	N	C	N	H	N	Wi-Fi
	V-5	8.9 HIGH	H	N	H	N	N	C	L	H	H	Wi-Fi
Amazon FireTV Cube	V-4	5.9 MEDIUM	H	L	H	N	N	C	N	H	N	Wi-Fi
Google Home (Google Assitant)	V-5	8.9 HIGH	H	N	H	N	N	C	L	H	H	Wi-Fi
dbell Wi-Fi Smart Video Doorbell DB01-S Gen 1	V-8	9.8 CRITICAL	S	N	L	N	N	U	H	H	H	Wi-Fi
Amazon's Ring Doorbell Camera	V-9	9.3 CRITICAL	S	N	L	N	N	C	H	H	N	Wi-Fi
August Door Lock	V-10	6.8 MEDIUM	S	N	H	N	R	U	H	H	N	Wi-Fi
August Doorbell Cam and Cam pro	V-11	5.9 MEDIUM	S	A	H	N	N	U	N	H	N	Wi-Fi
Belkin Wemo Insight Smart Plug	V-12	10 CRITICAL	S	N	L	N	N	C	H	H	H	Wi-Fi
Belkin Wemo Enabled Crock-Pot	V-13	9.8 CRITICAL	S	N	L	N	N	U	H	H	H	Wi-Fi
Belkin N750	V-14 V-15	9.8 CRITICAL	S	N	L	N	N	U	H	H	H	Wired
	V-16	7.5 HIGH	S	N	L	N	N	U	N	H	N	Wired
Belkin WeMo	V-17	5.9 MEDIUM	S	N	H	N	N	U	H	N	N	Wired
	V-18	8.7 HIGH	S	N	H	N	N	C	H	H	N	Wired
	V-19	8.0 HIGH	S	A	H	N	R	C	H	N	H	Wired
	V-20	8.0 HIGH	S	A	H	N	R	C	N	H	H	Wired
	V-21	8.1 HIGH	H	N	H	N	N	U	H	H	H	Wired
Belkin N600 Db Wi-fi Dual-band N	V-22	8.6 HIGH	S	N	L	N	N	C	N	H	N	Wired
+ Router F9k1102 Firmware	V-23	9.8 CRITICAL	S	N	L	N	N	U	H	H	H	Wired
	V-24 V-25	8.8 HIGH	S	N	L	N	R	U	H	H	H	Wired
Wemo Switch, Motion Sensor, Insight, Light Switch, Dimmer, Switch Mini, Link, Slow Cooker, Humidifier, Coffee Maker, Heater, Netcam HD+	V-26 V-27 V-28 V-29 V-30 V-31	5.3 MEDIUM	S	A	H	N	N	U	H	H	N	Wi-Fi
Honeywell Performance Series NVRS	V-32	5.3 MEDIUM	S	N	L	N	N	U	L	N	N	Wi-Fi
	V-33	8.1 HIGH	S	N	H	N	N	U	H	H	H	Wi-Fi
Honeywell Intermec PM Series Smart Printers	V-34	8.8 HIGH	S	L	L	L	N	C	H	H	H	Wired
Honeywell Midas Gas Detectors	V-35	9.0 CRITICAL	S	N	H	N	N	C	H	H	H	Wi-Fi
Honeywell Tuxedo Touch	V-36	5.8 MEDIUM	S	A	H	N	R	C	L	L	L	Wi-Fi
	V-37	5.0 MEDIUM	S	N	H	N	R	U	L	L	L	Wi-Fi
LG SmartThinQ Mobile App	V-38	6.8 MEDIUM	S	N	H	N	N	C	N	N	H	Wi-Fi
Netatmo Indoor Camera	V-39	6.7 MEDIUM	S	L	L	H	N	U	H	H	H	Wi-Fi
Nest Cam IQ indoor	V-40	9.0 CRITICAL	S	N	H	N	N	C	H	H	H	Wi-Fi
	V-41	7.5 HIGH	S	N	L	N	N	U	N	H	H	Wi-Fi
	V-42	5.3 MEDIUM	S	N	L	N	N	U	L	N	N	Wi-Fi
	V-43 V-44	7.5 HIGH	S	N	L	N	N	U	N	N	H	Wi-Fi
	V-45 V-46	8.8 HIGH	S	N	L	N	R	U	H	H	H	Wi-Fi
	V-47	7.5 HIGH	S	N	L	N	N	U	H	N	N	Wi-Fi
Nest Thermostat	V-48	6.8 MEDIUM	H	P	H	N	N	C	N	H	H	
Philips Hue Light Hue 2.1	V-49	7.9 HIGH	S	A	H	N	R	C	H	H	H	Zigbee
Samsung SmartThings Hub	V-52 V-53 V-58 V-60	8.8 HIGH	S	N	L	L	N	U	H	H	H	Zigbee
	V-50 V-57 V-59 V-61-V-63 V-67 V-69 V-70	9.9 CRITICAL	S	N	L	L	N	C	H	H	H	Zigbee
	V-64	8.2 HIGH	S	L	L	H	N	C	H	H	H	Zigbee
	V-65 V-66	8.6 HIGH	S	N	L	N	N	C	N	H	N	Zigbee
	V-68	7.5 HIGH	S	N	L	N	N	U	N	H	N	Zigbee
	V-71	5.9 MEDIUM	S	L	L	L	N	U	N	N	H	Zigbee
	V-72	5.9 MEDIUM	S	N	H	N	N	U	H	N	N	Zigbee
Sonos Wireless Speaker	V-73	9.6 CRITICAL	S	N	L	N	R	C	H	H	H	Wi-Fi
Arlo Wireless Security Camera	V-74 V-75 V-76	9.8 CRITICAL	S	N	L	N	N	U	H	H	H	Wi-Fi
	V-77	8.1 HIGH	S	N	H	N	N	U	H	H	H	Wi-Fi
Arlo Camera VMC3040, VMC3040S	V-78	6.8 MEDIUM	S	A	H	N	N	U	H	H	N	Wi-Fi
and ABC1000	V-79 V-80 V-81 V-82	5.3 MEDIUM	S	A	H	N	N	U	N	H	N	Wi-Fi
	V-83	8.1 HIGH	S	A	L	N	N	U	H	H	N	Wi-Fi
	V-84 V-85	6.8 MEDIUM	S	A	H	N	N	U	H	H	N	Wi-Fi
	V-86 V-87	5.3 MEDIUM	S	A	H	N	N	U	N	H	N	Wi-Fi
Trifo Ironpie M6 Smart Vacuum	V-88	7.4 HIGH	S	N	H	N	N	U	H	N	H	Wi-Fi
Dongguan Diqee Camera Robotic	V-89	7.5 HIGH	S	N	H	L	N	U	H	H	H	Wi-Fi
Vacuum Cleaner	V-90	7.8 HIGH	S	L	L	L	N	U	H	H	H	Wi-Fi
Samsung's Smart TV	V-91	5.3 MEDIUM	S	N	H	N	R	U	H	N	N	Wi-Fi
Vera Smart Home Controller	V-92	8.1 HIGH	S	N	L	L	N	U	H	N	N	Wi-Fi
Wink	V-93	9.8 CRITICAL	S	N	L	N	N	U	H	H	H	Wi-Fi
Amazon Kindle 8th Generation E-Reader	V-94	6.8 MEDIUM	S	A	H	N	N	U	H	N	N	Wi-Fi
	V-26	5.3 MEDIUM	S	A	H	N	N	U	N	H	N	Wi-Fi
	V-102	8.6 HIGH	X	X	X	X	X	X	X	X	X	X
Blipcare Blood Pressure Monitor	V-95	5.9 MEDIUM	S	N	H	N	N	U	H	N	N	BLE
	V-96	6.5 MEDIUM	S	A	L	N	N	U	N	N	H	BLE
	V-97	7.1 HIGH	S	A	L	N	N	U	H	L	N	BLE
FitBit Charge 2	V-98	6.5 MEDIUM	S	A	L	N	N	U	H	N	N	BLE
PSI GridConnect GmbH	V-99	8.8 HIGH	S	N	L	L	N	U	H	H	H	Wi-Fi
Impinj Speedway Connect R420	V-100	5.4 MEDIUM	S	N	L	L	R	C	L	L	N	Zigbee
RFID Reader	V-101	4.3 MEDIUM	S	N	L	N	R	U	N	L	N	Zigbee
Arduino	V-102	6.5 MEDIUM	H	A	L	N	N	U	N	H	N	Wired

The properties of each vulnerability are their type, attack vector(Vec.), attack complexity (Comp.), the privilege required (Pri.), user interaction (UserInt.), Scope, confidentiality (Conf.), integrity (Int.), availability (Avail.), and communication type (Comm.)

5.1 Device Type Detection

One of the fundamental problems in IoT profiling is device type detection, where the type of IoT device connected to the network is identified. For example, the model would determine that type X of vendor Y's IoT device is connected to the underlying network. Identifying the type of IoT device is very valuable in managing and applying security and privacy mechanisms, e.g., a camera should be treated differently than a fitness tracker, utilizing different security profiles for monitoring and securing the respective devices.

Bruhadeshwar et al. [25] proposed a fingerprinting based framework to recognize the device type using the network activity of devices such as protocols used, set of observed commands, response sequence. Their

fingerprinting approach generates a behavioural profile of device type. It collects incoming and outgoing network traffic, extracts features of interest using statistical tools, aggregates the features, and considers it a reference for device type identification. This approach allows monitoring device behavior throughout its lifetime.

In [95], authors proposed a mechanism named IOT SENTINEL, which is able to recognize the types of IoT devices linked to the network automatically and create rules to prevent communications from compromised devices in order to reduce the harm caused by their compromise. This is achieved by monitoring the traffic flow of compromised devices using an SDN system to secure other network devices and avoid data leakage. Most IoT users do not have the opportunity or desire to upgrade their IoT system software. Users may also forget to disconnect devices no longer in use in larger IoT networks. In this case, the security gateway is an SDN-based traffic monitoring unit operating as a gateway router. Devices are connected either by Wi-Fi or Ethernet links to the security gateway, where all devices' actions are monitored and profiled by generating a specific fingerprint for each device. Then, the devices' fingerprints are sent to the IoT security service unit to identify and evaluate them. This unit returns an isolation level based on this analysis to be imposed on the system by the security gateway. The benefit of their approach is that it contains a set of trained classifiers that each of them had been trained to detect a specific type of device. Once a new device is introduced to the network, a new classifier is trained without changing existing classifiers, preventing an expensive relearning process.

Their dataset of 540 fingerprints was collected from 27 IoT devices in the smart home environment, extracting 23 features from the traffic flow. They have used nine machine learning classifiers, where RF performed best with over 95% accuracy. However, consider devices utilizing other communication channels, such as Bluetooth or LTE. In these cases, the security gateway cannot recognize it, and their method cannot identify devices from the same vendor.

In [91], they utilize different ML algorithms for network traffic data. Nine different IoT devices were utilized for gather and labeling network traffic data. A multi-stage meta classifier is employed by using supervised learning implemented on a stream of sessions emanating from a particular device (i.e., a specific IP address) to decide if the traffic relates to a smartphone, a PC, or a recognized IoT device. The multi-stage meta classifier discriminates between non-IoT and IoT generated traffic in the first stage. Next, the classifier detects device type by associating a specific class to each particular device type. The accuracy of the proposed model is 99.281.%. To the best of their knowledge at the time, they were the first to implement machine learning on network traffic of IoT devices. The result of their work effectively proved that IoT devices can be classified by the characteristics of their generated network traffic.

However, any given IoT device can have varying network traffic in response to user interactions, making it difficult to classify the traffic into a fixed pattern. There is a need for automatic cross-device classification in a real-world setting, as training the model every time a new device is connected is time-consuming and challenging. To supplement this need, [16] proposes an automated classification method to classify unseen and new devices through certain network traffic attributes produced by IoT devices. Their main contributions are introducing a centralized structure for automatic IoT device analysis, as well as producing a method for deriving invariant dependencies within the devices. To do this, the authors propose an LSTM-CNN cascade model to organize the devices by obtaining their traffic features. To classify unknown IoT devices automatically as per their functions into categories, they used the network traffic time-dependencies. Their methodology is comprised of three main components.

- pre-processing for network traffic: when an IoT device joins the network, it automatically create in-and-out traffic based on specific application settings. Most of the generated traffic utilizes TCP/IP protocols. This traffic is classified as time-series data, including valuable knowledge regarding user devices, network status, and habits. This traffic can be captured using Wireshark or Tcpdump.

- Segmentation and feature extraction: Since the IoT device produces a considerable traffic volume, segmentation is necessary. They have segmented the traffic flow into sub-traffic flows of a fixed period T (5 minutes). A considerable amount of features can be obtained from the different perspectives of the segmented sub-flows. They divided the selected features into:
  - Packet quantity features: Number of total packets, number of user-related- packets (TCP, UDP, HTTP), number of access-control-packets (ICMP, ARP, DNS, NTP), and number of packets for protocols such as DNS.
  - Packet length Statistic features: Determine different information such as mean, maximum, sum, minimum, and variance. These features can create an influential feature from IoT traffic flow considering the vast number of packets created and transmitted by IoT devices over the Internet.
  - Protocol related features: The count of different types of protocol packets contained within a segment.
- Device type classification: As previously stated, they use the cascading LSTM-CNN model. The input is fed into the LSTM layers, and the results are in the shape of vectors. Their method joins the vectors and creates a 2D vector, which is supplied to the convolutional layer. The output of the convolutional layer is supplied into the max-pooling layer immediately. After max-pooling, data is reshaped to a vector, six of the most distinctive features were chosen: the length of packets, the number of packets, average length of the packets, maximum number of the packets, control packet numbers, and the peak and average of the control packets. Their best accuracy is 80.1% and average accuracy of 74.8% after repeating the experiment 5 times. They have also compared different classification techniques. The dataset they have used is from the [123] with 15 devices belong to 4 categories.

Authors in [124] proposed a specific method for analyzing the network traffic produced by IoT devices to distinguish their type and behaviour characteristics. The authors collected IoT traffic traces from their experimental smart campus, including over 20 IoT devices like lights, cameras, appliances, and health monitors for three weeks. The captured traffic is analyzed, and analytical characteristics such as burstiness, different data rates, and signalling patterns build a machine learning classifier that could distinguish between IoT and non-IoT traffic and detect individual IoT devices with more than 95% accuracy.

Traffic corresponding to different IoT devices are distinguished in signalling overheads (DNS, NTP, and broadcast) and activity patterns (burstiness, traffic rate, and idle duration). The classification techniques learn the behaviour of the devices and can classify them based on their dedicated profile. Tcpdump tool running on OpenWrt [**?**] was used to collect the LAN side traffic. They have used the K-means algorithm to examine the importance of different attributes via visualization. The RF algorithm with 10-fold cross-validation was applied to achieve more than 97% accuracy in identifying IoT devices and accuracy of over 95% for other independent test results. Therefore with a high probability, the algorithm can uniquely identify an IoT device.

The classifier is trained with the data collected for two weeks and tested with the third week's data. Each training instance has the following attributes: active volume, active time, sleep time, mean rate, packet size average, number of protocols, amount of servers, DNS interval, unique DNS requests, NTP period, label, and most common port to identify the specific device.

In [141], a measurement framework has been developed to capture and characterize network traffic produced by IoT devices in an edge network from IP-spatial, temporal, and service dimensions, resulting in a multidimensional activity profile. IoT devices' behavioural profile is based on a broad range of their traffic characteristics based on three edge networks' dimensions. The IP-spatial dimension analyzes host IP addresses, such as DNS and NTP servers, then try to determine which IoT devices have communicated with each other. In the temporal dimension, three unique traffic patterns of the connected objects at the edge network are identified. IoT devices usually interact using a limited set of typical HTTP, NTP, and DNS applications because of their unique functionalities in the cloud dimension. The framework investigates multi-dimensional behavioural fingerprints for IoT device

groups, network security monitoring, and anomaly traffic detection for many resource constraints of IoT devices on the internet. It can identify unique and undiscovered IoT devices based on the created profiles. Furthermore, programmable edge routers have continuously collected incoming and outgoing traffic of IoT devices in distributed networks. Each flow record collects source and destination IP addresses, source, destination port numbers, protocol, start and end timestamps of the session, byte count, and packet count.

5.2 Device Instance Detection

In device instance detection, specific instances of IoT devices connected to the network are being identified. That means instances A and B of IoT devices of type X of vendor Y are identified, e.g., two different instances of August Doorbell Cam connected to the network can be differentiated. Subsequently, unique security profiles should screen their activity. Identifying IoT devices' instance is extremely valuable in managing and applying security and privacy mechanisms to a specific device rather than all devices of that type.

In [49], authors examined a specific IoT device type recognition problem by investigating a series of time-stamped packets from high-level network traffic. They have created a fingerprint profile for each white-listed device using flow-based features by utilizing supervised machine learning methods. They have enforced some rules for restraining the IoT device communication by the assigned privileges. In the next step, an individual instance of the device with the same vendor, model, and unknown device types is recognized. Features are extracted from a sequence of 20-21 network packets header and payload, which act as the main structure to observe the device behaviour while connected to the Internet. They have created a fingerprint profile for every device. The profiles are consist of 67 distinct features such as TCP payload data offset, inter-arrival-time, ten highest magnitudes of fast Fourier transform, packet rate, and TCP window size. They have used nine machine learning classifiers, in which random forest (RF) with 100 estimators (RFC100) outperforms the other techniques with 90.3% accuracy. However, similar to [95], if the device is connected to the network through other communication technologies like Bluetooth, Zigbee, or LTE, their model will not work.

In [123], authors develop a structure for IoT device categorization, utilizing network traffic. The primary objective is to create a machine learning framework to recognize and classify IoT devices' baseline behaviour based on different network characteristics. They have instrumented a smart environment consist of twenty-eight IoT devices, including motion sensors, digital cameras, plugs, smart lights, and healthcare-related monitoring devices. Next, they have captured the traffic trails for six months and publicly released a subset of this traffic for the research community. They then calculated some statistical attributes such as port numbers and activity cycles to present an insight into the non-IoT and IoT generated network traffic. Their work resulted in the development of a multi-criteria machine learning-based classification method to identify the specific type of IoT devices based on their traffic behaviour with over 99% precision.

They have defined the activity pattern of IoT devices based on the properties of their network activity. Four key attributes are taken into consideration: device sleep time (i.e., The period during which there is no active flow of the IoT device), flow volume (i.e., the sum total of download and upload bytes), flow duration (i.e., the time between the first and the last packet in a flow) and average flow rate (i.e., flow volume divided by the flow duration). The traffic trace is collected for a period of 26 weeks. A small amount of data per flow seems to be shared by each IoT device. In the application layer protocol:

(1) Port numbers: IoT devices communicate with a small set of ports, although non-IoT devices use a much wider variety of services hence communicate with a big list of ports, e.g. 1000 port numbers. IoT devices from a certain manufacturer share certain ports.
(2) DNS queries: IoT devices can be easily distinguished by the domain names they communicate and how often this communication happens. Prominent domain names are shared between all IoT devices. At the same time, non-IoT devices communicate with a large number of domains.

(3) NTP queries: precise and verifiable timing is important for IoT operation hence most IoT devices use NTP protocol.
(4) Cipher suites: numerous IoT devices use TLS/SSL protocol by negotiating a security algorithm with the servers by sending a "Client Hello" packet with a list of "Cipher suites". This cipher suits signals deliver a unique signature for each IoT device.

They have used the following eight key attributes: device sleep time (Single-valued), flow duration, flow volume, average flow rate, NTP queries, DNS queries, cipher suites (Nominal attributes multi-valued) and server port numbers. The three multi-valued attributes are fed into stage-0 Naive Bayes multinomial classifier. This classifier's output, along with the five single-valued attributes, is fed into stage-1 RF classifier. The overall performance reported here is 99.8%.

Lastly, the authors have also discussed the advantage and disadvantages of choosing between speed, performance, and cost in implementing the model in real-time. For instance, real-time extraction of features such as domain names or cipher suite strings requires packet inspection. This incurs a high extraction cost and is detrimental to computing speed, but significantly benefits their precision. In contrast, utilizing low-cost extraction features is beneficial to speed, but lowers their accuracy to 97.85%. Their model's limitation is they have used a multi-stage classifier that could degrade the performance in a practical environment, but with some improvements, this framework has the potential to identify anomalous system activity.

5.3 Unseen Device Detection

Unseen device detection methods help discover the new devices connected to the organization's network that were not included in the training data. In [26], the authors suggested an automated profiling method focused on a semantic interpretation of technical information that does not require physical access to the device. It also preserves the privacy-sensitive features of current and future IoT devices. To extract the semantic features, they have used direct and indirect resources (mobile application of IoT device, certification registries, vendor documentation, user manuals, online information firmware updates) to extract information about the IoT device. Once the information is extracted from those sources, they use related criteria sets to recognize and reveal the security-sensitive capabilities of IoT devices and profile their functionality. Their method consists of two stages, defined as the Discovery Stage and the Profiling Stage. First, they automate the collection of various information sources in the Discovery Stage, then build a criteria set in the Profiling Stage. The criteria set is built from information incuding, but not limited to, feature changelogs, firmware updates, and companion application information. The authors provide two case studies, one on the Garmin Forerunner 230, and one on the Amazon Echo Dot V2.

In [131], a decentralized approach to fingerprinting, i.e., DEFT, is defined, which is different from the current approaches. In the ISP network, the DEFT control logic exists and manages a series of IoT device fingerprinting gateways. The controller performs the training and provides models to the gateways; first, the session data is collected and the potential features are extracted at the gateways, then the provided models are applied to separate the traffic of the session at some specific time interval. The low probability sessions are used for retraining and error correction. In their proposed framework, a total of 111 features are clustered to generate a fingerprint for a category of IoT devices and detect unknown devices. They have achieved 70% accuracy for unknown devices and 97% for known devices.

5.4 Anomaly Detection

Potential anomalous attacks of the IoT devices can be detected using IoT profiling; the moment a specific device violates the baseline profile, it is quarantined for further monitoring. In [113], the authors have detected variations from the valid communication performance, especially on the physical layer, by monitoring and profiling radio

signal strength indication (RSSI) and affiliated wireless transmission of connected objects. They concentrate on identifying attacks possible in smart homes, with the purpose of determining device mode in the following situations: unused wireless technology, anomalous location, an unusual usage pattern, and the unusual period of the day. A Neural Network Machine Learning algorithm is used to differentiate between valid and suspicious communication. However, this approach and the other approaches that aim to profile devices based on physical features suffer from environmental noise.

By analyzing the connections created by a device, a user profiling technique has been proposed in [131]. A behavioural profile is developed to fingerprint the device. N connections are considered for feature generation, which share the same destination, same source, and the same service. They gather statistical features for all connections like destination and source IP addresses, destination and source ports, connection duration, the total number of IP addresses, and so on. They extract counters for counting the instances of used protocols such as TCP, IP, UDP, NTP, ARP, ICMP, LLC, EAPoL, HTTP, HTTPS, FTP, DNS, DHCP. They have also extracted protocol fields such as SYN, Router Alert, Padding, Urgent, REJ, etc. For different types of attacks, an accuracy of 74 to 99% is achieved using a c-means clustering algorithm for data categorization and detecting an anomaly. The anomalous activity can be detected either by IoT device traffic monitoring or by authentication.

5.4.1 Traffic Monitoring. Traffic monitoring is another advantage obtained using profiling. In [120], the authors propose IoTScanner, which passively tracks the network traffic at layer two (link layer) and analyzes this traffic during special observation time windows using frame header information. Their primary purpose is to differentiate between active IP cameras and other non-camera equipment as per discerned traffic patterns within the time window of captured traffic. A downside of this method is that two types of devices can be classified as two distinct types of devices due to the traffic-produced changes through the traffic acquisition time period. For network visualization, this approach is helpful at a high level, but doing this analysis routinely can be tedious.

5.4.2 Authentication. Two adversarial threats are considered in [118]: (a) adversaries that can mimic addresses of the device, encryption keys, and type of data transmitted, and (b) replicate the object software, such as the object's network address and security keys. They have cloned the device-centric data like signal strength, transmission speed, humidity, temperature, and processing speed. Ambient changes related to the environment surrounding IoT entities are the features that can not be reproduced by attackers, for example, changes in humidity, temperature, wind, or any physical modifications that affect IoT devices. Based on the above observations, the authors design a fingerprinting methodology that utilizes a three-tiered domain's characteristics and models the fingerprint. First, the object level characteristics are clock skew, memory usage, CPU load, and the object's surrounding temperature. Second, signal spectral characteristics, signal strength, and packet arrival times are the features that are received by the monitoring objects. The third tier of characteristics is obtained on the IoT server-side. These characteristics are measured by the traffic properties of objects and the received packet frequency. The authors use these features to construct a training fingerprint for a given entity and to quantify similarity using the Bhattacharya distance [58]. They measured the environment's effect on the fingerprint using a linear noise model that rotates and converts the fingerprint.

Each IoT device's special neighbourhood is considered for calculating the noise metrics, and those objects referenced fingerprint and singular value decomposition are used to solve two transformation metrics. Similarities are computed using reverse transformation to the estimate fingerprint and validated using 25 RFID tags. This approach incorporates both behavioural and non-behavioural profiling as well as environmental impact assessment. While promising, many devices require remote device fingerprinting, which does not allow evaluating features at the object-level as defined in this work. Moreover, this approach puts overhead in the calculation of a sample profile during testing. The fingerprinting tools will need to provide a thorough knowledge of the object's neighbourhood in order for this approach to be effective.

Table 4. Papers Discussed in Section 5, Classified by Detection Type

Category	References
Device Type	[25], [95], [91], [16], [124], [141], [49], [123], [131], [113]
Device Instance	[49], [123]
Unseen Device	[49], [26], [131]
Anomaly	[141], [113], [120], [118], [58]

6 FEATURES EXTRACTION FOR IOT SECURITY PROFILING

Feature extraction is an essential component of any profiling system. In IoT profiling, the IoT devices' behaviour is modelled using efficient features extracted from the device's network traffic. It means that network traffic flowing into and out of the device and extracting effective features are important elements for monitoring the devices' behaviour via profiling. In this section, we propose a list of possible features that are useful to train a machine learning model for different types of detection, such as selecting similar device types, unseen device detection, anomaly detection, etc. A minimal subset of the features that could classify the IoT device with an efficient performance should be selected to track IoT devices connected to our network and observe their operation[117][91][25][49][95]. The number of features should be as minimal as possible, and performance should be high. However, IoT devices are usually very conservative in generating network traffic data; they do not make much traffic. So, features like payload lengths, TCP window size, and entropy are very specific to the type of IoT device and are statistically significant in profiling and generating a unique fingerprinting for each device [25].

Table 5 shows a detailed list of features with descriptions used in IoT profiling related problems. Problem type refers to what specific type of problem can be solved using the respective feature. The level of practicality refers to how the selected feature can contribute to model the IoT profiling. It can be high, medium, or low. The level of efficiency refers to the complexity/efforts required to extract the feature. It could be high, medium, or low, where high means that features could be extracted with less amount of complexity/effort.

We have categorized the features used in IoT profiling related problems into the following categories:

- **Size related features:** present important information about the processing speed and memory capacity of the IoT devices. The practicality is usually high with high efficiency for most of the size-related features. Deciding on which specific size-related features to use for the profiling is usually device-centric and differs as per device nature. e.g. a smart TV can be easily differentiated from a doorbell using size-related features.
- **Service related features:** list important information about what specific services the IoT device can perform. The practicality is usually high with medium efficiency for most of the service-related features, deciding which specific service-related features to use for profiling are also device-centric. e.g. a smart TV and a security camera can be differentiated if we add service-related features to our features vector. We can't surely differentiate them using only size-related features.
- **Time related features:** give important information about the timing and synchronization of the IoT device. Some IoT devices are more active at a specific time and less active at some other time. Time-related features can also hint at the processing speed of the IoT device.
- **Statistical features:** take a leading role in the profiling of IoT devices. Most IoT devices are expected to occasionally send a small amount of data with a short active time and long sleep time. Several packets of varying lengths are included in the segmented sub-traffic flow. Therefore, it is important to investigate these packet lengths' statistics to profile IoT devices into different types. For example, authors in [16] extracted statistics features such as minimum, maximum, sum, mean, variance, standard deviation, kurtosis, and skewness to profile IoT devices. Similarly, authors in [123] proposed a method that utilizes some

other statistical attributes, including port numbers, activity cycles, cipher suites, and signalling patterns, to understand the underlying network traffic properties of IoT devices [123].

- **Communications protocol:** play an important role in identifying and classifying IoT devices. IoT devices are using different communication protocols to communicate with each other. However, some IoT devices are using specialized protocols to communicate. Network traffic frame header information at the link layer is passively observed and analyzed during specific observation time windows[120]. The four well-known protocol layers are used to profile IoT devices [25, 95], namely Application Layer Protocols, Transport Layer Protocols, Network Layer Protocols, and Link Layer Protocols.
- **Wireless Communication Standard/Wireless Transmission Technology:**
  With the growth of wireless network sizes and complexities, an appropriate tool is required to profile IoT devices. Often, vendors of the IoT devices will sell their application gateway with an integrated access point, which can hide the underlying communication protocol to a certain degree from the device owner. Despite the advantages of wireless communication such as accessibility, flexibility, and usability, there are security concerns like privacy and controllability in this kind of environment [121]. Most IoT devices are using Wi-Fi, Zigbee, and Bluetooth wireless communication standards. For example, Internet access can happen via Wi-Fi communication, and personalized health monitoring devices in the healthcare domain can use Bluetooth, electronic gadgets in a smart home (e.i. smart lighting) can use Zigbee communication. The source and destination addresses, frame type and sub-type, SSIDs present are some examples of useful information that relies on wireless communication. Since parsing Zigbee, Bluetooth LE, and Wi-Fi frames are significantly different from each other; it is necessary to be treated on a protocol by protocol basis to extract this relevant information from these frames for targeted analytics [120]. However, there is no straightforward way or any properties to show which protocol is used by a device; understanding the type of protocols used by devices primarily depends on the devices' usage or their energy consumption behaviour.
  - **Wi-Fi-Frame:** It is the predominant mode of communication for IoT devices. Wi-Fi frame features are very efficient in profiling IoT devices. The IoT devices accessing the Internet mainly use Wi-Fi. To capture a Wi-Fi frame, we can use the TP-Link TL-WN722N adapter [**?**].
  - **Bluetooth-Frame:** The presence of Bluetooth is increasing in IoT devices, especially in the healthcare domain; for example, most on-body/wearable IoT devices such as smartwatches and blood pressure monitors utilize Bluetooth to communicate. In this case, we can use Ubertooth One [**?**] as an open-source monitoring tool for capturing Bluetooth frames.
  - **Zigbee-Frame:** Most smart home appliances and gadgets use Zigbee frames to communicate, e.g. Philips Hue lighting system. Hence, Zigbee frame-based features can play an important role in IoT profiling to differentiate and profile those devices from other devices. To capture the Zigbee frame, we can use RZUSBstick [**?**].

7 MACHINE LEARNING METHODS IN IOT PROFILING

The enormous scale and heterogeneous structure of IoT would intensify conventional attacks on the network and open an entry point to the organization's network. Machine learning has been extensively used for the classification of traffic and detection of intrusion. Researchers [82, 136] have used machine learning to classify the network traffic to identify the services used on the source computers; for example, Sun et al. [130] utilized transfer learning approaches with promising results to classify the network traffic. In addition, many researchers have also used machine learning for detecting malicious and benign traffic and for automatically detecting and identifying device behaviour [5, 27, 135].

Table 5. List of Features with properties used in IoT profiling related problems.

Category		Feature Name	Description	Problem Type	Practical	Efficiency
Size Related Features		IP packet size (length)	Entire packet size in bytes including IP header embeded protocol headers and data. 16-bit field. [25, 49, 91, 95, 124]	DTD, DID	High	High
		TCP Payload size (length)	The carrying capacity of a packet.[25, 49]	DTD, DID	High	Medium
		TCP window size	The maximum number of bytes that can be sent at any point. (Memory capacity and processing speed of IoT device).[25, 49]	DTD, DID	High	Medium
		Traffic Volume	A measure of the total work done by a resource over a period of time.Traffic volume = Traffic intensity × time.[123]	DID	Medium	Low
		Traffic rate	Amount of data moving across a network at a given point of time.[123]	DID	Medium	Medium
		Payload Entropy	Entropy is a measure of disorder. A larger entropy value shows more variety of data sizes being transferred.[25]	DTD	Medium	Medium
		Ethernet Frame size	The frame size of a standard Ethernet frame is the sum of the Ethernet header, the payload, and the frame check sequence (FCS) field.	DTD	Medium	Medium
		IP Header size	Size of IPV4 header (IHL).[49]	DID	High	High
		TCP Payload data offset	Data offset gives data start information to the upper network layers.[49]	DID	Medium	High
		Number of IP destinations	Number of destination IP addresses accessed by the IoT device [49, 95]	DTD, DID	High	Medium
		Packet Rate	Number of incoming and outgoing packets per second.[49]	DID	High	Medium
		Flow Volume	Sum total of download and upload bytes.[123]	DID	High	Medium
		Active Volume	The time slot when IoT session is active (IoT activity lifespan) [124]	DTD	High	Medium
		Number of servers	Number of servers accessed by the IoT device.[124]	DTD	High	Medium
		Protocol count	Number of protocols used by the device.[16, 124, 131]	DTD, UDD, AD	High	Medium
		Byte Count	Number of Bytes sent and received by the device.[16]	DTD	High	Medium
		Packet Count (DNS, ARP, NTP)	Number of DNS, ARP, NTP packets sent and received by the device.[16]	DTD	High	Medium
		Number of total Packets	Total number of packets.[16]	DTD	Medium	Medium
		Total number of IP address	Total number of IP addresses accesed by thedevice.[131]	UDD, AD	High	Medium
Service Related Features		Source IP address	Source IP address of the IoT Device. [91, 141]	DTD	High	Low
		Destination IP address	Destination IP address of the IoT Device. [91, 131, 141]	DTD, UDD, AD	High	Low
		Protocol (L3, L4, L7)	Protcols used in layerr 3, 4 and 5.[25, 91, 95, 123, 141]	DTD	High	Low
		Source Port	Source Port of the device.[49, 91, 95, 131, 141]	DTD, DID, UDD, AD	High	Low
		Destination Port	Destination Port of the device.[49, 91, 95, 131, 141]	DTD, DID, UDD	High	Low
		DNS Queries(Request)	A DNS Query is a request for information sent from a DNS Client to a DNS Server. [123, 124]	DTD	High	Medium
		NTP Queries	NTP communication consists of time requests and control queries. Time requests provide the standard client/server relationship in which a client requests time synchronization from an NTP server.[123]	DID	High	Medium
		Router Alert	Used to inform a router that the current IP packet has some peculiarities that should be studied before it is forwarded on. [25, 95]	DTD	Low	High
		Raw Data	Data that has not been processed for use [95]	DTD	Low	Low
		Cipher Suites	A cipher suite is a set of algorithms that help secure a network connection that uses transport layer security (TLS) or secure socket layer (SSL).[123]	DID	High	Low
		Transmission speed	The rate at which data are moved across a communications channel [118]	A	Medium	Medium
		Signal Strength	Transmitter power output of IoT device as received by a reference antenna at a distance from the transmitting antenna.[118]	A	Low	Low
		processing speed	Processing speed is a measure of the time required to respond to and/or process information by the IoT device.[118]	A	Medium	Low
		Operating temperature	The temperature under which IoT device can operate normally.[118]	A	Low	Low
		humidity	The humidity under which IoT device can operate normally.[118]	A	Low	Low
		Received Packets	The number of packets received by IoT device. [16]	DTD	Medium	Medium
		Server Port Numbers	The sever port number being access by IoT device [123]	DID	High	Medium
		Transmited Packets	The number of packets sent by IoT device.[16]	DTD	Medium	Medium
Time Related Features		Flow duration	Time between the first and last packet in a flow [123]	DID	High	Medium
		TTL for TCP and UDP packets	time to live (TTL) or hop limit is the lifespan or lifetime of a packet. [49]	DID	Medium	High
		Inter arrival time	The time between the start of two packets.[124]	DTD	High	Medium
		Device Sleep Time	Time over which IoT device has no active flow [123, 124]	DTD	High	Medium
		Active time	The amount of traffic being exchanged by IoT session.[124]	DTD	High	Medium
		DNS interval	Time slot between two DNS Queries.[124]	DTD	High	Medium
		NTP interval	Time slot between two NTP Queries.[124]	DTD	High	Medium
		Session start timestamp	Starting time of the session.[141]	DTD	High	Medium
		Session end timestamp	Finishing time of the session [141]	DTD	Low	Medium
Wireless Communication Protocol Features	Wi-Fi	Type	Indicates whether the frame is a Management, Control or Data frame (Two bit field).[120]	TM	Low	High
		Sub-Type	Describe the detail of the frame type (4 bit field) [120]	TM	High	High
		Length	Length of the frame [120]	TM	High	Medium
		MAC Address	MAC address of the device generated the frame [120]	TM	High	Low
		SSID	The number of access points seen in the environment.[120]	TM	High	Medium
	Bluetooth	Type	Type of packet. There are 12 types of packets for each SCO and ACL physical links, and four types of common control packets for both.[120]	TM	High	High
		Length	Length of the frame [120]	TM	High	Medium
		MAC Address Type (Random)	BLE adds the ability to periodically change the address it could static and private [120]	TM	High	Medium
		MAC Address Type (Public)	This is the standard, IEEE-assigned 48-bit universal LAN MAC address which must be obtained from the IEEE Registration Authority [120]	TM	High	Medium
		MAC Address	A Bluetooth address sometimes referred to as a Bluetooth MAC address, is a 48-bit value that uniquely identifies a Bluetooth device [120]	TM	High	High
		Node Local Name	Local Node for the Bluetooth .[120]	TM	High	Medium
	Zigbee	Type	Type of the frame ,[120]	TM	High	High
		Length	Length of the frame .[120]	TM	High	Medium
		PAN ID	Unique ID for Zigbee frame .[120]	TM	High	Medium
		Addresses	Address for the Zigbee frame.[120]	TM	High	Low
Communication Protocol Features	Link	ARP	The address resolution protocol (arp) is used by the Internet Protocol (IP) to map IP network addresses to the hardware addresses.[95]	DTD	Medium	High
		LLC	logical link control (LLC) sub-layer deals with addressing and multiplexing. [95]	DTD	Medium	High
	Network	IP	Internet Protocol is the principal communications protocol for transmitting datagrams across the network.[25, 95]	DTD	High	High
		ICMP	The Internet Control Message Protocol used to diagnose network communication issues.[25, 95]	DTD	High	High
		ICMPv6	ICMP for IPv6 [25, 95]	DTD	Medium	High
		EAPoL	extensible authentication protocol (EAP) over LAN is a network port authentication protocol.[25, 95]	DTD	Low	High
	Transport	TCP	The transmission control protocol is used to connect network devices to the Internet (Connection-Oriented).	DTD	High	High
		UDP	User Datagram Protocol is used for establishing low-latency and loss-tolerating connections (Connection less).	DTD	High	High
	Application	HTTP	Hypertext transfer protocol is used to deliver data on the world wide web (WWW).	DTD	High	High
		HTTPS	Secure version of HTTP.[25, 95]	DTD	High	High
		DHCP	Dynamic Host Configuration Protocol used to automate the process of configuring devices on IP networks (Dynamic configuration).[25, 95]	DTD	Medium	High
		BOOTP	The Bootstrap Protocol is used to automatically assign an IP address to network devices from a configuration server (Static configuration). [25, 95]	DTD	Low	Medium
		SSDP	The Simple Service Discovery Protocol used for advertisement and discovery of network services.[25, 95]	DTD	Low	Medium
		DNS	The domain name services translate domain names into IP Addresses.[25, 95]	DTD	Medium	High
		MDNS	Multicast DNS protocol resolves hostnames to IP addresses within small networks that do not include a local name server.[25, 95]	DTD	Low	High
		NTP	The Network Time Protocol used for clock synchronization [25, 95]	DTD	Medium	High
Statistical Features		Flow size	A sequence of packets from a source to a destination	DTD	Medium	High
		Minimum	Minimum packet in a flow [49]	DID	High	Low
		Maximum	Maximum packet in a flow [49]	DID	Medium	Low
		Standard deviation	How spread out the packets are in a flow [16]	DTD	Medium	Low
		Sum	Aggregate of the data [16]	DTD	Medium	Low
		first quartile	The middle number that falls between the smallest value of the data set and the median.[49]	DID	Medium	Medium
		Median	The median of dataset [49]	DID	Medium	Low
		third quartile	The central point that lies between the median and the highest number of distribution.[49]	DID	Medium	Medium
		Mean	The average of the numbers.[49]	DID	Medium	High
		Variance	The measurement of the spread between numbers in a data set [49]	DID	Medium	Medium
		Skewness	Skewness is the degree of distortion from the symmetrical bell curve in a probability distribution.[16]	DTD	Medium	low
		Kurtosis	Kurtosis is a measure of whether the data are heavy-tailed or light-tailed relative to a normal distribution. [16]	DTD	Medium	Low
		inter-quartile	the amount of spread in the middle 50% of a dataset.[49]	DID	Medium	Low
		fast Fourier transform (FFT)	A mathematical method for transforming a function of time into a function of frequency (from the time domain to the frequency domain).[49]	DID	Medium	Low
		User Packet #. (TCP, UDP, HTTP)	Include user data and device server communication packets.[16]	DTD	Medium	Low
		User Packet Length Avg.	Average length of the use packet [16]	DTD	Medium	Low
		User Packet Length Peak	Maximum value for the user packet length [16]	DTD	Medium	Medium
		Control Packet #. (ICMP, ARP, DNS, NTP)	Are supporting functional protocol packets such as ICMP, ARP, DNS, and NTP packets. [16]	DTD	Medium	Medium
		Control Packet Avg.	Average of the control packet [16]	DTD	Medium	Medium
		Control Packet Peak	Maximum value of the control packet [16]	DTD	Medium	Medium
		Average Flow Rate	Flow volume divided by Flow duration.[123]	DID	High	Medium
		Interval between flow	the time window between two flows.[124]	DTD	Medium	Medium
		Average Packet size	The average size of the package of an IoT session.[124]	DTD	High	Medium
		Mean rate	The average rate .[124]	DTD	Low	Medium
		Peak to mean ratio	The ratio of average and peak .[124]	DTD	High	Low
		Most frequent port number	Port number that is most frequently accessed by the IoT device.[124]	DTD	High	Low

Problem type can be device type detection (DTD), device instance detection(DID), unseen device detection (UDD), anomaly detection (AD), traffic monitoring (TM), authentication (A).

Table 6. Overview of Previous Machine Learning approaches

Reference	ML Classifier	Behaviour Based (B) / Statistical Based (S)	Device Type (T) / Unique Instance (U)	Header Information (H) / Flow Information (F)	Prediction Accuracy (%)	Number of Devices in Dataset
[25]	GB	B	T	H	86.00 - 99.00	14
[49]	RFC (100)	S	U	H+F	90.00	27
[117]	RF	B	T	H	99.90	4
[95]	RF	B	T	H	95.00	27
[91]	Multiple	S	T	F	99.28	9
[16]	LSTM + CNN	S	T	F	74.80 - 80.01	15
[123]	NBMC + RF	S	U	F	99.80	28
[124]	RF	B + S	T	F	97.00	21
[28]	LSIF	B	T	F	90.00	111
[118]	Transfer Learning	B + S	T	Environment that surrounds IoT	90.00	25 RFID Tags
[12]	DT	B + S	T	H + Payload	97.00	6
[114]	RF	B + S	T	H + Payload	97.00	7
[66]	NN	B + S	U	H + Payload	99.00	10
[82]	RNN + CNN	B	T	H + Payload	98.00	100
[56]	CNN + RNN + DNN	B + S	T	H	99.00	6

Since the profiling of IoT devices and IoT device identification is a recent topic, many researchers have used machine learning techniques for the profiling of IoT devices, and IoT device identification in the network [25, 49, 91, 95, 117, 123, 124, 141]. Using machine learning for profiling and identifying IoT devices has several advantages in terms of performance and applicability. However, these approaches require feature engineering methods like feature extraction, feature selection, and feature tunning. It could sometimes be expensive since the latest methods need several sessions to detect unwanted IoT devices and traffic. They also require a multi-stage profiling model. It also requires large training data for training the classifier. One approach may be to recognize authorized and unwanted IoT devices using a single session and a solution free of feature engineering overhead and glitches that can be introduced during feature engineering.

Beside pure machine learning-based solutions, many approaches have combined statistical feature analysis with behaviour features and feed the feature vector to the machine learning to identify IoT devices [12, 56, 124]. However, there are also drawbacks to statistical and behavioural features, including defining an effective feature list and pre-processing them to feed into ML models, all of which entail considerable domain awareness. Table 6 shows an overview of previous research for profiling the IoT devices using machine learning techniques with correspondence performance.

Experimental analysis by most research shows that the RF machine learning algorithm performs better than the other algorithms in IoT device identification. The RF proposed by [24] is a classifier composed of several classification trees. Each time, the RF uses a uniform probability to choose the possible splitting variables. They compared hundreds of classifiers on 121 datasets in this analysis [134] and concluded that classifiers based on the RF and the SVM are the best in terms of network traffic classifications.

7.1 IoT Data Characteristics

IoT devices connect uniquely with a few server ports, and devices from the same provider also use these ports. While in the case of non-IoT devices, a much wider range of services is used by the device (e.g. streaming content on YouTube, browsing a website, etc.). Furthermore, IoT devices usually exchange a minimal volume of data per flow compared to non-IoT devices, with the majority of their communication occurring in short bursts during user interaction. Otherwise, they periodically generate a small amount of activity (e.g. DNS, NTP). Overall, their contribution to a network's traffic volume is relatively low in comparison to non-IoT.

As IoT devices are primarily designed for particular uses, the device accesses a limited number of domains corresponding to their vendor-specific end-point servers. IoT systems can subsequently be distinguished equally from the domain names they communicate with and how often they access a DNS protocol. Every IoT device

has a different pattern in the frequency of accessing a DNS protocol. It might be easy to differentiate between the generated traffic by IoT or non-IoT devices by comparing all those characteristics. Therefore, identifying an IoT device from a non-IoT device could be easily achieved by taking the specialized IoT features and behaviours into consideration. However, detecting and identifying a specific IoT device from another IoT device would be challenging due to heterogeneity.

7.2 Statistical Machine Learning Approaches

7.2.1 Supervised learning. A near-real-time classification approach is proposed by [12] to classify device types using supervised machine learning algorithms based on network features derived both from traffic flow characteristics and the payloads of the packets. Their approach automatically detects a newly connected device on the home network and uses a representative and heterogeneous collection of actual IoT devices to test their method's efficiency. Results indicate autonomous identification, based on decision tree models using the one-vs.-all methods, with 97% average accuracy.

Bruhadeshwar et al. [25] used a supervised fingerprinting based framework to recognize the device type using the network activity of devices such as protocols used, set of observed commands, response sequence. Their fingerprinting approach generates a behavioural profile of device type by collecting the network traffic in and out of device extract features of interest using statistical tools, aggregate the features and use gradient boost supervised learning algorithm to identify the devices with 86-99%. The experimental analysis dataset includes 14 devices, and they have used 20 features.

The authors of [49] have constructed a fingerprint for each white-listed device using flow-based features by studying a sequence of time-stamped packets from high-level network traffic. Features are extracted from a sequence of 20-21 network packets header and payload, which act as a baseline to continuously monitor the device's behaviour while connected to the internet. They have calculated summary statistics (minimum, maximum, first quartile, median, third quartile, mean, variance, inter-quartile) and fast Fourier transform for some used features. They have used 67 features to train a supervised machine learning algorithm. They have compared the different algorithms' performance and showed that the RF classifier with 100 trees (RFC100) outperforms the other algorithm with 90.03% accuracy. Their dataset includes 27 IoT devices.

In [117], they have used a system based on supervised machine learning to classify the types of network-connected devices by extracting distinguishable features from packet streams sent and received. Their method can identify the traffic generated by different IoT devices in their experimental smart home environment consisting of four different types of IoT devices. Their dataset includes four devices. They have trained six different machine learning algorithms and have shown that the RF classifier has a promising result than others with 99.9% overall accuracy. In [95] they have collected 23 features from 27 different IoT devices and trained supervised machine learning algorithms. They have shown that the RF algorithms can achieve 95% accuracy. Their mechanism can automatically recognize the types of devices linked to an IoT network and allow rules to prevent communications from compromised devices to be applied to reduce the harm caused by their compromise.

Their framework uses a supervised machine learning algorithm for IoT device classification using network traffic [123]. They have instrumented a smart environment consist of 28 IoT devices. They have collected the traffic traces for six months and publicly released a subset of this traffic for the research community. They have calculated some statistical attributes like activity cycles, port numbers, signalling patterns, and cipher suites to present an insight into the network traffic generated by IoT and non-IoT devices. They have developed a multi-stage machine learning-based classification algorithm to identify specific IoT devices based on their network traffic. They have used a combination of naive base multinominal and the RF classifiers, which results in over 99% accuracy.

The authors of [124] have collected IoT traffic traces from their experimental smart campus, including over 20 IoT devices. Their training vector contains 12 features. The traffic traces are analyzed, and statistical attributes such as burstiness and data rates, signalling patterns, activity cycles. These attributes are used to build a supervised machine learning classifier that could distinguish between IoT and non-IoT traffic and Identify individual IoT devices with more than 95% accuracy. They have shown that the RF performs better than other algorithms.

7.2.2 Unsupervised learning. Most of the approaches use a labelled dataset to identify the devices in the network. However, this could potentially question the scalability of the approach. Devices not included in the dataset can not be recognized by their approaches.

In this paper, [88] they present AUDI, a model for identifying IoT devices in the network by considering their periodic communication using an unsupervised machine learning algorithm. Their model can learn without label data and human intervention. They can identify any previously unseen device. The accuracy of their model is 98.2%.

The authors in [101] proposed an algorithm called DIoT, a distributed self-learning approach to identify malicious devices. Their system can detect malicious devices without human intervention and training data by using federated learning. They could also identify new and unknown attacks. Their experiment consists of 31 IoT devices with a 95.6% detection rate with minimal false alarms.

7.2.3 Semi-Supervised learning. Since the collection of label data is a challenge and time-consuming, [**?**] suggest IoT device profiling and identification based on a semi-supervised machine learning algorithm. Their model can identify IoT and non-IoT devices with 99% accuracy by using 5% label data. They have used a convolutional neural network and multi-task learning. They have also suggested features that could differentiate the most among the devices.

7.3 Deep Learning Approaches

Deep learning is a promising way to acquire various IoT devices' characteristics by learning from their RF data. Three distinct deep learning models are considered by [56] to classify wireless devices and differentiate between wireless devices with the same manufacture. These three models are deep neural network (DNN), convolutional neural network (CNN) and recurrent neural network (RNN). For the experimental reason, a large data collection of RF traces is obtained using a USRP-based testbed from six 'identical' Zigbee devices. Experimental findings indicate high accuracy of deep learning methods for detecting wireless devices that could possibly strengthen IoT security.

[16] proposes an automated classification method for IoT devices to classify new and unseen devices through the network traffic attributes generated by IoT devices. They have cascaded LSTM and CNN models to identify the semantic type of a device automatically. Their dataset includes 15 devices, and they have used six features to train the model. The performance of their model is 74.8-80.01% accuracy. They can also classify the unseen devices which do not exist in the training dataset. Kotak and Elovici used deep learning techniques on the TCP payload of network traffic for IoT device classification, identification, and detection of white-listed IoT devices in the network traffic [66]. A single TCP session is needed to detect the source IoT device. Their approach is free of feature engineering overhead.

Lopez et al. [82] proposed a new classification methodology for network traffic based on a combination of deep learning models that can be used for IoT traffic. They showed that the best detection results are given by a recurrent neural network (RNN) coupled with a convolutional neural network (CNN). The natural domain for CNN, which is image processing, was quickly and automatically applied to NTC. They demonstrated that without needing any feature engineering, as is common when implementing other models, the proposed approach provides better detection results than alternative algorithms. A full analysis is conducted on many CNN and

RNN architectures, including the effect of the selected features and the duration of the training's network flows. Authors in [143] compiled a key set of fingerprinting terminologies along with identification of important features to achieve effective and accurate fingerprinting of IoT devices. The proposed IDWork model is a systematic framework for categorizing IoT devices in fingerprinting mechanisms. It compares and selects suitable machine learning augmented techniques to be applied in IoT fingerprinting mechanisms.

As tens of thousands of tiny LoRa devices are deployed over large geographic areas, a key component to the success of LoRa will be the development of reliable and robust authentication mechanisms. To this end, radio frequency fingerprinting (RFFP) through deep learning has been heralded as an effective zero-power supplement or alternative to energy-hungry cryptography. DeepLoRa [7], proposed a data augmentation technique based on ITU-R channel models, to enhance the reliability and robustness of the RFFP algorithms.

Fingerprinting IoT devices behind a network address translation (NAT) has been investigated in [99]. In this study, the authors explored the capabilities of unsupervised and semi-supervised shallow and deep learning methods to classify IoT devices deployed in a NAT. This work has applied two unsupervised clustering algorithms, i.e., K-Means and DBSCAN. Due to inefficiency in the unsupervised methods, they devised two semi-supervised ML algorithms that encompass a Logistic Regression classifier with either Auto-encoders or Restricted Boltzmann Machine (RBM). Consequently, the results showed that the algorithm that featured auto-encoders has a higher classification rate for non-IoT devices rather than IoT devices. By relying on an explain-ability technique, they found that several features affected these rates. Moreover, the evaluation of the Logistic Regression with RBM features' classifier provided sound fingerprinting of IoT devices behind a NAT with a state-of-the-art accuracy of 98%.

The pre-identification of IoT devices connected to the network can help administrators set related security policies according to the functionality and heterogeneity of the devices. However, the existing methods are based on manually extracted features and prior knowledge to identify the IoT devices, which increases the difficulty of the device identification task and reduces the promptness. Authors in [145], presented CBBI to identify IoT devices. CBBI uses a hybrid neural network model ConvBiLSTM to automatically learn the representative spatial and temporal features from the network traffic, such as the position relationship of the internal organization structure in network communication traffic, the time sequence of the data packets, and the duration of the network flow. Moreover, to resolve the data imbalance in deep learning and improve the accuracy of the model, the data augmentation module FGAN has also been introduced.

The Smart Recon method [132] was introduced to identify known IoT devices with an accuracy of 98% using only a single packet sniffed from a network traffic flow. While a high degree of accuracy has been achieved, effective feature extraction and acceptable computational overhead have also been addressed. The combination of locality-sensitive hashing to create feature vectors from .pcap files and a simple neural network allowed for the training of a classifier that was able to produce a high degree of accuracy with a very small input sample.

To solve the problem of IoT device identification in a low-overhead manner, [34] introduced a lightweight IoT device identification scheme based on feature selection and machine learning algorithms. The authors demonstrated its ability to identify IoT devices with over 99.5% accuracy with less cost than other schemes. Flow-related statistical characteristics and the time interval of feature extraction have been studied. They introduced symmetric uncertainty and correlation coefficient and proposed a novel low-overhead feature selection method to perform feature selection on the extracted flow-related statistical features in IoT device identification, and the valid features were filtered while reducing the dimensionality of the features.

8 CHALLENGES AND FUTURE RESEARCH DIRECTIONS

8.1 Challenges

8.1.1 IoT Profiling Problems. Most of the existing approaches may not be suitable for fingerprinting IoT devices. The feature list considered by any approach is best suited for the devices in their dataset. Taking into account the vast amount of IoT devices, those feature sets will not be able to represent a profile for every IoT device. As a result, the defence framework to monitor devices could block valid IoT devices, leading to the malfunction of the industry's operation and financial loss. With this scalability issue, a future challenge is to select a universal set of features that is able to profile the most IoT devices possible. Another approach that can be considered is to create categorical feature sets for each IoT device type.

Additionally, most of the existing approaches do not have a mechanism for continuously updating the devices' profiles. Ideally, the initially created profile would be effective for the lifetime of the device. However, if the physical location of some IoT devices is changed, or network issues occur, their behaviour can change.

Furthermore, all the approaches use previously collected training traffic data, and generating adversarial examples to manipulate the training data could trick the profiling algorithm. Accuracy and identification rate is another key issue and challenge.

8.1.2 IoT Profiling Feature Extraction. Feature extraction is another challenge. In this study, we have outlined the features used by previous methods in terms of efficiency and practicality. From the feature table 5, we can see that the efficiency (i.e. the complexity of extracting a given feature) is high, which could lead to computational overhead.

Feature engineering is the most challenging task in training for almost any machine learning classifier. Feature pre-processing and extraction tasks for profiling the devices are potential roadblocks in the scalability of a proposed profiling method. Thus, a strong profiling method requires less feature engineering overhead. An open challenge in this domain is developing a profiling approach that could identify the device in the network, while using a minimal feature engineering overhead. Feature engineering could also pave the way for some adversarial attacks on profiling approaches. Some profiling approaches even used a more complex set of features extracted from application-layer response data. A hybrid set of features that could identify most IoT device types, instances, unseen and anomaly detection in the network is required. This hybrid set of features should be extractible with less computation overhead.

8.1.3 Unavailability of Training Datasets. The unavailability of public datasets for IoT profiling is an unavoidable challenge in the IoT environment. To the best of our knowledge, there are only four datasets publicly available for the research community until the time of this research, namely the IoT Sentinel dataset[95][1], UNSW dataset [123][2], LSIF dataset [28][3], and the Information Exposure dataset [111][4].

Most of the previous approaches used a small experimental test board with few devices in their dataset, and their dataset is not publicly available. This makes it difficult to compare the effectiveness of profiling approaches with one another. An open issue is to generate a public, standardized dataset with the maximum number of features possible that could be used as a baseline where different profiling approaches performance can be measured against this dataset. Without an standardized public dataset as a baseline, proposed approaches often report high accuracy for a small, specialized set of IoT devices.

Previously suggested approaches could perform poorly if a specific device from a different domain is included in their experiment. Since the domain of IoT profiling is an emerging and recent topic, there is a need for a

[1]https://research.aalto.fi/en/datasets/iot-devices-captures
[2] https://iotanalytics.unsw.edu.au/iottraces
[3]https://github.com/networks-lab/LSIF
[4]This dataset is available upon request from https://moniotrlab.ccis.neu.edu/imc19dataset/

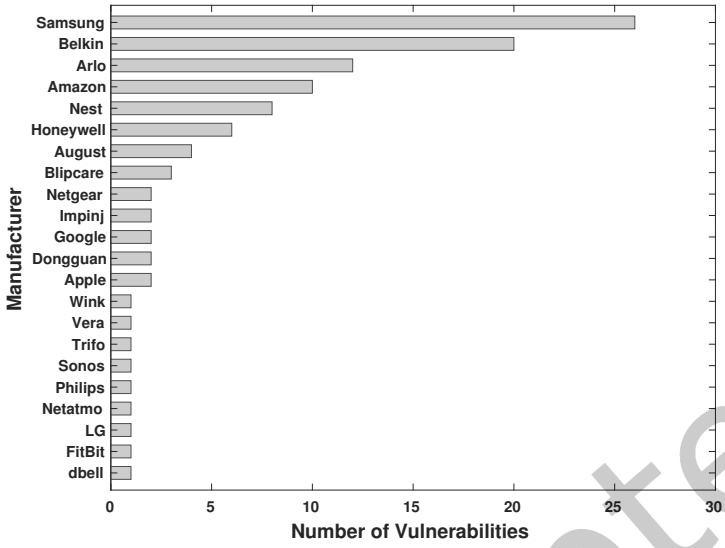

Fig. 8. Vulnerabilities in different manufacturers.

standard dataset that includes enough IoT devices from different domains with different natures and a maximum number of features.

8.1.4 Challenges Related to Common Vulnerabilities. As IoT is a relatively new market, many devices are produced by small vendors. Common vulnerabilities may not be addressed due to lack of resources, or simply because it is more profitable to release a new version of the device with the implemented changes. As a result of these design flaws, a varied range of attacks can be performed as discussed by [48]. In contrast, many larger companies and vendors often address bugs and implement security patches in their devices. Some companies, like Samsung, offer rewards for reporting bugs or security issues, incentivizing community involvement in creating a better product.

There is also a need for a comprehensive database to provide common vulnerabilities for IoT devices in various domains. This database will be used by network administrators and the companies to detect potential security threats in their network. In this research, we have discussed a list of vulnerable devices. Figure 8 shows the number of vulnerabilities in different IoT devices in our research. As shown, Samsung has reported the highest number of vulnerabilities, whereas dbell has the lowest. However, this is not indicative of significant security flaws. Samsung has more devices in the market and any potential vulnerability is discussed by Samsung with corresponding patches publicly. There could be thousands of vulnerabilities with the small vendor IoT devices, but they are not publicly discussed, or not discovered due to a smaller consumer base. For future work, we are planning to enhance the database to include more devices from different domains.

8.1.5 Pre-processing and Feature Extraction. IoT profiling is strongly dependent on feature extraction from network traces generated by the IoT devices. Since a large number of IoT devices of a heterogeneous nature are connected to the network, they produce a volatile traffic pattern. This variant traffic pattern makes it complicated to choose a set of features to identify the devices. Therefore, there is a need to extract a huge number of features to cover the various devices. Because of the devices' heterogeneity, the traffic generated by a specific IoT device is mostly different from the traffic generated by other IoT devices. Choosing which particular set of features should

be used to profile the devices is therefore difficult. Some features may suit a certain set of devices well, whereas those features will contribute much less in profiling a different set of devices. Therefore, there is a need to find a set of strong features in identifying devices with different communication protocols and different nature. This feature extraction could be an expensive process in the vast number of IoT devices.

8.1.6 Learning for IoT profiling. Learning is considered to be another limitation for profiling IoT devices. Profiling is strongly dependent on training data. The training data has to be collected for approximately a week in most cases. This produces a significant time constraint in the real-world scenario. This dependency on the training data is undesirable in larger networks where new devices are continuously connecting to the network. There is a need for a rule-based approach that is not strongly dependent on training data. Another solution could be having already trained models for the vast number of devices. Once a new device connects, it can be matched to the database of devices. This approach may also fail when device vendors produce upgraded versions. Data on a certain device may no longer be relevant due to changes in the new version, making this environment unpredictable. However, as of 2019, manufacturer usage description (MUD) has become a standard for device identification. A promising future for IoT intrusion detection systems utilizing machine learning and MUD is reviewed by [90].

8.1.7 Secure and Privacy Preserving Learning. The network traffic pattern of IoT devices used to create the device profile could carry important information about the activity pattern of the IoT devices and the device user. Due to lower security implementations, this is a valid and significant privacy concern for users. Most IoT device users are not willing to share the network traffic pattern of their devices, as traffic data could contain personally identifiable information. Therefore, there is a need for a profiling technique that preserves the user's privacy. Since many IoT devices do not perform complex tasks, their activity pattern could be easily predicted from their network traffic traces. This prediction of activity patterns creates a significant challenge for the user of the devices and the industry.

8.2 Future Work

8.2.1 Integrating Contextual Information. A combination of contextual information and network traffic, generated by IoT devices, could produce a strong profiling technique that could identify a huge number of devices with high accuracy.

8.2.2 Live Traffic IoT Profiling. A profiling approach that could profile the live network capture devices without using any training data would outperform traditional IoT profiling techniques. Profiling devices based on the live network capture would not only reduce the training data processing overhead, it would also make identification easier for new devices that are added to the network. Some statistical features extracted from the IoT devices could help to achieve this goal.

8.2.3 Semi-supervised Frameworks. A semi-supervised framework can help in profiling IoT devices with less dependency on the training data. However, a good profiling approach would be the one that does not depend on training data at all. Achieving this goal would be challenging due to the heterogeneity of the environment. In this regard, proposing a semi-supervised learning framework can be helpful to achieve IoT profiling being less dependent on the training data.

8.2.4 Implementation of ML at the edge . Most of the IoT profiling techniques depend on the centralized approach. There is a need for distributive profiling of IoT devices to profile devices in the local node. The distributive approach would also preserve the privacy of devices and their users, since the traffic pattern of devices are not shared with the centralized server; instead, it is shared with the local fog/edge node.

8.2.5 Integration of ML with blockchain for IoT security. Blockchain is known for being a good way of transporting transparent, yet secure data. Implementing blockchain into IoT data transmission could exponentially increase IoT device security. It is worth considering the use of blockchain to profile devices, as it would preserve privacy and divide the overhead of the centralized server to edge nodes.

9 CONCLUSION

Due to the non-standardized implementation of IoT devices, coupled with surging rates of connected devices, there is a need for specialized tools to identify joined devices, as well as new ones attempting to connect to the network to maintain network integrity. It is essential that consumers of all levels can effectively identify, monitor, and isolate IoT devices based on their network behaviour.

IoT device identification can be challenging because of the existence of several device types, control sequences, and transmission protocols. Generalized procedures are not enough to accommodate the diverse nature of these devices. In addition, due to the limited resources IoT devices have, they have little to no security implementation, making them vulnerabilities on the network that are hard to identify. This paper serves as a thorough view of various profiling methods, such as device type, device instance, and anomaly detection. We investigate a taxonomy for IoT profiling based on a multitude of security issues in the field. Furthermore, we compiled a detailed list of common IoT vulnerabilities in various domains along with corresponding security adjustments. The list can contribute to improved risk assessment and mitigation of vulnerabilities. Organizations, network administrators, and regular consumers can utilize the propose list to identify possible security threats to their network, or devices that require adjustments for security.

We have also detailed common identification features used by various approaches to profile IoT devices, such as device type identification, device instance identification, and unseen device identification, along with the overhead of feature engineering in terms of efficiency and practicality of the features. The current research can help to choose the most realistic and effective collection of features for the best profiling method by choosing a set of features with the highest practicality and highest efficiency. This could achieve better performance both in terms of accuracy and scalability of the approach. Besides that, a hybrid collection of features with high practicality and efficiency may be used to produce the best profiling framework for real-world problems. This paper also addresses the machine learning and deep learning algorithms used for IoT device profiling. The machine learning classifier used by a majority of state-of-the-art approaches for profiling the IoT devices with the best performance has been identified.

REFERENCES

[139]]cisco [n. d.]. APIs, sdks, Sandbox, and community for Cisco developers. https://developer.cisco.com/docs/mud/#!what-is-mud/what-is-mud

[139]]calc [n. d.]. Common Vulnerability Scoring System Calculator. https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator [Accessed: 2021-01-30].

[139]]tp-link [n. d.]. TL-WN722N 150Mbps High Gain Wireless USB Adapter. https://www.tp-link.com/ca/home-networking/high-gain-adapter/tl-wn722n/ [Accessed: 2021-01-30].

[4] A. R. Abdallah and X. S. Shen. 2014. Lightweight lattice-based homomorphic privacy-preserving aggregation scheme for home area networks. In *2014 Sixth International Conference on Wireless Communications and Signal Processing (WCSP)*. 1–6.

[5] Abbas Acar, Hossein Fereidooni, Tigist Abera, Amit Kumar Sikder, Markus Miettinen, Hidayet Aksu, Mauro Conti, Ahmad-Reza Sadeghi, and Selcuk Uluagac. 2020. Peek-a-Boo: I see your smart home activities, even encrypted!. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. 207–218.

[6] Noura Al Nuaimi, Asma Alshamsi, Nader Mohamed, and Jameela Al-Jaroodi. 2015. e-Health Cloud Implementation Issues and Efforts. *IEOM 2015 - 5th International Conference on Industrial Engineering and Operations Management, Proceeding*. https://doi.org/10.1109/IEOM.2015.7093757

[7] Amani Al-Shawabka, Philip Pietraski, Sudhir B Pattar, Francesco Restuccia, and Tommaso Melodia. 2021. DeepLoRa: Fingerprinting LoRa Devices at Scale Through Deep Learning and Data Augmentation. In *Proceedings of the Twenty-second International Symposium*

on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing. 251–260.
[8] Amir H. Alavi, Pengcheng Jiao, William G. Buttlar, and Nizar Lajnef. 2018. Internet of Things-enabled smart cities: State-of-the-art and future trends. *Measurement* 129 (2018), 589 – 606. https://doi.org/10.1016/j.measurement.2018.07.067
[9] A. AlHammadi, A. AlZaabi, B. AlMarzooqi, S. AlNeyadi, Z. AlHashmi, and M. Shatnawi. 2019. Survey of IoT-Based Smart Home Approaches. In *2019 Advances in Science and Engineering Technology International Conferences (ASET)*. 1–6.
[10] W. Ali, G. Dustgeer, M. Awais, and M. A. Shah. 2017. IoT based smart home: Security challenges, security requirements and solutions. In *2017 23rd International Conference on Automation and Computing (ICAC)*. 1–6.
[11] F. Alsubaei, A. Abuhussein, and S. Shiva. 2017. Security and Privacy in the Internet of Medical Things: Taxonomy and Risk Assessment. In *2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops)*. 112–120.
[12] Nesrine Ammar, Ludovic Noirie, and Sébastien Tixeuil. 2020. Autonomous Identification of IoT Device Types based on a Supervised Classification. In *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 1–6.
[13] I. Andrea, C. Chrysostomou, and G. Hadjichristofi. 2015. Internet of Things: Security vulnerabilities and challenges. In *2015 IEEE Symposium on Computers and Communication (ISCC)*. 180–187.
[14] Noah Apthorpe, Dillon Reisman, Srikanth Sundaresan, Arvind Narayanan, and Nick Feamster. 2017. Spying on the Smart Home: Privacy Attacks and Defenses on Encrypted IoT Traffic. (08 2017).
[139]]a2-82 Sabir Hussain Awan, Sheeraz Ahmed, Nadeem Safwan, Zeeshan Najam, M Zaheer Hashim, and Tayybah Safdar. [n. d.]. Role of Internet of Things (IoT) with Blockchain Technology for the Development of Smart Farming. ([n. d.]).
[16] Lei Bai, Lina Yao, Salil S. Kanhere, Xianzhi Wang, and Zheng Yang. 2018. Automatic Device Classification from Network Traffic Streams of Internet of Things. *2018 IEEE 43rd Conference on Local Computer Networks (LCN)* (2018), 1–9.
[17] S. B. Baker, W. Xiang, and I. Atkinson. 2017. Internet of Things for Smart Healthcare: Technologies, Challenges, and Opportunities. *IEEE Access* 5 (2017), 26521–26544.
[18] Luís Barreto and António Amaral. 2018. Smart farming: Cyber security challenges. In *2018 International Conference on Intelligent Systems (IS)*. IEEE, 870–876.
[19] A Bartoli, J Hernández-Serrano, M Soriano, M Dohler, A Kountouris, and D Barthel. 2011. Security and privacy in your smart city. In *Proceedings of the Barcelona smart cities congress*, Vol. 292. 1–6.
[20] MJ Bogaardt, KJ Poppe, V Viool, and E van Zuidam. 2016. *Cybersecurity in the Agrifood sector*. Technical Report. Capgemini Consulting.
[21] Jens-Matthias Bohli, P Langendorfer, and Antonio F Skarmeta. 2013. Security and privacy challenge in data aggregation for the iot in smart cities. *Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems* (2013), 225–244.
[22] JULIE BORT. 2014. REFRIGERATOR HACKED: Here's The Biggest Problem Facing The Internet Of Things. https://www.businessinsider.com.au/hackers-use-a-refridgerator-to-attack-businesses-2014-1 [Accessed: 2021-01-30].
[23] Miloš Brajović, Stefan Vujović, and Slobodan Đukanović. 2015. An overview of smart irrigation software. In *2015 4th Mediterranean Conference on Embedded Computing (MECO)*. IEEE, 353–356.
[24] Leo Breiman. 2001. Random Forests. *Mach. Learn.* 45, 1 (Oct. 2001), 5–32. https://doi.org/10.1023/A:1010933404324
[25] Bezawada Bruhadeshwar, Maalvika Bachani, Jordan Peterson, Hossein Shirazi, Indrakshi Ray, and Indrajit Ray. 2018. IoTSense: Behavioral Fingerprinting of IoT Devices. *ArXiv* abs/1804.03852 (2018).
[26] A. Bytes, S. Adepu, and J. Zhou. 2019. Towards Semantic Sensitive Feature Profiling of IoT Devices. *IEEE Internet of Things Journal* 6, 5 (Oct 2019), 8056–8064. https://doi.org/10.1109/JIOT.2019.2903739
[27] Z Berkay Celik, Robert J Walls, Patrick McDaniel, and Ananthram Swami. 2015. Malware traffic detection using tamper resistant features. In *MILCOM 2015-2015 IEEE Military Communications Conference*. IEEE, 330–335.
[28] B. Charyyev and M. H. Gunes. 2020. Locality-Sensitive IoT Network Traffic Fingerprinting for Device Identification. *IEEE Internet of Things Journal* (2020), 1–1. https://doi.org/10.1109/JIOT.2020.3035087
[29] Hongmei Chi, Stephen Welch, Eugene Vasserman, and Ezhil Kalaimannan. 2017. A framework of cybersecurity approaches in precision agriculture. In *Proceedings of the ICMLG2017 5th International Conference on Management Leadership and Governance*. Acad. Conf. Publ. Int. Reading, UK, 90–95.
[30] Cédric Clastres. 2011. Smart grids: Another step towards competition, energy security and climate change objectives. *Energy policy* 39, 9 (2011), 5399–5408.
[31] R. Dagar, S. Som, and S. K. Khatri. 2018. Smart Farming – IoT in Agriculture. In *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)*. 1052–1056.
[32] P. Datta and B. Sharma. 2017. A survey on IoT architectures, protocols, security and smart city based applications. In *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. 1–5.
[33] B. D. Davis, J. C. Mason, and M. Anwar. 2020. Vulnerability Studies and Security Postures of IoT Devices: A Smart Home Case Study. *IEEE Internet of Things Journal* (2020), 1–1.
[34] Ruizhong Du, Jingze Wang, and Shuang Li. 2022. A Lightweight Flow Feature-Based IoT Device Identification Scheme. *Security and Communication Networks* 2022 (2022).

[35] Adel S Elmaghraby and Michael M Losavio. 2014. Cyber security challenges in Smart Cities: Safety, security and privacy. *Journal of advanced research* 5, 4 (2014), 491–497.

[139]]semi Linna Fan, Shize Zhang, Yichao Wu, Zhiliang Wang, Chenxin Duan, Jia Li, and Jiahai Yang. [n. d.]. An IoT Device Identification Method based on Semi-supervised Learning. ([n. d.]).

[37] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem. 2019. A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming. *IEEE Access* 7 (2019), 156237–156271.

[38] P. P. Gaikwad, J. P. Gabhane, and S. S. Golait. 2015. A survey based on Smart Homes system using Internet-of-Things. In *2015 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC)*. 0330–0335.

[39] Gartner. 2015. Smart Homes: The Next CPG Battleground? https://blogs.gartner.com/don-scheibenreif/2015/11/24/smart-homes-the-next-cpg-battleground/ [Accessed: 2021-01-30].

[40] D. Geneiatakis, I. Kounelis, R. Neisse, I. Nai-Fovino, G. Steri, and G. Baldini. 2017. Security and privacy issues for an IoT based smart home. In *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. 1292–1297.

[41] Dimitris Geneiatakis, Ioannis Kounelis, Ricardo Neisse, Igor Nai Fovino, Gary Steri, and Gianmarco Baldini. 2017. Security and privacy issues for an IoT based smart home. https://doi.org/10.23919/MIPRO.2017.7973622

[42] Ammar Gharaibeh, Mohammad A Salahuddin, Sayed Jahed Hussini, Abdallah Khreishah, Issa Khalil, Mohsen Guizani, and Ala Al-Fuqaha. 2017. Smart cities: A survey on data management, security, and enabling technologies. *IEEE Communications Surveys & Tutorials* 19, 4 (2017), 2456–2501.

[43] Sanjay Goel. 2015. Anonymity vs. security: The right balance for the smart grid. *Communications of the Association for Information Systems* 36, 1 (2015), 2.

[44] GSMA. 2018. New GSMA Study: Operators Must Look Beyond Connectivity to Increase Share of $1.1 Trillion IoT Revenue Opportunity. https://www.gsma.com/newsroom/press-release/new-gsma-study-operators-must-look-beyond-connectivity-to-increase-share/ [Accessed: 2021-01-30].

[45] The Guardian. 2016. The Guardian, Why the internet of things is the new magic ingredient for cyber criminals. http://engineering.purdue.edu/~mark/puthesis [Accessed: 2021-01-30].

[46] M. Gupta, M. Abdelsalam, S. Khorsandroo, and S. Mittal. 2020. Security and Privacy in Smart Farming: Challenges and Opportunities. *IEEE Access* 8 (2020), 34564–34584.

[47] Rajesh Gupta, Sudeep Tanwar, Sudhanshu Tyagi, and Neeraj Kumar. 2019. Tactile-Internet-Based Telesurgery System for Healthcare 4.0: An Architecture, Research Challenges, and Future Directions. *IEEE Network* 33 (2019), 22–29.

[48] Hamed HaddadPajouh, Ali Dehghantanha, Reza M Parizi, Mohammed Aledhari, and Hadis Karimipour. 2021. A survey on internet of things security: Requirements, challenges, and solutions. *Internet of Things* 14 (2021), 100129.

[49] S. A. Hamad, W. E. Zhang, Q. Z. Sheng, and S. Nepal. 2019. IoT Device Identification via Network-Flow Based Fingerprinting and Learning. In *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. 103–111. https://doi.org/10.1109/TrustCom/BigDataSE.2019.00023

[50] Jigna J. Hathaliya and Sudeep Tanwar. 2020. An exhaustive survey on security and privacy issues in Healthcare 4.0. *Computer Communications* 153 (2020), 311 – 335. https://doi.org/10.1016/j.comcom.2020.02.018

[51] Lars Huning, Jan Bauer, and Nils Aschenbruck. 2017. A Privacy Preserving Mobile Crowdsensing Architecture for a Smart Farming Application. In *Proceedings of the First ACM Workshop on Mobile Crowdsensing Systems and Applications*. 62–67.

[52] P. E. Idoga, M. Agoyi, E. Y. Coker-Farrell, and O. L. Ekeoma. 2016. Review of security issues in e-Healthcare and solutions. In *2016 HONET-ICT*. 118–121.

[53] Sidra Ijaz, Munam Ali Shah, Abid Khan, and Mansoor Ahmed. 2016. Smart cities: A survey on security concerns. *International Journal of Advanced Computer Science and Applications* 7, 2 (2016), 612–625.

[54] Andreas Jacobsson, Martin Boldt, and Bengt Carlsson. 2016. A risk analysis of a smart home automation system. *Future Generation Computer Systems* 56 (2016), 719 – 733. https://doi.org/10.1016/j.future.2015.09.003

[55] A. Jacobsson and P. Davidsson. 2015. Towards a model of privacy and security for smart homes. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. 727–732.

[56] Hossein Jafari, Oluwaseyi Omotere, Damilola Adesina, Hsiang-Huang Wu, and Lijun Qian. 2018. Iot devices fingerprinting using deep learning. In *MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM)*. IEEE, 1–9.

[57] Molly M Jahn, WL Oemichen, GF Treverton, et al. 2019. Cyber Risk and Security Implications in Smart Agriculture and Food Systems. *Accessed: Nov* 14 (2019), 2019.

[58] T. Kailath. 1967. The Divergence and Bhattacharyya Distance Measures in Signal Selection. *IEEE Transactions on Communication Technology* 15, 1 (1967), 52–60.

[59] Kai KANG, Zhi bo PANG, and Cong WANG. 2013. Security and privacy mechanism for health internet of things. *The Journal of China Universities of Posts and Telecommunications* 20 (2013), 64 – 68. https://doi.org/10.1016/S1005-8885(13)60219-8

[60] Ayush Kapoor, Suchetha I Bhat, Sushila Shidnal, and Akshay Mehra. 2016. Implementation of IoT (Internet of Things) and Image processing in smart agriculture. In *2016 International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS)*. IEEE, 21–26.
[61] Baltej Kaur, Danish Inamdar, Vishal Raut, Akash Patil, and Nayan Patil. 2016. A Survey On Smart Drip Irrigation System|. *International Research Journal of Engineering and Technology (IRJET)* 3, 02 (2016).
[62] V Keerthi and GN Kodandaramaiah. 2015. Cloud IoT based greenhouse monitoring system. *International Journal of Engineering Research and Applications* 5, 10 (2015), 35–41.
[63] C Kempenaar, C Lokhorst, EJB Bleumer, RF Veerkamp, Th Been, FK van Evert, MJ Boogaardt, L Ge, J Wolfert, CN Verdouw, et al. 2016. *Big Data analysis for smart farming: results of TO2 project in theme food security*. Technical Report. Wageningen University & Research.
[64] L. U. Khan, I. Yaqoob, N. H. Tran, S. M. A. Kazmi, T. N. Dang, and C. S. Hong. 2020. Edge Computing Enabled Smart Cities: A Comprehensive Survey. *IEEE Internet of Things Journal* (2020), 1–1.
[65] N. Komninos, E. Philippou, and A. Pitsillides. 2014. Survey in Smart Grid and Smart Home Security: Issues, Challenges and Countermeasures. *IEEE Communications Surveys Tutorials* 16, 4 (2014), 1933–1954.
[66] Jaidip Kotak and Yuval Elovici. 2020. IoT Device Identification Using Deep Learning. *arXiv preprint arXiv:2002.11686* (2020).
[67] M. Koutli, N. Theologou, A. Tryferidis, D. Tzovaras, A. Kagkini, D. Zandes, K. Karkaletsis, K. Kaggelides, J. Almela Miralles, V. Oravec, and S. Vanya. 2019. Secure IoT e-Health Applications using VICINITY Framework and GDPR Guidelines. In *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. 263–270.
[68] Denis Kozlov, Jari Veijalainen, and Yasir Ali. 2012. Security and Privacy Threats in IoT Architectures General. *Proceedings of the 7th International Conference on Body Area Networks (BodyNets '12)* (09 2012). https://doi.org/10.4108/icst.bodynets.2012.250550
[69] J Sathish Kumar and Dhiren Patel. 2014. A Survey on Internet of Things: Security and Privacy Issues. *International Journal of Computer Applications* 90 (02 2014). https://doi.org/10.5120/15764-4454
[70] Aparna Kumari, Sudeep Tanwar, Sudhanshu Tyagi, Neeraj Kumar, Michele Maasberg, and Kim-Kwang Raymond Choo. 2018. Multimedia big data computing and Internet of Things applications: A taxonomy and process model. *Journal of Network and Computer Applications* 124 (2018), 169 – 195. https://doi.org/10.1016/j.jnca.2018.09.014
[71] Saba Latif, Hamra Afzaal, and Nazir Ahmad Zafar. 2018. Modelling of graph-based smart parking system using internet of things. In *2018 International Conference on Frontiers of Information Technology (FIT)*. IEEE, 7–12.
[72] S. Latif and N. A. Zafar. 2017. A survey of security and privacy issues in IoT for smart cities. In *2017 Fifth International Conference on Aerospace Science Engineering (ICASE)*. 1–5.
[73] Changmin Lee, Luca Zappaterra, Kwanghee Choi, and Hyeong-Ah Choi. 2014. Securing smart home: Technologies, security challenges, and security requirements. 67–72. https://doi.org/10.1109/CNS.2014.6997467
[74] Seokcheol Lee, Jongwan Kim, and Taeshik Shon. 2016. User Privacy-Enhanced Security Architecture for Home Area Network of Smartgrid. *Multimedia Tools Appl.* 75, 20 (Oct. 2016), 12749–12764. https://doi.org/10.1007/s11042-016-3252-2
[75] T. Li, J. Ren, and X. Tang. 2012. Secure wireless monitoring and control systems for smart grid and smart home. *IEEE Wireless Communications* 19, 3 (2012), 66–73.
[76] Z. Li, S. Lu, S. Myagmar, and Y. Zhou. 2006. CP-Miner: finding copy-paste and related bugs in large-scale software code. *IEEE Transactions on Software Engineering* 32, 3 (2006), 176–192.
[77] Chiehyeon Lim, Kwang-Jae Kim, and Paul Maglio. 2018. Smart cities with big data: Reference models, challenges, and considerations. *Cities* 82 (05 2018). https://doi.org/10.1016/j.cities.2018.04.011
[78] Jun Lin, Zhiqi Shen, Anting Zhang, and Yueting Chai. 2018. Blockchain and IoT based Food Traceability System. *International Journal of Information Technology* 24, 1 (2018), 1–16.
[79] Jie Lin, Wei Yu, Nan Zhang, Xinyu Yang, Hanlin Zhang, and Wei Zhao. 2017. A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet of Things Journal* 4, 5 (2017), 1125–1142.
[80] Amy Poh Ai Ling and Mukaidono Masao. 2011. Selection of model in developing information security criteria on smart grid security system. In *2011 IEEE Ninth International Symposium on Parallel and Distributed Processing with Applications Workshops*. IEEE, 91–98.
[81] Zhao Liqiang, Yin Shouyi, Liu Leibo, Zhang Zhen, and Wei Shaojun. 2011. A crop monitoring system based on wireless sensor network. *Procedia Environmental Sciences* 11 (2011), 558–565.
[82] Manuel Lopez-Martin, Belen Carro, Antonio Sanchez-Esguevillas, and Jaime Lloret. 2017. Network traffic classifier with convolutional and recurrent neural networks for Internet of Things. *IEEE Access* 5 (2017), 18042–18050.
[83] Gilad David Maayan. 2020. The IoT Rundown For 2020: Stats, Risks, and Solutions. https://securitytoday.com/articles/2020/01/13/the-iot-rundown-for-2020.aspx [Accessed: 2021-01-30].
[84] Gilad David Maayan. 2020. Twelve highlights from our 2020 research. https://www.mckinsey.com/mgi/overview [Accessed: 2021-01-30].
[85] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zualkernan. 2015. Internet of things (IoT) security: Current status, challenges and prospective measures. In *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*. 336–341.

[86] T. Mantoro, M. A. Ayu, and S. M. binti Mahmod. 2014. Securing the authentication and message integrity for Smart Home using smart phone. In *2014 International Conference on Multimedia Computing and Systems (ICMCS)*. 985–989.
[87] Window Marc. 2019. Security in Precision Agriculture: Vulnerabilities and risks of agricultural systems.
[88] Samuel Marchal, Markus Miettinen, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and N Asokan. 2019. Audi: Toward autonomous iot device-type identification using periodic communication. *IEEE Journal on Selected Areas in Communications* 37, 6 (2019), 1402–1412.
[89] Antoni Martínez-Ballesté, Pablo A Pérez-Martínez, and Agusti Solanas. 2013. The pursuit of citizens' privacy: a privacy-aware smart city is possible. *IEEE Communications Magazine* 51, 6 (2013), 136–141.
[90] Noman Mazhar, Rosli Salleh, Muhammad Zeeshan, and M Muzaffar Hameed. 2021. Role of Device Identification and Manufacturer Usage Description in IoT Security: A Survey. *IEEE Access* 9 (2021), 41757–41786.
[91] Yair Meidan, Michael Bohadana, Asaf Shabtai, Juan Guarnizo, Martin Ochoa, Nils Ole Tippenhauer, and Yuval Elovici. 2017. ProfilIoT: A Machine Learning Approach for IoT Device Identification Based on Network Traffic Analysis. https://doi.org/10.1145/3019612.3019878
[92] M. S. Mekala and P. Viswanathan. 2017. A Survey: Smart agriculture IoT with cloud computing. In *2017 International conference on Microelectronic Devices, Circuits and Systems (ICMDCS)*. 1–7.
[93] Andrew Meola. 2020. How IoT are creating a more a more efficient precision agriculture industry. https://www.businessinsider.com/smart-farming-iot-agriculture [Accessed: 2021-01-30].
[139]]zig Microchip. [n. d.]. The Guardian, Why the internet of things is the new magic ingredient for cyber criminals. https://www.microchip.com/ [Accessed: 2021-01-30].
[95] M. Miettinen, S. Marchal, I. Hafeez, N. Asokan, A. Sadeghi, and S. Tarkoma. 2017. IoT SENTINEL: Automated Device-Type Identification for Security Enforcement in IoT. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. 2177–2184. https://doi.org/10.1109/ICDCS.2017.283
[96] Dragos Mocrii, Yuxiang Chen, and Petr Musilek. 2018. IoT-based smart homes: A review of system architecture, software, communications, privacy and security. *Internet of Things* 1-2 (2018), 81 – 98. https://doi.org/10.1016/j.iot.2018.08.009
[97] Fiona Edwards Murphy, Michele Magno, Padraig Whelan, and Emanuel Popo Vici. 2015. b+ WSN: Smart beehive for agriculture, environmental, and honey bee health monitoring—Preliminary results and analysis. In *2015 IEEE Sensors Applications Symposium (SAS)*. IEEE, 1–6.
[98] Ammar Awad Mutlag, Mohd Khanapi Abd Ghani, Net al Arunkumar, Mazin Abed Mohammed, and Othman Mohd. 2019. Enabling technologies for fog computing in healthcare IoT systems. *Future Generation Computer Systems* 90 (2019), 62–78.
[99] Christelle Nader and Elias Bou-Harb. 2021. Revisiting IoT Fingerprinting behind a NAT. In *2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*. IEEE, 1745–1752.
[100] Milind Naphade, Guruduth Banavar, Colin Harrison, Jurij Paraszczak, and Robert Morris. 2011. Smarter cities and their innovation challenges. *Computer* 44, 6 (2011), 32–39.
[101] T. D. Nguyen, S. Marchal, M. Miettinen, H. Fereidooni, N. Asokan, and A. Sadeghi. 2019. DÏoT: A Federated Self-learning Anomaly Detection System for IoT. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*. 756–767. https://doi.org/10.1109/ICDCS.2019.00080
[102] Sukhvir Notra, Muhammad Siddiqi, Hassan Habibi Gharakheili, Vijay Sivaraman, and Roksana Boreli. 2014. An experimental study of security and privacy risks with emerging household appliances. *2014 IEEE Conference on Communications and Network Security* (2014), 79–84.
[103] I. Olaronke and O. Oluwaseun. 2016. Big data in healthcare: Prospects, challenges and resolutions. In *2016 Future Technologies Conference (FTC)*. 1152–1157.
[139]]openwrt openwrt. [n. d.]. Welcome to the OpenWrt Project. https://openwrt.org/ [Accessed: 2021-01-30].
[139]]blue Ossmann and Michael. [n. d.]. Project Ubertooth. https://github.com/greatscottgadgets/ubertooth [Accessed: 2021-01-30].
[106] R. Pankomera and D. van Greunen. 2016. Privacy and security issues for a patient-centric approach in public healthcare in a resource constrained setting. In *2016 IST-Africa Week Conference*. 1–10.
[107] Nisha Panwar, Shantanu Sharma, Sharad Mehrotra, Lukasz Krzywiecki, and Nalini Venkatasubramanian. 2019. Smart Home Survey on Security and Privacy. *CoRR* abs/1904.05476 (2019). arXiv:1904.05476 http://arxiv.org/abs/1904.05476
[108] Akash Suresh Patil, Bayu Adhi Tama, Youngho Park, and Kyung-Hyune Rhee. 2017. A framework for blockchain based secure smart green house farming. In *Advances in Computer Science and Ubiquitous Computing*. Springer, 1162–1167.
[109] R. Priya, S. Sivasankaran, P. Ravisasthiri, and S. Sivachandiran. 2017. A survey on security attacks in electronic healthcare systems. In *2017 International Conference on Communication and Signal Processing (ICCSP)*. 0691–0694.
[110] P Rajalakshmi and S Devi Mahalakshmi. 2016. IOT based crop-field monitoring and irrigation automation. In *2016 10th International Conference on Intelligent Systems and Control (ISCO)*. IEEE, 1–6.
[111] Jingjing Ren, Daniel Dubois, David Choffnes, Anna Mandalari, Roman Kolcun, and Hamed Haddadi. 2019. Information Exposure From Consumer IoT Devices: A Multidimensional, Network-Informed Measurement Approach. *IMC '19: Proceedings of the Internet Measurement Conference*, 267–279. https://doi.org/10.1145/3355369.3355577

[139]]grandview Grandview Research. [n. d.]. https://www.grandviewresearch.com/ [Accessed: 2021-01-30].
[113] Jonathan Roux, Eric Alata, Guillaume Auriol, Vincent Nicomette, and Mohamed Kaâniche. 2017. Toward an intrusion detection approach for IoT based on radio communications profiling. In *2017 13th European Dependable Computing Conference (EDCC)*. IEEE, 147–150.
[114] Ola Salman, Imad H Elhajj, Ali Chehab, and Ayman Kayssi. 2019. A machine learning based framework for IoT device identification and abnormal traffic detection. *Transactions on Emerging Telecommunications Technologies* (2019), e3743.
[115] Mourjo Sen, Anuvabh Dutt, Shalabh Agarwal, and Asoke Nath. 2013. Issues of privacy and security in the role of software in smart cities. In *2013 International Conference on Communication Systems and Network Technologies*. IEEE, 518–523.
[116] Laxmi S Shabadi and Hemavati B Biradar. 2008. Design and implementation of IOT based smart security and monitoring for connected smart farming. *Int. J. Comput. Appl.* 975 (2008), 8887.
[117] Mustafizur Rahman Shahid, Gregory Blanc, Zonghua Zhang, and Hervé Debar. 2018. IoT Devices Recognition Through Network Traffic Analysis. https://doi.org/10.1109/BigData.2018.8622243
[118] Yaman Sharaf-Dabbagh and Walid Saad. 2018. Authentication of Everything in the Internet of Things: Learning and Environmental Effects. *CoRR* abs/1805.00969 (2018). arXiv:1805.00969 http://arxiv.org/abs/1805.00969
[119] N. M. Shrestha, A. Alsadoon, P. W. C. Prasad, L. Hourany, and A. Elchouemi. 2016. Enhanced e-health framework for security and privacy in healthcare system. In *2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC)*. 75–79.
[120] Sandra Siby, Rajib Maiti, and Nils Ole Tippenhauer. 2017. IoTScanner: Detecting and Classifying Privacy Threats in IoT Neighborhoods. (01 2017).
[121] S. Sicari, A. Rizzardi, L.A. Grieco, and A. Coen-Porisini. 2015. Security, privacy and trust in Internet of Things: The road ahead. *Computer Networks* 76 (2015), 146 – 164. https://doi.org/10.1016/j.comnet.2014.11.008
[122] Bhagya Silva, Murad Khan, and Kijun Han. 2018. Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities. *Sustainable Cities and Society* 38 (02 2018). https://doi.org/10.1016/j.scs.2018.01.053
[123] Arunan Sivanathan, Hassan Habibi Gharakheili, Franco Loi, Adam Radford, Chamith Wijenayake, Arun Vishwanath, and Vijay Sivaraman. 2018. Classifying IoT Devices in Smart Environments Using Network Traffic Characteristics. *IEEE Transactions on Mobile Computing* PP (08 2018), 1–1. https://doi.org/10.1109/TMC.2018.2866249
[124] A. Sivanathan, D. Sherratt, H. H. Gharakheili, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman. 2017. Characterizing and classifying IoT traffic in smart cities and campuses. In *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. 559–564.
[125] V. Sivaraman, H. H. Gharakheili, A. Vishwanath, R. Boreli, and O. Mehani. 2015. Network-level security and privacy control for smart-home IoT devices. In *2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. 163–167.
[126] S Srisruthi, N Swarna, GM Susmitha Ros, and Edna Elizabeth. 2016. Sustainable agriculture using eco-friendly and energy efficient sensor technology. In *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*. IEEE, 1442–1446.
[127] A. Strielkina, O. Illiashenko, M. Zhydenko, and D. Uzun. 2018. Cybersecurity of healthcare IoT-based systems: Regulation and case-oriented assessment. In *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*. 67–73.
[128] George Suciu, Alexandru Vulpe, Simona Halunga, Octavian Fratu, Gyorgy Todoran, and Victor Suciu. 2013. Smart cities built on resilient cloud computing and secure internet of things. In *2013 19th international conference on control systems and computer science*. IEEE, 513–518.
[129] D. Sun, J. Huai, J. Sun, J. Zhang, and Z. Feng. 2008. A new design of wearable token system for mobile device security. *IEEE Transactions on Consumer Electronics* 54, 4 (2008), 1784–1789.
[130] Guanglu Sun, Lili Liang, Teng Chen, Feng Xiao, and Fei Lang. 2018. Network traffic classification based on transfer learning. *Computers & electrical engineering* 69 (2018), 920–927.
[131] V. Thangavelu, D. M. Divakaran, R. Sairam, S. S. Bhunia, and M. Gurusamy. 2019. DEFT: A Distributed IoT Fingerprinting Technique. *IEEE Internet of Things Journal* 6, 1 (2019), 940–952.
[132] Jay Thom, Nathan Thom, Shamik Sengupta, and Emily Hand. 2022. Smart Recon: Network Traffic Fingerprinting for IoT Device Identification. In *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 0072–0079.
[133] Stephen Ugwuanyi and James Irvine. 2020. Security Analysis of IoT Networks and Platforms. In *2020 International Symposium on Networks, Computers and Communications (ISNCC)*. 1–6. https://doi.org/10.1109/ISNCC49221.2020.9297267
[134] Joost Verbraeken, Matthijs Wolting, Jonathan Katzy, Jeroen Kloppenburg, Tim Verbelen, and Jan S. Rellermeyer. 2020. A Survey on Distributed Machine Learning. *ACM Comput. Surv.* 53, 2, Article 30 (March 2020), 33 pages. https://doi.org/10.1145/3377454
[135] Wei Wang, Ming Zhu, Xuewen Zeng, Xiaozhou Ye, and Yiqiang Sheng. 2017. Malware traffic classification using convolutional neural network for representation learning. In *2017 International Conference on Information Networking (ICOIN)*. IEEE, 712–717.
[136] Zhanyi Wang. 2015. The applications of deep learning on traffic identification. *BlackHat USA* 24, 11 (2015), 1–10.

[137] Mi Wen, Jingsheng Lei, and Zhongqin Bi. 2013. Sse: A secure searchable encryption scheme for urban sensing and querying. *International Journal of Distributed Sensor Networks* 9, 12 (2013), 302147.
[138] Jason West. 2018. A prediction model framework for cyber-attacks to precision agriculture technologies. *Journal of Agricultural & Food Information* 19, 4 (2018), 307–330.
[139]]a1-1 Wikipedia. [n. d.]. Precision agriculture. https://en.wikipedia.org/wiki/Precision_agriculture [Accessed: 2021-01-30].
[140] Charlie Wilson, Tom Hargreaves, and Richard Hauxwell-Baldwin. 2014. Smart homes and their users: A systematic analysis and key challenges. *Personal and Ubiquitous Computing* 19 (02 2014), 463–476. https://doi.org/10.1007/s00779-014-0813-0
[141] Kuai Xu, Yinxin Wan, Guoliang Xue, and Feng Wang. 2019. Multidimensional behavioral profiling of internet-of-things in edge networks. In *IWQoS '19*.
[142] L. D. Xu, W. He, and S. Li. 2014. Internet of Things in Industries: A Survey. *IEEE Transactions on Industrial Informatics* 10, 4 (2014), 2233–2243.
[143] Poonam Yadav, Angelo Feraudo, Budi Arief, Siamak F Shahandashti, and Vassilios G Vassilakis. 2020. Position paper: A systematic framework for categorising IoT device fingerprinting mechanisms. In *Proceedings of the 2nd International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things*. 62–68.
[144] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao. 2017. A Survey on Security and Privacy Issues in Internet-of-Things. *IEEE Internet of Things Journal* 4, 5 (2017), 1250–1258.
[145] Feihong Yin, Li Yang, Jianfeng Ma, Yasheng Zhou, Yuchen Wang, and Jiahao Dai. 2021. Identifying IoT Devices Based on Spatial and Temporal Features from Network Traffic. *Security and Communication Networks* 2021 (2021).
[146] Yawei Yue, Shancang Li, Phil Legg, and Fuzhong Li. 2021. Deep Learning-Based Security Behaviour Analysis in IoT Environments: A Survey. *Security and Communication Networks* 2021 (2021).
[147] Jiliang Zhang, Chaoqun Shen, Haihan Su, Md Tanvir Arafin, and Gang Qu. 2021. Voltage over-scaling-based lightweight authentication for IoT security. *IEEE Trans. Comput.* (2021).
[148] Eda Zhou, Joseph Turcotte, and Lorenzo De Carli. 2020. Enabling Security Analysis of IoT Device-to-Cloud Traffic. In *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. 1888–1894. https://doi.org/10.1109/TrustCom50675.2020.00258

10 APPENDIX

Table 7 lists the possible Vulnerabilities along with their description in different IoT devices.

Table 7. List of Vulnerabilities along with their description in different IoT devices.

ID	Vulnerability
V-1:	Amazon echo 1st generation speakers are susceptible to KRACK (series of weaknesses in the WPA2 protocol dubbed the "Key Re-installation Attack") (CVE-2017-13078)
V-2:	CVE-2018-11567
V-3:	CVE-2017-13077
V-4:	Microphones will respond to a focused light (laser pointer) pointed directly at them.
V-5:	A group of students showed they could hide commands in white noise and YouTube videos played over loud-speakers to get smart devices to turn on airplane mode or open a website.
V-6:	Researchers at Checkmarx turned Alexa into a spy device with no intensive meddling required.
V-7:	How Amazon's voice assistant gadget might be silently hijacked for surveillance (Chinese researchers).
V-8:	CVE-2019-13336
V-9:	Ring Doorbells had vulnerability leaking Wi-Fi login info.
V-10:	In August 2016, a vulnerability with August's Guest Access allowed guests to hack August's software and "enroll a new key".
V-11:	August Doorbell is susceptible to KRACK.
V-12:	CVE-2018-6692
V-13:	CVE-2019-12780
V-14:	CVE-2018-1144 or CVE-2018-1143
V-15:	CVE-2018-1145
V-16:	CVE-2018-1146
V-17:	CVE-2013-6948
V-18:	CVE-2013-6949
V-19:	CVE-2013-6950
V-20:	CVE-2013-6951
V-21:	CVE-2013-6952
V-22:	CVE-2015-5987
V-23:	CVE-2015-5988
V-24:	CVE-2015-5989
V-25:	CVE-2015-5990
V-26:	CVE-2017-13078: Re-installation of group key in 4-way handshake.
V-27:	CVE-2017-13079: Re-installation of the integrity group key in 4-way handshake.
V-28:	CVE-2017-13080: Re-installation of the group key in the group key handshake.
V-29:	CVE-2017-13081: Re-installation of the integrity group key in the group key handshake.
V-30:	CVE-2017-13087: Re-installation of the group key (GTK) when processing a Wireleess Network Management (WNM) Sleep Mode Response frame.
V-31:	CVE-2017-13088: Re-installation of the integrity group key (IGTK) when processing a wireless network management (WNM) Seep Mode Response frame
V-32:	CVE-2019-13523
V-33:	CVE-2017-14263
V-34:	CVE-2017-5671
V-35:	CVE-2015-7908
V-36:	CVE-2015-2848
V-37:	CVE-2015-2847
V-38:	The HomeHack vulnerability was disclosed in July 2017 affects LG's SmartThinQ mobile app which is used to control all of LG's smart home appliances.
V-39:	CVE-2019-17101
V-40:	CVE-2019-5035
V-41:	CVE-2019-5043
V-42:	CVE-2019-5034
V-43:	CVE-2019-5036
V-44:	CVE-2019-5037
V-45:	CVE-2019-5038
V-46:	CVE-2019-5039
V-47:	CVE-2019-5040
V-48:	The hardware infrastructure lacks proper protection, allowing attackers to install malicious software into the unit.
V-49:	CVE-2020-6007
V-50:	CVE-2018-3856
V-51:	CVE-2018-3863

ID	Vulnerability
V-52:	CVE-2018-3864
V-53:	CVE-2018-3865
V-54:	CVE-2018-3866
V-55:	CVE-2018-3867
V-56:	CVE-2018-3872
V-57:	CVE-2018-3878
V-58:	CVE-2018-3879
V-59:	CVE-2018-3880
V-60:	CVE-2018-3897
V-61:	CVE-2018-3902
V-62:	CVE-2018-3904
V-63:	CVE-2018-3905
V-64:	CVE-2018-3906
V-65:	CVE-2018-3907 CVE-2018-3908 CVE-2018-3909
V-66:	CVE-2018-3911
V-67:	CVE-2018-3912 CVE-2018-3917
V-68:	CVE-2018-3918
V-69:	CVE-2018-3919
V-70:	CVE-2018-3925
V-71:	CVE-2018-3926
V-72:	CVE-2018-3927
V-73:	CVE-2018-11316
V-74:	CVE-2019-3950
V-75:	CVE-CVE-2019-3949
V-76:	CVE-2016-10115
V-77:	CVE-2016-10116
V-78:	CVE-2017-13077
V-79:	CVE-2017-13078
V-80:	CVE-2017-13079
V-81:	CVE-2017-13080
V-82:	CVE-2017-13081
V-83:	CVE-2017-13082
V-84:	CVE-2017-13084
V-85:	CVE-2017-13086
V-86:	CVE-2017-13087
V-87:	CVE-2017-13088
V-88:	Researchers at Checkmarx said they have discovered the potential flaws in the Trifo Ironpie M6 smart vacuum cleaner. (read more...).
V-89:	CVE-2017-10987
V-90:	CVE-2017-10988
V-91:	Samsung's Smart TV privacy policy defined by the company says "if your spoken words include personal or other sensitive information, that information will be captured and transmitted to a third party. (read more...)
V-92:	CVE-2013-4861
V-93:	CVE-2017-5249
V-94:	CVE-2017-13077
V-95:	CVE-2017-11578
V-96:	CVE-2017-11580
V-97:	CVE-2017-11579
V-98:	CVE-2014-10374
V-99:	CVE-2019-6528
V-100:	CVE-2018-5303
V-101:	CVE-2018-5304 CVE-2019-13991
V-102:	CVE-2021-30354