Device Discovery in the Smart Home Environment

Mounib Khanafer*, Logan Kostick†, Chixiang Wang‡, Wondimu Zegeye§,
Weijia He‡, Berkay Kaplan¶, Nurzaman Ahmed‡, Kevin Kornegay§, David Kotz‡, Timothy J. Pierson‡
* mkhanafer@auk.edu.kw, American University of Kuwait, Salmiya, Kuwait
† lkostic1@jhu.edu, Johns Hopkins University, Baltimore, MD
‡{chixiang.wang.gr, weijia.he, nurzaman.ahmed, david.f.kotz, timothy.j.pierson}@dartmouth.edu, Dartmouth College, Hanover, NH
§ {wozeg, kevin.kornegay}@morgan.edu, Morgan State University, Baltimore, MD
¶ berkayk2@illinois.edu, University of Illinois, Urbana-Champaign, IL

***Abstract*—With the availability of Internet of Things (IoT) devices offering varied services, smart home environments have seen widespread adoption in the last two decades. Protecting privacy in these environments becomes an important problem because IoT devices may collect information about the home's occupants without their knowledge or consent. Furthermore, a large number of devices in the home, each collecting small amounts of data, may, in aggregate, reveal non-obvious attributes about the home occupants. A first step towards addressing privacy is discovering what devices are present in the home. In this paper, we formally define device discovery in smart homes and identify the features that constitute discovery in that environment. Then, we propose an evaluative rubric that rates smart home technology initiatives on their device discovery capabilities and use it to evaluate four commonly deployed technologies. We find none cover all device discovery aspects. We conclude by proposing a combined technology solution that provides comprehensive device discovery tailored to smart homes.**

I. Introduction

The Internet of Things (IoT) transforms everyday things into 'smart' objects that can compute and connect with other devices. These *smart things* – TVs, watches, refrigerators, light bulbs, door locks, and so forth – were not traditionally part of the Internet. With IoT capabilities, however, these devices can collect a large amount of data about home occupants. With potentially dozens (or even hundreds) of smart devices operating in a home soon, an adversary has a large attack surface to exploit. Learning about what devices are operating in the home environment is, therefore, an essential step toward creating a safe environment where the occupants' privacy is protected. That is, *device discovery* becomes a critical problem for privacy in smart homes.

Device discovery, as a general challenge, has been studied thoroughly; Roopa et al. [1] provides an excellent survey. The literature lacks, however, a comprehensive definition of device discovery tailored to smart homes. In this paper, we address that shortcoming and suggest that comprehensive smart-home device discovery should encompass four elements:

1) *Device presence* – detecting all devices in a home, even if they do not cooperate with a discovery inquiry;
2) *Device identity* – identifying each device's make, model, and software version;
3) *Device membership* – determining which devices are part of the home's infrastructure and which devices are not part of the home (e.g., differentiate between a home's devices and transient or neighbor's devices); and
4) *Device location* – locating the device in three dimensions so home residents can physically find the device.

After discussing why device discovery in modern smart homes is challenging, we define in Section III these four discovery elements and justify why each is essential to smart home-oriented device discovery. In Section IV, we propose an evaluative rubric for reasoning about device discovery. We then review a confusing alphabet soup of disparate industry-supported initiatives designed to solve device discovery and evaluate them against our rubric. None of the industry-supported projects fully support all four aspects of comprehensive device discovery in smart homes. Finally, in Section V, we propose a comprehensive solution that combines several technologies. In summary, this paper makes the following **contributions**:

- A comprehensive definition of device discovery tailored to the smart-home environment;
- An evaluative rubric that scores smart-home technology initiatives on their device discovery capabilities; and
- A proposed approach that combines multiple industry-supported initiatives to discover devices in a smart home according to that definition and rubric.

II. Background

In this section, we briefly review why device discovery is challenging in a modern smart home comprised of a large number of heterogeneous devices. We discuss how existing discovery methods used in homes are insufficient and highlight how approaches used in corporate networks also fail to suit the particular needs of residents in smart homes.

Today, households commonly host numerous devices used by a single family member, such as laptops, tablets, and smartphones. In a typical four-person household, this already results in over a dozen devices. Additionally, homes increasingly contain devices used by multiple residents, such as smart speakers like Amazon Echo or Apple HomePod, and other devices like smart light bulbs, door locks, or thermostats. Shafiq et al. predict homes will contain an average of nearly ten networked devices per person within the next two years [2]. Thus, homes of the near future will likely contain dozens

or even hundreds of smart devices. Managing this many devices becomes a challenge as residents struggle to track which devices should be present and to remember the data each device collects. To address this issue, residents need a comprehensive way to discover home devices.

A. Existing device discovery methods are insufficient

In addition to a visual search, many existing device discovery methods involve device interrogation, network traffic analysis, or machine learning to identify devices [3]. These methods, however, are not sufficient for modern smart homes.

1) Visual search: Many smart-home devices often appear indistinguishable from conventional counterparts. For example, a smart door lock resembles a regular lock until its sensor is activated. A visual inventory could mistake smart devices for their 'dumb' counterparts and miss them. A visual search is also labor-intensive and error-prone. Additionally, some devices may intentionally be out-of-sight to evade visual detection, as seen with 'spy cameras' in Airbnb residences [4]. An ideal device discovery solution would identify and localize *all* devices present in the home, even if they are hidden from view.

2) Device interrogation: Some approaches to device discovery rely on one device sending an inquiry message across a network; compatible devices then respond to the inquiry message, and the sender inventories the responding devices. Some devices may not respond to a discovery inquiry, however, either because they do not implement the interrogation protocol, or because they intentionally ignore the query (to avoid detection). Still, others may respond with misinformation to disguise their true nature. A smart-home-oriented device discovery solution must account for all devices – even if they fail to respond to discovery inquiries.

3) Sniffing: Heterogeneous devices in a smart home will likely communicate over many protocols and frequency bands. Non-mobile devices, such as smart refrigerators, may be wired over Ethernet. Others may communicate wirelessly using protocols such as Wi-Fi, Bluetooth, Zigbee, Thread, LoRA, or cellular. Even within a single protocol, there may be many frequency bands in use. A sniffer attempting to detect all devices in a home would need to be capable of every protocol and would need to cover multiple bands and frequencies. Furthermore, some devices may transmit infrequently (or at all). Missing one of these infrequent transmissions, at best, means a long delay in discovering the device.

B. Neighbors' devices will further complicate some scenarios

In the device discovery literature, there is often an implicit assumption that a home is a free-standing residence occupied by a single family (or even a single person). In many cases, however, the situation is more complicated. In an apartment building, for example, there may be many residents living in separate apartments but near each other. Any technique to discover devices based on their wireless transmissions will detect signals that originated from devices in other apartments. A device discovery protocol should determine which devices belong to a particular home and which do not.

Devices that are not part of the home's network may not be a threat – they may be a neighbor's device going about its regular business. Devices that are outside the home but attempt to join the home network – or communicate with devices in the home's network – may be a security concern. Likewise, unknown devices located inside the home may be of concern. A discovery solution should be able to determine whether each device belongs to the home network, and whether that device is physically inside or outside the home.

C. Corporate approaches do not apply to homes

Corporate networks managed by professional IT staff are accustomed to dealing with a large number of devices. In corporate environments, IT departments purchase, inventory, and configure devices before providing them to employees. Similarly, professional IT staff decommission devices and remove them from inventory when devices are no longer needed. This gives the IT staff the opportunity to keep detailed records of the number of and types of devices the company owns. Households, however, are not (usually) staffed by IT professionals. The purchase and installation of devices may be done by several residents. For example, consider a situation where multiple roommates share a residence. It could be the case that each roommate buys and installs their own devices. In this scenario, the landlord may also buy and install devices in the residence. As a result, some residents may not be aware of devices purchased and installed by others.

III. Defining Device Discovery

We have seen that existing approaches to device discovery do not adequately cover the unique needs of a smart home. In this section, we formulate a comprehensive definition of the device discovery problem in the context of smart-home environments. We envision *device discovery* to rest on four features, namely: *presence*, *identity*, *membership*, and *location*. We first describe each feature and then present a rubric that evaluates four commonly deployed technology initiatives related to device discovery.

A. Device Presence

We define *device presence* to mean that an electronic device is *inside* the home or is *nearby*. That is, our interest is to confirm if a device is *currently* present in the home. As it is challenging to craft a precise definition of inside, we consider any device present within the walls that enclose the home to be *inside*, and any device within short-range radio range (e.g., within Wi-Fi range, but not necessarily within long-range protocols such as cellular or LoRA) of the home network to be *nearby*. In some home contexts, such as an apartment building, a device in the neighboring apartment is *nearby* but *not inside*. In other contexts, such as a self-standing single-family home, nearby may include a porch, deck, patio, or garage.

In terms of device presence, we envision a range of capabilities for device discovery, from the most basic to the most sophisticated. We rank the level of detection sophistication as:

- **Low**: presence is discovered if it actively joins the home network. In that case, a device such as a router will learn of the device's presence. This includes the case when a device responds to a device discovery inquiry (e.g., the router "pings" a device in some manner, and the device responds). This approach requires low sophistication to detect devices, but may not yield much information about the discovered device.
- **Medium**: presence is discovered if it is observed communicating using any of the home's network protocols (e.g., Wi-Fi, Bluetooth, Zigbee), even if it does not attempt to join the home's network.
- **High**: presence is discovered if it is observed communicating with *any* network protocol – even if other devices in the home do not use that protocol (e.g., cellular or LoRA) – or does not transmit at all. Ideally, devices are discovered even if they are powered off.

B. Device Identity

We define *device identity* as the determination of the device's type, make, model, version, and instance of a device present within the home. For example, at one author's home is a Samsung 43UK6300PUE television, serial number 807MXAYL1720, running webOS 05.10.45; that information defines the make, model, type, instance, and version, respectively. In some devices, there may be additional dimensions about the model (for example, a given refrigerator model may have an optional icemaker) or version (OS and application versions).

We envision a system that can *identify* device information to different degrees. We rank identification capabilities as:

- **Low**: the method can only determine a device's unique identifier, such as the MAC address or serial number. This information distinguishes this device from other identical devices but only derives basic information such as device manufacturer by OUI lookup.
- **Medium**: determine the instance, make, model, and type (e.g., by querying the device or a directory service).
- **High**: determine the characteristics defined as Medium plus a device's current firmware, software, OS versions, and other attributes (e.g., does the refrigerator have an ice maker).

C. Device Membership

Device membership attempts to discern if a device belongs to a home. We define 'belonging' to mean a device that is *intended* by the residents (or possibly another person with a legitimate right to the home, such as a landlord) to be part of the home. As noted above, in a dense living environment such as an apartment building, many devices may be 'present' nearby. However, some devices may belong to neighbors or may be carried by people as they travel near a home. Those devices are not considered members of the home's network.

A device discovery system may provide a range of membership capabilities. We rank membership as:

- **Low**: device's intended network is unknown.
- **Medium**: determine that the device is intended to be part of a home.
- **High**: determine the particular person or organization that currently owns or primarily operates the device (e.g., this is Alice's iPad). In some cases the owner and operator may be different. For example, a smart thermostat may be owned by a landlord but operated by a tenant.

D. Device Location

We define *device location* to indicate the device's physical location within the home. A discovery system may be able to determine the location of a device present in the home in varying forms:

- **Low**: can only determine whether the device is inside or outside the boundaries of the home.
- **Medium**: determine in which room the device is located, or that it is outside.
- **High**: determine the device's 3D location within several centimeters relative to a coordinate system representing the home.

IV. Evaluating commonly deployed technology initiatives

Several technology initiatives are commonly deployed in IoT environments. We study the capabilities of popular technologies supported by major technology companies such as Apple, Cisco, Samsung, or Amazon, are promoted by alliances such as FIDO Alliance or Connectivity Standards Alliance, or are approved by standardization bodies such as NIST or IETF. In particular, we consider FIDO, MUD, NETCONF, and Matter. While this is not an exhaustive list of smart home technology initiatives, this paper aims to study how these popular IoT-based systems support device discovery. We first summarize our evaluative rubric, then provide an overview of these technology initiatives, assessing their support of device discovery in the next section using our rubric.

A. Rubric

Based on the descriptions above, we introduce an evaluative rubric that is summarized in Table I. This rubric helps categorize technology initiatives based on the extent to which they support the four facets of device discovery in smart homes. By examining different initiatives on each discovery facet, each initiative's relative strengths and limitations are revealed.

We rate each of the four technology initiatives on each discovery facet (presence, identity, membership, location) on a scale of None, Low, Medium or High in terms of their ability to discover IoT devices in a smart home environment. None means the technology has no capabilities in a facet, other ratings follow the descriptions above. Using a level, rather than a numeric value achieves two goals: 1) the rubric is easier to use because it is often hard to describe an initiative's support of a feature with a precise numeric value, and 2) the rubric better recognizes that different initiatives come with diverse capabilities and comparing them against specific features is not a straightforward task.

TABLE I
RUBRIC FOR COMPARING SMART HOME DISCOVERY TECHNOLOGIES.

	Low	Medium	High
Presence	Device detected if it actively joins the home network.	Detected if observed communicating using any of the home's network protocols, even if it does not attempt to join the home's network.	Detected if observed communicating with any network protocol or does not transmit at all.
Identity	Can only determine a device's unique identifier.	Can determine a device's make, model, and type.	Can determine a device's current firmware, software, OS versions, and other attributes.
Membership	Device's intended network is unknown.	Can determine that a device is intended to be part of a home.	Can determine the particular person or organization that currently owns or primarily operates the device.
Location	Can only determine whether the device is inside or outside the boundaries of the home.	Can determine in which room the device is located, or that it is outside.	Can determine the 3D location within several centimeters.

We now rate each of the four popular technology initiatives using our rubric.

B. FIDO

FIDO is a technology initiative to authenticate devices to online services [5]. It is supported by Amazon, Apple, Google, Intel, Meta, Microsoft, NIST, and many other contributors under the umbrella of the FIDO Alliance [6]. FIDO protocols use standard public-key cryptography techniques to provide strong authentication in a two-phase manner. During the first phase, a device is registered with an online service. The user's device creates a new public/private key pair, retaining the private key and registering the public key with the online service. In the second phase, a device attempts to authenticate to the online service. Authentication is done by the client device proving possession of the private key to the service by signing a service-issued challenge.

For device discovery, we observe the following:

- *Presence*: FIDO can detect the Presence of registered devices, but it cannot detect devices that have not been registered or communicate using protocols that the home network does not support. We give it a Low score for Presence as it can only detect registered devices that the user directs to actively join the network.
- *Identity*: FIDO supports device-specific modules that identify information like firmware updates and Wi-Fi network setup. Thus, FIDO's support of the Identity falls under the High level for devices that implement it (FIDO was penalized above for not detecting the presence of all devices, but it fully knows Identity of those it detects).
- *Membership*: FIDO knows that registered devices are part of the home's infrastructure and knows which user registered the device. We score FIDO in the High category.
- *Location*: FIDO does not localize devices. Thus, we rate it as None for Location.

C. MUD

Manufacturer Usage Description (MUD) is an architecture designed for manufacturers to clearly define how their IoT products should behave on a network [7]. This architecture helps manufacturers create a list of acceptable network behaviors for their IoT devices, reducing the potential for attacks.

MUD expects manufacturers to create a MUD file for each IoT product, which outlines the hosts their product should use to communicate, along with details such as port numbers, protocols, and traffic direction. The MUD file is a serialized YANG (Yet Another Next Generation) data model [8]. When a MUD-compatible device is installed, it will emit a URL pointing to its MUD file (available in the manufacturer's MUD File Server) and a signature from the manufacturer to verify the file's authenticity. The closest switch or router, serving as a MUD manager, retrieves the MUD file using the URL and checks the signature. If the signature is valid, the MUD manager notifies the network administrators, who can implement the specified access-control policies via access control lists (ACLs). If the device is later removed by an administrator, all associated access-control policies for the device are also removed.

Although MUD is proposed for network protection, it can also be used for device discovery and identification. A MUD-compatible router can detect the presence of an IoT device by capturing its emitted URL, making MUD capable of presence detection. However, if the device is not compatible with MUD and does not emit the MUD URL, then MUD does not detect the presence of the device.

The MUD file retrieved from the URL contains information about the device, including its model name. Even if the model name is not specified in the MUD file, the network behavior of different products or models can be distinct, making it possible to distinguish between different devices based on each device's MUD file.

For device discovery, we observe the following:

- *Presence*: When a MUD device emits its URL, the device can be detected. Non-MUD-compliant devices do not emit a URL and would not be detected using this technique. Additionally, MUD cannot detect devices that do no attempt to communicate over the home's network

or that communicate over protocols such as Zigbee that are not supported by the home's network. As with FIDO, MUD's support of Presence is categorized as Low.
- *Identity*: MUD provides detailed device attributes in the YANG data model. Therefore, its support of Identity is categorized as High.
- *Membership*: MUD only provides information about the device's capabilities and expected behavior, but does not identify the network to which a device is intended to join. Therefore, its support of Membership is categorized as None.
- *Location*: MUD cannot detect devices' locations. Therefore, its support of Location is categorized as None.

D. NETCONF

NETCONF (the NETwork CONFiguration protocol) provides mechanisms to install, manipulate, and delete the configuration of network devices like routers and switches [9]. The protocol defines a simple mechanism to manage a network device, retrieve configuration data information, and upload and manipulate new configuration data. In particular, NETCONF defines a Manager (the client) that communicates with an Agent (the server; installed at the network device) using remote procedures calls (RPCs) to retrieve its state data and manipulate its configuration data. NETCONF relies on YANG data models to describe the capabilities of a device. By viewing the list of capabilities, the developer learns about the data that can be retrieved or configured. YANG models can be created to describe any device or appliance, including IoT devices. It is worth mentioning that even though NETCONF/YANG is mainly used to manage and configure enterprise network devices, there have been several works in the literature that build architectures based on NETCONF/YANG to manage and configure IoT devices in a smart-home environment [10], [11].

For device discovery, we observe the following:

- *Presence*: NETCONF can connect to any compatible device connected to the network. NETCONF was originally designed to manage and configure routers or switches, but as noted above, can be extended to support IoT devices. Because it cannot detect non-NETCONF devices, like FIDO and MUD, its support of Presence fits in the Low category.
- *Identity*: Because it was originally intended for network routers and switches, NETCONF cannot determine an IoT device's Identity by default, but the YANG model can be extended to include administrator-entered attributes that describe the device's identity. Because NETCONF does not query IoT devices for live information about attributes such as firmware version, its support of Identity can be categorized as Medium for smart home devices.
- *Membership*: NETCONF by default cannot determine the device's membership. The YANG model, however, could be extended to include attributes entered by an administrator to define membership. Because NETCONF does not query IoT devices for live Membership characteristics, its support can be categorized as Medium for smart home devices.
- *Location*: NETCONF by default cannot determine the device's location unless the YANG model is extended with administrator-entered attributes that define the location. These static attributes, however, would not track the location of mobile devices. Therefore, NETCONF's support of Location can be categorized as Low level.

E. Matter

The Matter protocol, formerly known as Project Connected Home over Internet Protocol (CHIP), is an application framework standard maintained by the Connectivity Standards Alliance (CSA) [12]. Many well-known companies such as Amazon, Apple, Google, and Samsung actively support Matter. With these heavyweight backers invested in the technology, the protocol promises to proliferate in the IoT space.

Matter connects devices across various manufacturers and heterogeneous wireless technologies and ecosystems. The Matter protocol leverages existing IP technologies to build a unified wireless connectivity ecosystem for smart homes. IP-based networking provides manufacturers with simplified development while improving device compatibility for consumers. For instance, a user can control Apple devices using Google Voice.

For device discovery, we observe the following:

- *Presence*: Matter can detect the presence of devices registered with a Matter controller. It cannot detect unregistered devices. As with other initiatives, Matter's support of Presence falls under the Low level.
- *Identity*: Matter includes mechanisms to determine a registered device's identity including OS version, firmware, and several other characteristics. Therefore, for registered devices its support of Identity falls under the High level.
- *Membership*: Matter does not support a mechanism for identifying a device's membership, but it can identify devices that belong to the same network. Therefore, its support of Membership falls under the Medium level.
- *Location*: The Matter protocol enables GPS-equipped devices to include their location in a standardized format. Otherwise, Matter is unable to localize devices. Therefore, its support of Location falls under the Low level for most smart home devices.

F. Summary

Figure 1 compares the technologies in a radar chart. None of the reviewed initiatives adequately covers all aspects of device discovery as defined in Section III. A particular shortcoming of these initiatives is the lack of localization. Furthermore, none of the initiatives can detect devices that never transmit on the network, let alone devices that are powered off.

V. Comprehensive solution

We propose combining MUD and NETCONF into an approach with the capability to address all four aspects of

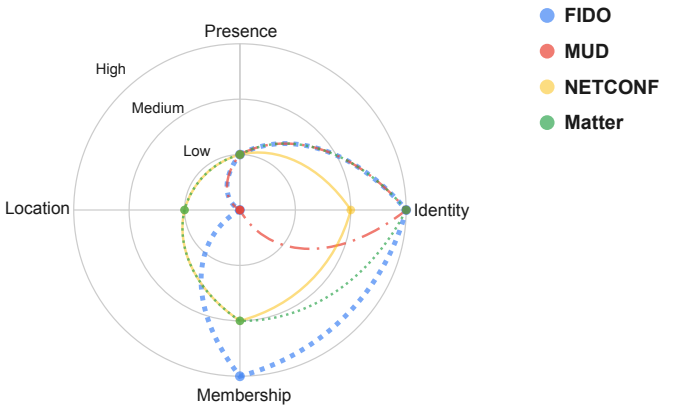

Fig. 1. Radar chart to visualize the support of device discovery features by FIDO, MUD, NETCONF, and Matter.

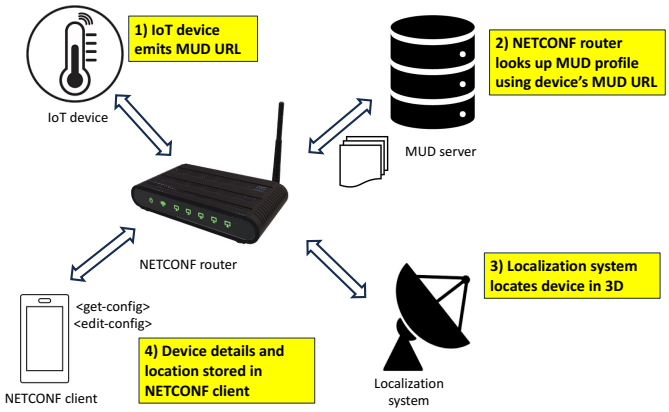

Fig. 2. A proposed system that combines MUD and NETCONF as well as a localization system for comprehensive device discovery. (1) IoT device emits MUD URL to NETCONF router, (2) NETCONF router looks up the device's MUD profile from MUD server, (3) NETCONF router instructs the localization system to locate the device in three dimensions, and (4) device details and location are stored in the NETCONF client as YANG models.

device discovery. To do so, we would need to integrate one of the many localization approaches described in the literature; Zafari et al. provide a comprehensive review [13]. MUD has strong support for identity features, and we can boost its support for location and membership features by leveraging the NETCONF protocol. This is possible by noticing that the MUD files collected by the MUD-compatible router for each IoT device at home are, in fact, YANG models, and NETCONF has the capability of retrieving a YANG model of a network device to edit it. As such, if a NETCONF server is implemented at the MUD-compatible router, a NETCONF client can retrieve YANG models of IoT devices (provided by MUD) and edit them to include information about devices' membership (also provided by MUD) and locations (from a localization system). This can be straightforwardly achieved using the NETCONF operations *get-config* to retrieve the configuration data of the router and *edit-config* to edit the device configuration data. This process is illustrated in Figure 2.

To go further, and detect devices even if they are powered off or do not transmit, these methods could be augmented by the 'harmonic radar' developed by Perez et al., which detects the presence of electronic devices based on non-linear electrical components [14], [15]. As discussed in Section II, this feature could be important to ensure *all* devices are discovered, regardless of their communication modality.

VI. Summary

This paper focuses on the problem of IoT device discovery in smart home environments. First, we define the features of device discovery: presence, identity, membership, and location. Second, we develop an evaluative rubric that assesses how a smart-home initiative supports each feature. We then apply that rubric to a confusing alphabet soup of existing technology initiatives. We find that none of them fully support device discovery in smart homes. Finally, we propose a comprehensive device discovery solution.

Acknowledgements

This research results from the SPLICE research program, supported by a collaborative award from the SaTC Frontiers program at the the National Science Foundation under award numbers CNS-1955805, CNS-1955172, CNS-1955228, and CNS-1955231. The work was also supported by the VeChain Foundation and the Dartmouth College and American University of Kuwait (Dartmouth-AUK) Fellowship program. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors. Any mention of specific companies or products does not imply any endorsement by the authors, by their employers, or by the sponsors.

References

[1] R. M.S., S. Pattar, R. Buyya, V. K.R., S. Iyengar, and L. Patnaik, "Social Internet of Things (SIoT): Foundations, thrust areas, systematic review and future directions," *Computer Communications*, vol. 139, pp. 32–57, 2019, DOI https://doi.org/10.1016/j.comcom.2019.03.009.

[2] M. Shafiq, P. Singh, I. Ashraf, M. Ahmad, A. Ali, A. Irshad, M. Khalil Afzal, and J.-G. Choi, "Ranked sense multiple access control protocol for multichannel cognitive radio-based IoT networks," *Sensors*, vol. 19, no. 7, 2019, DOI 10.3390/s19071703.

[3] Y. Meidan, M. Bohadana, A. Shabtai, J. D. Guarnizo, M. Ochoa, N. O. Tippenhauer, and Y. Elovici, "ProfilIoT," in *Proceedings of the Symposium on Applied Computing*. ACM, Apr. 2017, DOI 10.1145/3019612.3019878.

[4] K. Komando. How to check for hidden cameras in Airbnb, VRBO, or vacation rentals. Accessed: 2023-01-22. Online at https://www.usatoday.com/story/tech/columnist/komando/2022/06/23/how-check-hidden-cameras-airbnb-vrbo-vacation-rentals/7652726001/.

[5] FIDO Alliance. How FIDO works. Accessed: 2023-01-22. Online at https://fidoalliance.org/how-fido-works/.

[6] FIDO Alliance. Accessed: 2023-01-24. Online at https://fidoalliance.org.

[7] E. Lear, R. Droms, and D. Romascanu, "Manufacturer Usage Description Specification," IETF, RFC 8520, Mar. 2019, Accessed: 2023-01-24. Online at http://tools.ietf.org/rfc/rfc8520.txt.

[8] L. Lhotka, "JSON Encoding of Data Modeled with YANG," IETF, RFC 7951, Aug. 2016, Accessed: 2023-01-24. Online at http://tools.ietf.org/rfc/rfc7951.txt.

[9] R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, "Network Configuration Protocol (NETCONF)," Tech. Rep., June 2011, DOI 10.17487/rfc6241.

[10] T. Scheffler and O. Bonneß, "Manage resource-constrained IoT devices through dynamically generated and deployed YANG models," in *Proceedings of the Applied Networking Research Workshop*. New York, NY, USA: Association for Computing Machinery, 2017, p. 42–47, DOI 10.1145/3106328.3106331.

[11] K. Seklou, P. Kokkinos, N. D. Tselikas, and A. C. Boukouvalas, "Monitoring and management of home appliances with NETCONF and YANG," in *Proceedings of the Pan-Hellenic Conference on Informatics*. Association for Computing Machinery, 2019, p. 25–32, DOI 10.1145/3368640.3368643.

[12] C. S. Alliance. (2023) Build with Matter Smart Home Device Solution - CSA-IOT. Online at https://csa-iot.org/all-solutions/matter/.

[13] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.

[14] B. Perez, G. Mazzaro, T. J. Pierson, and D. Kotz, "Detecting the Presence of Electronic Devices in Smart Homes Using Harmonic Radar Technology," *Remote Sensing 2022, Vol. 14, Page 327*, vol. 14, no. 2, p. 327, Jan. 2022, DOI 10.3390/RS14020327.

[15] B. Perez, T. J. Pierson, G. Mazzaro, and D. Kotz, "Identification and classification of electronic devices using harmonic radar," in *International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. IEEE, 2023, pp. 248–255.