Towards Examining The Security Cost of Inexpensive Smart Home IoT Devices

TJ OConnor
Florida Institute of Technology
Melbourne, FL, USA
toconnor@fit.edu

Dylan Jessee
Florida Institute of Technology
Melbourne, FL, USA
djessee2020@my.fit.edu

Daniel Campos
Florida Institute of Technology
Melbourne, FL, USA
dcampos2015@my.fit.edu

***Abstract*—A myriad of security challenges has accompanied the rapid proliferation of internet-of-things (IoT) smart-home devices. While smart-home security cameras, locks, digital speakers, and thermostats offer the promise of security, their naive implementations often introduce vulnerability into our digitally connected lives. We argue that the consumer demand for inexpensive IoT has led to a supply of grossly insecure devices. To examine this hypothesis, we examine the security of five inexpensive IoT devices from three separate vendors. In all five devices, our work uncovers immature software security efforts. Our findings discover new vulnerabilities, document legacy vulnerabilities due to software bill of materials (SBOM) issues, explore security mitigations in firmware, and examine the unsecured communication within the ecosystems of the devices. Our analysis discusses the root causes of these vulnerabilities. While these results indicate a snapshot of an immature and naive state of IoT software, there are several software development lifecycle processes that vendors can immediately implement to overcome the root causes of these vulnerabilities.**

***Index Terms*—internet of things, security and privacy, secure software development**

I. Introduction

Security remains the greatest obstacle to the adoption of IoT in consumer homes. While internet-connected cameras allow us the promise of security, they also introduce the vulnerability that an attacker will access and record our most personal moments. Attackers can leverage smart doorbells to surveil the activities of unsuspecting victims instead of preventing package thieves [1], [2]. Unfortunately, there is also a growing history of IoT technology-facilitated abuse and stalking [3]–[5]. Further, intelligence agencies have developed capabilities around exploiting IoT devices and their connected sensors [6]–[9]. In 2017, WikiLeaks revealed a joint venture by the CIA and MI5 intelligence agencies that developed a tool that targeted the microphone on Samsung Smart TVs [6]. Two years later, the US National Defense Authorization Act (NDAA) included a provision that banned IoT security cameras from Hikvision and Dahua, citing concerns of state-sponsored mass espionage by the vendors [7]. Researchers further revealed the presence of firmware backdoors that enabled attackers to use the camera systems for covert and surreptitious spying [8]. As Hikvision and Dahua also provide turnkey solutions for IoT development, the backdoor propagated to dozens of other original equipment manufacturer (OEM) vendors, corrupting their products' software supply chain. In 2019, the US government also grounded eight hundred unmanned aerial drones after the US Cybersecurity and Infrastructure Security Agency raised security concerns [9].

The immature state of IoT further enables these attacks and is growing worse while the demand for inexpensive IoT devices grows. A 2019 study by Cyber ITL examined 1,294 devices' firmware and identified that firmware updates were more likely to remove security hardening features than add them [10]. We argue that this omnipresent lack of security is a result of the consumer demand for inexpensive devices. Vendors, faced with sub-$100 device markets, fail to consider the cost of security and fail to incorporate security into their software development lifecycle. To support this hypothesis, we examine a sample of five inexpensive IoT devices purchased from retail stores. We extract and analyze firmware to assess the security mitigations, identify vulnerabilities due to the software bill of materials (SBOM), and develop novel vulnerabilities. This paper makes the following contributions:

1) We examine the security of five IoT devices from three separate vendors, uncovering immature software security efforts that leverage turnkey platforms, legacy binaries, and fail to implement binary hardening protections.
2) We uncover seven new vulnerabilities to examine the immature and naive state of IoT software stemming from issues with improper access control, use of hard-coded credentials, and missing encryption of sensitive data.

Organization: Section II investigates previous work related to the security and privacy of smarthome IoT devices. In Section III, we provide an overview of our approach to examine the security cost of inexpensive IoT devices. Section IV examines our experiment and the results of our experimental evaluation. Section V offers insight and examines future challenges. Section VI summarizes our conclusions.

II. Prior Work

A growing body of work exists on the security and privacy of IoT devices [11]–[19]. The distributed and constantly changing attack surface of IoT presents a complex and poorly understood problem [16]. To examine an understanding of threats, Kent et al. examined research from 234 organizations that provide cybersecurity guidance [20]. Their work identified a lack of clear explanation of the threats. Emami-Naeini

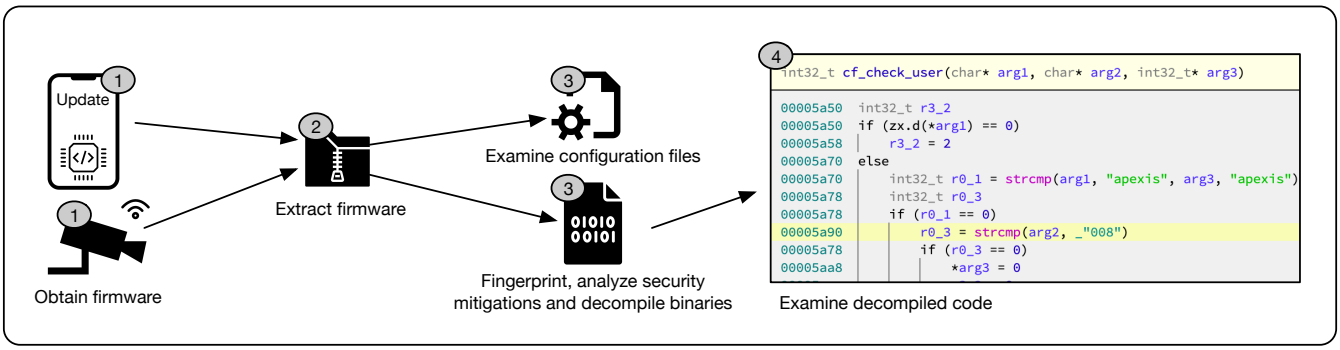

Fig. 1. Our work analyzed the binaries from the extracted firmware of five inexpensive IoT devices, uncovering issues with improper access control, use of hard-coded credentials, and a lack of encryption.

et al. interviewed twenty-four participants about recent IoT purchases [21]. Further, they conducted an MTurk survey of 200 participants to examine the participants' concerns for the security and privacy of IoT devices. They discovered that half of the participants had limited or incorrect knowledge of security and confidentiality, impacting their ability to make educated decisions [21]. A lack of understanding of the attack surface propagates into guidance as well. A 2017 Department of Justice guidance memo suggests that users should install anti-virus on their IoT devices [22]. Such naive advice ignores the constraint that most consumer IoT devices are resource-constrained embedded devices without access to the operating system. Attacks against IoT devices take many forms across the IoT architecture at services running on the device [23], the mobile application [24], the cloud or content distribution network provider [15], or the network transport channels [13], [25]–[27]. This broad scope presents a difficulty for customers trying to analyze the security and privacy of their devices. To this end, Emami-Naeini et al. proposed security labels, identifying the factors that experts believed are important for consumers purchasing IoT devices [28]. Their work identified 12 primary and 13 secondary layer factors recommended by most experts. At the primary layer, experts focused on previous assessments, security updates, the type of sensors and data collected, cryptography, and default passwords [28]. We use this work to motivate the factors examined in our experimental evaluation.

III. Overview

Figure 1 provides an overview of our methodology for our experiment examining the security of five low-cost IoT devices. Our analysis examined the device firmware to assess the security of the devices. The firmware contains the kernel, bootloader, filesystem, applications, and configurations. In our experiments, (1) we first obtained the compressed firmware images from various methods we discuss in Section IV, including reading directly from the device flash memory, intercepting unencrypted traffic with the companion application, or simply logging into an unauthenticated telnet service. We then (2) extracted and explored the filesystem from the compressed firmware image using *BinWalk* utility. Next, we (3) explored configuration files and binaries for security and privacy weaknesses. We fingerprinted operating system binaries to determine the BusyBox version, analyzed the exploit mitigations (non-executable stack, address space layout randomization, stack canaries, and relocated read-only sections). Finally, we (4) decompiled interesting binary files for analysis to discovered unintended functionality or vulnerabilities in the various devices.

IV. Experiment

A. Devices Tested

Table I contains a list of the devices we examined in our experiments. We focus our investigative efforts on security doorbells and privacy cameras due to their increasing popularity and adoption in the home market. We selected devices available in the last two years that were sold in brick-and-mortar Walmart or Best Buy stores. We focus our efforts on devices with a cost under $100, where the vendor focuses their efforts on gaining revenue off the annual subscriptions.

B. Obtaining the Firmware

Companion App Traffic: We obtained the NightOWL doorbell and camera firmware by intercepting message traffic between the companion iOS application and NightOWL servers. Since the companion application did not enforce certificate pinning, we used *Mitmproxy* to generate a spoofed certificate and intercept traffic. When we examined the message traffic, we observed JSON messages containing the version and URL of the latest firmware images hosted on Amazon AWS cloud servers. We then downloaded the firmware.

Serial Peripheral Interface (SPI) Flash: We obtained the Geeni Doorbell and Camera firmware from the Serial Peripheral Interface (SPI) of the devices' chipsets. To extract the firmware, we wired an *Attify badge* wired to the chip's ground, clock, input, output, and chip-select pins and issued an SPI read transaction using the *flashrom* utility. The resulting bitstream contained the contents of the firmware image, including the kernel and filesystems for the devices.

Raw Copy from Telnet Access: We obtained the Kangaroo Privacy Camera firmware by copying the raw contents of

TABLE I
OUR ANALYSIS UNCOVERED THAT THE DEVICES' FIRMWARE RELIED ON VULNERABLE LEGACY ENVIRONMENTS AND TURNKEY PLATFORMS AND FAILED TO IMPLEMENT BEST PRACTICES FOR EXPLOIT MITIGATION.

Vendor	Device	F/W Obtained	F/W Version	BusyBox Version	Binary Hardening	Device Cost	Subscription Cost
Kangaroo	Privacy Camera (A0006)	Telnet Copy	1.09	1.20.2 (2012)	55%	$69.00	$98.99
NightOwl	Doorbell (WDB20V2)	Companion App	20190314	1.20.2 (2012)	53%	$59.99	$48.00
NightOwl	Camera (WNIP2LTABS)	Companion App	20201201	1.19.3 (2011)	47%	$79.99	$48.00
Geeni	Doorbell (GNCCW013)	SPI Flash	1.8.1	1.21.1 (2013)	42%	$99.99	n/a
Geeni	Camera (GNCCW028)	SPI Flash	2.7.2	1.20.2 (2012)	56%	$79.99	$19.99

the filesystem from an unauthenticated telnet session. The camera, built on a turnkey platform, contained a telnet server that granted a root shell to an unauthenticated user. We used this access to login into the device and copy the memory technology device (mtd) special devices, which pointed to the kernel and filesystem images.

C. Software Bill of Materials (SBOM) Issues

Developers rarely build IoT firmware from scratch. Instead, IoT firmware development often relies on integrating open-source or turnkey development frameworks, libraries, and binaries into device-specific solutions. This opaque process often obscures the software bill of materials (SBOM) that constitutes each firmware release. Our observations uncovered relatively new firmware releases built on legacy development frameworks, libraries, and binaries. As an indication of this problem, Table I lists *BusyBox* version (and year of the version release) we discovered in each firmware. The BusyBox environment provides lightweight operating system utilities for embedded environments. However, relying on legacy BusyBox builds introduces substantial risk. For example, researchers have documented fifteen common vulnerabilities and exposures (CVEs) in the 1.19.3 BusyBox version included in the NightOwl camera firmware. The reliance on turnkey platforms that we discuss in Section V, explains the lack of current binaries and libraries. Rather than building the entire software bill of materials, vendors modify pre-packaged turnkey platforms to construct specific solutions for their IoT devices.

D. Lack of Binary Hardening

We analyzed the binary hardening features of each firmware's executables. This analysis helps us uncover the maturity of security in firmware. Binary hardening complicates exploit development by including compile-time optimizations that introduce exploit mitigations. Specifically, we examined the use of no execute (NX) stacks, position independent executables (PIE), relocation read-only (RELRO), stack canaries, and binary stripping. In Table I, we present the overall binary hardening coverage in each firmware. These results echo findings from [10] that examined binary hardening across a broad set of 1,294 IoT devices. The lack of secure coding practices in IoT magnifies the importance of compile-time hardening features. As an example, the Kangaroo privacy camera's main application, *encoder*, calls the unsafe textitstrcpy function over 200 times. The developers compiled the application without support for position-independent-executable code, stack canaries, or full relocation read-only (RELRO). By failing to incorporate these hardening mechanisms, the developers greatly simplify the ability for attackers to develop working exploits. It is unclear why a developer would choose to ignore such mechanisms that the gcc compiler turns on by default. However, this again echos the findings of [10] that identified developers are actually getting worse about incorporating binary hardening into IoT executables.

E. Reported Vulnerabilities

After extracting the firmware and examining the filesystem and applications, we discovered seven unique vulnerabilities. We reported each of the vulnerabilities to both the affected vendor and MITRE. MITRE validated and assigned a CVE to each of the following vulnerabilities. These vulnerabilities focused on common weaknesses including improper access control, use of hard-coded credentials, and missing encryption of sensitive data.

- *CVE-2021-33559*: The Kangaroo Privacy Camera contains a telnet server with unauthenticated root access.
- *CVE-2021-31793*: The NightOWL Doorbell contains a webserver that allows an unauthenticated user to gain access to snapshots and video stream.
- *CVE-2020-28713*: The NightOWL Doorbell communicates events and content over an unencrypted HTTP connection, leading to the ability for an attacker to falsely report events and notifications.
- *CVE-2020-28998*: The Geeni DoorScreen Doorbell contains a telnet server with hardcoded credentials.
- *CVE-2020-28999*: The Geeni DoorScreen Doorbell contains a streaming server with hardcoded credentials.
- *CVE-2020-29000*: The Geeni DoorScreen Doorbell contains a RTSP server that responds to a specially crafted messages that can deliver a telnet session to an attacker.
- *CVE-2020-29001*: The Geeni Camera contains a REST-Ful service with hardcoded credentials. An attacker can abuse this service to enable remote access to the device.

V. DISCUSSION

In the following section, we discuss the systemic issues we observed that allowed us to compromise the security and privacy of the evaluated devices. Specifically, we discuss how a lack of understanding the attack surface, relying on turnkey frameworks, ignoring cryptographic practices and silently discontinuing product lines leads an abysmal state of security.

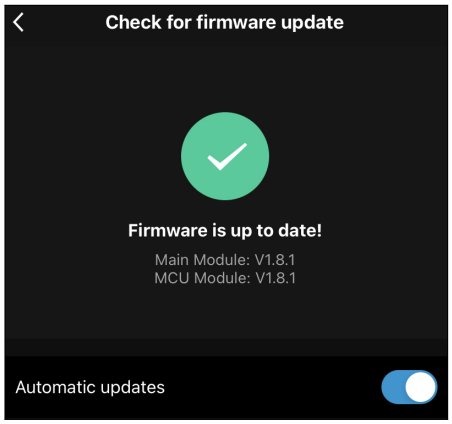

Fig. 2. IoT applications often prevent an opaque state of the devices. This discontinued IoT device provides a confusing message indicating it has received the latest firmware update.

Failing to Map The Complete Attack Surface: Our preliminary results suggest vendors fail to map their attack surfaces with holistic approaches. IoT attack surfaces present a complex surface consisting of integrated hardware, firmware, companion applications, and a distributed architecture of networks. An understanding of attacker motivations and goals drives the development of attack surface models. However, vendors struggle to realize the attack surface even when they are capable of realizing attacker motivations and threat models. Consider the issue of the *Kangaroo Privacy Camera* to illustrate this issue. The vendor designed the camera with a *privacy mode*, which activates a polymer-dispersed liquid crystal lens that turns opaque. This privacy-focused design feature provides the user with assurance that the camera cannot record any imagery when placed into this mode. This well-thought addition suggests the vendor realized that internet-connected cameras present privacy threats. However, the same vendor enabled an unauthenticated telnet service on the device firmware. We reported this issue as CVE-2021-33559. After reverse-engineering the device's firmware, we discovered we could manipulate the state of the privacy lens of the device by writing a value to a general-purpose input/output (GPIO) pin. Although the vendor implemented a hardware mechanism with the best intentions to provide privacy, a separate firmware issue nullified any benefit the hardware provides.

Failing to Implement Cryptography: Examining the network captures of the devices, we identified that NightOwl devices failed to encrypt most traffic. In particular, we observed the NightOwl doorbell pushed event notifications and content over unencrypted HTTP requests. By modifying the request's parameters, we could falsely report events, such as ringing the doorbell of any NightOwl device. We reported this finding as CVE-2020-28713. This finding was not surprising as Sivanathan et al. identified 65% of IoT device traffic is unencrypted [29]. The relatively naive nature of IoT does present a problem in establishing the architecture of an encrypted communication channel [30]. Encryption certificates must be purchased, distributed, and enforced. However, NightOwl is a 12-year old vendor with a 2019 patent for the doorbell. We observed similar problems in the companion applications of the NightOwl companion application. Although the companion application encrypted traffic, it did not properly pin encryption certificates. This vulnerability allowed us to intercept the NightOwl companion app traffic, which included the URLs for the NightOwl firmware. We saw similar certificate pinning failures in the NightOwl device. Our findings echo previous work [17], [24] that uncovered the majority of IoT vendors fail to implement certificate pinning.

Silently Discontinued Product Lines: A lesser-discussed issue of IoT is the silent abandoning of product lines. End-of-life issues are common in software. For example, Microsoft shutdown security updates for the Windows XP operating system after 12 years of use. While consumers can usually overcome the threat of software end-of-life by purchasing new software versions, firmware/hardware presents a greater challenge [31]. When an IoT vendor discontinues firmware updates for a product, a consumer will most likely need to procure a new product. However, that is predicated upon the consumer identifying that the product has reached end-of-life. In response to notifying Geeni of the vulnerabilities in their DoorScreen doorbell, the vendor acknowledged that the *"model that has been discontinued for some time and represents less than 0.1% of our active devices."*. However, examining the companion application provides no insight that the device has reached end-of-life. The companion application, depicted in Figure 2 would suggest that the consumer is still receiving automatic updates.

Relying on TurnKey Frameworks: In examining the device firmware, we uncovered the Geeni and Kangaroo vendors leveraged *turnkey* frameworks to construct their doorbell and privacy camera's firmware. These third-party or turnkey providers offer complete solutions that provide the required infrastructure and hardware components for an IoT ecosystem. They offer the app SDK, Cloud API, and software libraries to rapidly develop IoT ecosystems. Turnkey provider solutions reduce technical requirements for entering the IoT market by abstracting away development and allowing vendors to focus on manufacturing and deploying new products. The Geeni DoorScreen developers build the firmware on top of an Apexis turnkey platform (Apexis APM-H803-MPC). Although MITRE published a high-risk vulnerability for this platform in 2017, the developers adopted the turnkey firmware framework without mitigations to prevent the existing vulnerabilities. We subsequently reported three new Common Vulnerabilities and Exposures (CVEs) caused by integrating the Apexis platform on the Geeni DoorScreen doorbell (including CVE 2020-28999, 2020-29000, and 2020-28998). The highest risk CVE identified that the Apexis platform contained a streaming video application with hard-coded credentials. After identifying this vulnerability, we uncovered that the vulnerability propagated to several other IoT vendors that relied on the Apexis turnkey framework. Examining the application version string for the

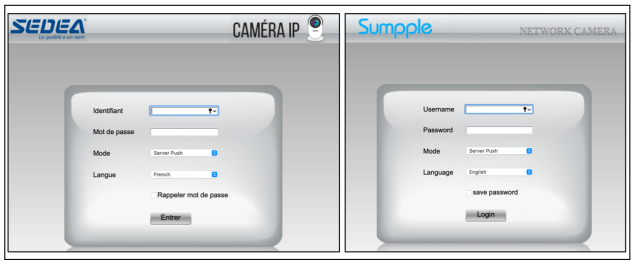

Fig. 3. The Geeni, Sedea, and Sumpple vendors all leveraged the same turnkey firmware to produce their products, inheriting an application with hardcoded backdoor credentials.

application on Shodan, we observed 4,413 internet-facing vulnerable cameras running the same Apexis backdoored application. Figure 3 depicts two other vendors (Sedea, and Sumpple) that separately adopted the Apexis firmware and its inherent vulnerabilities.

VI. Conclusion

In this work, we hypothesized that the low cost of IoT devices has led to a supply of IoT devices lacking security. To initially explore this hypothesis, we sampled and analyzed the security of five inexpensive IoT devices from three separate vendors. Our work uncovered immature software security efforts across all three vendors. We documented persistent SBOM issues, identifying all the devices that relied on eight-year-old software suites. Next, our findings discovered new vulnerabilities that exploited common weaknesses, including improper access control, hard-coded credentials, and missing encryption of sensitive data. We discussed the root causes of these problems, including failing to map the entire attack surface, ignoring best cryptographic practices, and relying on turnkey solutions for firmware development. While smart home IoT devices introduce convenience, there is a long road ahead until they also deliver security and privacy.

Acknowledgements

This material is based upon work supported in whole or in part with funding from the Office of Naval Research (ONR) contract #N00014-21-1-2732. Any opinions, findings, conclusions, or recom- mendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the ONR and/or any agency or entity of the United States Government.

References

[1] "Police can get your Ring doorbell footage without a warrant, report says — Ars Technica." [Online]. Available: https://arstechnica.com/tech-policy/2019/08/police-can-get-your-ring-doorbell-footage-without-a-warrant-report-says/

[2] "Amazon's Ring is the largest civilian surveillance network the US has ever seen — Lauren Bridges — The Guardian." [Online]. Available: https://www.theguardian.com/commentisfree/2021/may/18/amazon-ring-largest-civilian-surveillance-network-us

[3] N. Bowles, "Thermostats, locks and lights: Digital tools of domestic abuse," https://nyti.ms/2KdGgVC, 2018.

[4] D. Freed, J. Palmer, D. Minchala, K. Levy, T. Ristenpart, and N. Dell, "A stalker's paradise: How intimate partner abusers exploit technology," in *Conference on Human Factors in Computing Systems (CHI)*. Montreal, Canada: ACM, 2018, pp. 1–13.

[5] R. Leitão, "Anticipating smart home security and privacy threats with survivors of intimate partner abuse," in *Designing Interactive Systems Conference*. San Diego, CA: ACM, 2019, pp. 527–539.

[6] T. Fox-Brewster, "Here's how the cia allegedly hacked samsung smart tvs – and how to protect yourself," https://www.forbes.com/sites/thomasbrewster/2017/03/07/cia-wikileaks-samsung-smart-tv-hack-security, Mar. 2017.

[7] M. Thornberry, "H.R.5515 - 115th congress (2017-2018): John S. Mccain National Defense Authorization Act for Fiscal Year 2019," https://www.congress.gov/bill/115th-congress/house-bill/5515/text, Aug 2018.

[8] A. Kharpal, "China's surveillance tech is spreading globally, raising concerns about beijing's influence," *CNBC. October*, vol. 8, 2019.

[9] J. Valente and A. A. Cardenas, "Understanding security threats in consumer drones through the lens of the discovery quadcopter family," in *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy*, 2017, pp. 31–36.

[10] P. Thompson, "Binary hardening in iot products," https://cyber-itl.org/2019/08/26/iot-data-writeup.html, Aug 2019.

[11] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis *et al.*, "Understanding the mirai botnet," in *USENIX Security*. Vancouver, Canada: USENIX, 2017, pp. 1093–1110.

[12] J. Lau, B. Zimmerman, and F. Schaub, "Alexa, are you listening? privacy perceptions, concerns and privacy-seeking behaviors with smart speakers," *Human-Computer Interaction*, vol. 2, pp. 1–31, 2018.

[13] T. OConnor, W. Enck, and B. Reaves, "Blinded and confused: Uncovering systemic flaws in device telemetry for smart-home internet of things," in *ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)*. Miami,FL: ACM, 2019, pp. 140–150.

[14] J. Choi, A. Anwar, H. Alasmary, J. Spaulding, D. Nyang, and A. Mohaisen, "Iot malware ecosystem in the wild: a glimpse into analysis and exposures," in *Symposium on Edge Computing*. Bellevue, WA: ACM, 2019, pp. 413–418.

[15] B. Janes, H. Crawford, and T. OConnor, "Never ending story: Authentication and access control design flaws in shared iot devices," in *Security and Privacy Workshops (SPW)*. San Francisco, CA: IEEE, 2020, pp. 104–109.

[16] B. Yuan, Y. Jia, L. Xing, D. Zhao, X. Wang, and Y. Zhang, "Shattered chain of trust: Understanding security risks in cross-cloud iot access delegation," in *USENIX Security*. Virtual Event: USENIX, 2020, pp. 1183–1200.

[17] T. OConnor, D. Jessee, and D. Campos, "Through the spyglass: Towards iot companion app man-in-the-middle attacks," in *Cyber Security Experimentation and Test Workshop*. Virtual Event: USENIX, 2021.

[18] A. Alhazmi, G. Kilani, W. Allen, and T. OConnor, "A replication study for iot privacy preferences," in *Conference on Omni-Layer Intelligent Systems (COINS)*. Virtual Event: IEEE, August 2021.

[19] A. Alhazmi, K. Alawaji, and T. OConnor, "Mpo: Mqtt-based privacy orchestrator for smart home users," in *Computers, Software, and Applications Conference (COMPSAC)*. Virtual Event: IEEE, July 2022.

[20] S. Turner, J. R. Nurse, and S. Li, "When googling it doesn't work: The challenge of finding security advice for smart home devices," 2021.

[21] P. Emami-Naeini, H. Dixon, Y. Agarwal, and L. F. Cranor, "Exploring how privacy and security factor into iot device purchase behavior," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–12.

[22] D. of Justice Computer Crime & Intellectual Property Section Criminal Division, "Securing your Internet of Things devices," https://www.justice.gov/criminal-ccips/page/file/984001/download, Juyl 2017.

[23] R. Mitev, M. Miettinen, and A.-R. Sadeghi, "Alexa lied to me: Skill-based man-in-the-middle attacks on virtual assistants," in *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*, 2019, pp. 465–478.

[24] H. Fereidooni, T. Frassetto, M. Miettinen, A.-R. Sadeghi, and M. Conti, "Fitness trackers: fit for health but unfit for security and privacy," in *2017 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)*. IEEE, 2017, pp. 19–24.

[25] A. Hariri, N. Giannelos, and B. Arief, "Selective forwarding attack on iot home security kits," in *Computer Security*. Springer, 2019, pp. 360–373.

[26] R. Mitev, A. Pazii, M. Miettinen, W. Enck, and A.-R. Sadeghi, "Leakypick: Iot audio spy detector," in *Annual Computer Security Applications Conference*, 2020, pp. 694–705.

[27] H. Mohajeri Moghaddam, G. Acar, B. Burgess, A. Mathur, D. Y. Huang, N. Feamster, E. W. Felten, P. Mittal, and A. Narayanan, "Watching you watch: The tracking ecosystem of over-the-top tv streaming devices," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 131–147.

[28] P. Emami-Naeini, Y. Agarwal, L. F. Cranor, and H. Hibshi, "Ask the experts: What should be on an iot privacy and security label?" in *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2020, pp. 447–464.

[29] A. Sivanathan, D. Sherratt, H. H. Gharakheili, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman, "Characterizing and classifying iot traffic in smart cities and campuses," in *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2017, pp. 559–564.

[30] D. Campos and T. OConnor, "Towards labeling on-demand iot traffic," in *Cyber Security Experimentation and Test (CSET)*. Virtual Event: USENIX, August 2021.

[31] M. Langheinrich, "Long live the iot," *IEEE Pervasive Computing*, vol. 19, no. 2, pp. 4–7, 2020.