Who Let the Smart Toaster Hack the House? An Investigation into the Security Vulnerabilities of Consumer IoT Devices

Yang Li
University College London
yang-li.22@ucl.ac.uk

Anna Maria Mandalari
University College London
a.mandalari@ucl.ac.uk

Isabel Straw
University College London
isabel.straw.20@ucl.ac.uk

Abstract

For smart homes to be safe homes, they must be designed with security in mind. Yet, despite the widespread proliferation of connected digital technologies in the home environment, there is a lack of research evaluating the security vulnerabilities and potential risks present within these systems. Our research presents a comprehensive methodology for conducting systematic IoT security attacks, intercepting network traffic and evaluating the security risks of smart home devices. We perform hundreds of automated experiments using 11 popular commercial IoT devices when deployed in a testbed, exposed to a series of real deployed attacks (flooding, port scanning and OS scanning). Our findings indicate that these devices are vulnerable to security attacks and our results are relevant to the security research community, device engineers and the users who rely on these technologies in their daily lives.

1 Introduction

In the age of technology, our homes are changing. The rapid development of smart devices and the emergence of the Internet of Things (IoT) is reshaping the environment in which we live and the means by which we carry out our daily lives [1][2]. The IoT has been described as the ubiquitous network of devices which communicate with one another, without human interaction, permeating the infrastructure of our experience [3]. The smart home is an application of an IoT environment, which comprises the physical entities and connectivity present in domestic settings [3]. In the home, the first primitive IoT device was a remotely controllable toaster, introduced in 1990 as a proof-of-concept [4]. Since this time consumer IoT devices for the home have flooded the market, ranging from smart TVs, to connected light bulbs, thermostats and door locks [4]. According to an extrapolation from 2021, the number of IoT devices will rise to 51 billion in 2023 and continue to increase in the foreseeable future, estimated to reach 75 billion by 2025 [4][2].

Security is of paramount concern to smart home users, evidenced by research from Aldossari and Sidorova, who highlighted the relationship between consumer device acceptance, trust and notions of security and privacy [2]. Traditionally in IT security, domain goals have consisted of ensuring confidentiality, integrity and accountability of systems and messages [5]. Yet researchers have illustrated that such frameworks are outdated and fail to account for the evolving risks of IoT systems [6]. IoT systems operate 24/7 and therefore are always available for attacks (e.g. botnet attacks) [7] and the heterogeneous plethora of possible devices present within the system novel security issues [3, 6].

The harm that can result from an attack is dependent on the end-point functionality of the device, which in the smart home encompasses a spectrum of harms ranging from a faulty smart fridge to an unresponsive smoke detector [3]. Previous research has described the means by which smart home design can open up the door to risks that range from exposing the privacy of householders, to facilitating crimes such as burglary using video feeds, to tampering with healthcare appliances to enact physical harm [8].

The rising prevalence of IoT devices results in a growing range of security and privacy risks. Many IoT devices can involuntarily become part of a botnet [9] and may be vulnerable to Denial of Service (DoS) attacks [10]. Other risks include leakage of Personally Identifiable Information (PII) because of lack of encryption or authorisation, misactivation [11], or malware attacks [12]. In this paper, we aim to evaluate the security and privacy risks of consumer IoT devices by developing a methodology for conducting systematic IoT attacks and intercepting the devices' network traffic. We use our large-scale IoT testbed, along with several Raspberry Pi 4s (RPi 4), to launch over 462 automated experiments against 11 IoT devices. The security attacks, privacy threats, and vulnerabilities that we evaluate include network attacks (e.g., port scanning, flooding, *etc.*).

Surprisingly, we find that IoT devices are indeed vulnerable to well-known and documented IoT security attacks.

Our key research contributions include the following:

- We develop an automated methodology for evaluating security vulnerabilities in common consumer IoT devices using large-scale, diverse experiments and sets of attacks;

- We assess the security vulnerabilities of popular IoT devices against existing network and device attacks, and identify privacy risks.

In summary, we find that consumer IoT devices are highly vulnerable to common IoT security attacks. We argue for increased security and privacy in this space, given the risks for the users when IoT devices are compromised.

We make our experiment software and datasets available at https://github.com/SafeNetIoT/spices.

2 Assumptions

In this section, we summarize the threat model and goals of this work.

2.1 Threat Model

We consider the following threat model.

Adversary. The adversary is any party that can access the internal IoT device network, such as malicious IoT devices.

Victim. The victim is any person in a smart home that owns an IoT device in a smart home.

Threat. We assume the presence of malicious or compromised IoT devices in a smart home. The malicious device has access to the home router. Adversaries may be incentivized to compromise other devices in the network for inferring user activities or denying the usage of them. We consider security threats (Mirai, Scan, *etc*. [13, 14]).

2.2 Goals and Non-Goals

The main goal of this work is to analyze the reaction of consumer IoT devices to common security threats. In particular, this work answers the following research questions (RQ):

RQ1. Are consumer IoT devices vulnerable to common security attacks? Our goal is to characterize how IoT devices react to security attacks. To address this, we propose a testbed for systematically studying their reaction and capturing their network traffic.

RQ2. Do the IoT devices detect threats? IoT devices may have security protection techniques in place and notify the user or manufacturer when detecting security threats. We check their capability to do so.

Non-Goals. In this initial study, we do not consider the following as goals, and leave them for future work.

No control over how an IoT device works internally. We consider the IoT device as a blackbox, we do not have control over how an IoT device works internally. However, we have the capability to interact with them using their companion app, and we can track their network activity.

We do not test all threats. Our methodology only focuses on a subset of security threats for every IoT device, so that we can cover the same threats for different devices.

Consumer IoT devices. We focus on IoT devices that target consumers; we do not consider medical or industrial IoT devices.

3 Testbed

In order to have a controlled environment for threat emulation, we build the testbed shown in Figure 1. The testbed consists of: (*i*) a *gateway* that provides IP connectivity to the IoT devices from the Internet and has the capability of capturing all the network traffic of the IoT devices; (*ii*) the *Attacker*, an RPi which acts as an IoT device in the network,
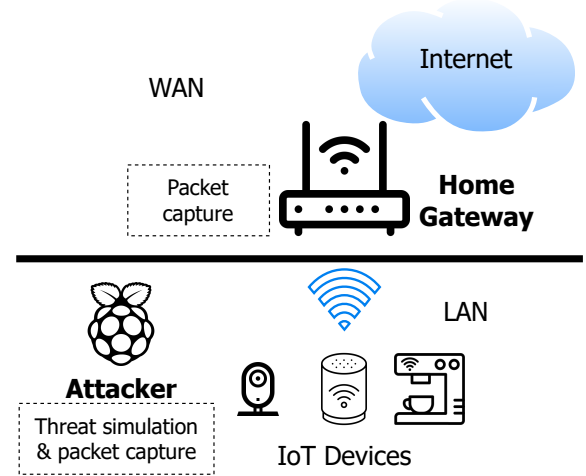

Figure 1. Overview of the testbed.

(*iii*) the *IoT devices under test*, a group of popular IoT devices all connected to the gateway; (*iv*) *threat scripts*, run at the attacker to execute IoT security threats experiments. More details on each component are presented below.

3.1 Gateway

The gateway is configured using a NAT setup. It has two network interfaces, a WAN interface with a public IPv4 address, and a LAN interface with a private IP address, used to give NAT Internet connectivity to the IoT devices and the attacker. The gateway has DHCP capabilities, effectively trying to mimic the typical configuration of a smart home network. The gateway is also capable of intercepting all the network traffic from the IoT devices and the attacker using tcpdump. The gateway is physically connected to an Android phone via the Android Debug Bridge, and has the ability to control the IoT devices through their Android companion app. Each device's traffic is filtered by MAC address into separate files.

3.2 Attacker

This component is an RPi, with one network interface (Wi-Fi) connected to the gateway, where all the IoT devices under test are connected. The attacker is responsible for running the *threat scripts* to simulate threats originating from an IoT device in the LAN network.

3.3 IoT Devices

Table 1 shows the IoT devices we consider. We examine 11 consumer IoT devices typically deployed in a smart home. We select these devices to provide diversity within different categories and among the most popular ones we could find on the market. We choose devices in 4 categories: smart speakers (3), smart doorbells (2), smart cameras (3), and appliances (3). To better represent how IoT devices behave, we try to keep their default configuration and privacy settings unaltered, and we do not perform user-initiated firmware upgrades. All the IoT devices are connected to the Wi-Fi interface of the gateway, making them part of the LAN where the attacker is located (i.e., their private IP addresses and DNS

Table 1. IoT devices.

Category	Device
Smart speaker	Bose Smart Speaker 500
	Sonos One (Gen2)
	Echo Dot 5
Smart doorbell	Ring Chime Pro
	Ring Video Doorbell (2nd Gen)
Smart camera	Google Nest Cam
	SimpliSafe Security Camera Indoor
	Furbo 360° Dog Camera
Appliances	WeeKett Smart Wi-Fi Kettle
	Govee Alexa LED Strip Lights
	Sensibo Sky Smart AC

Table 2. Devices and flooding attacks (Successful attack: ✓, Unsuccessful attack: ✗).

Devices	SYN	UDP	DNS	Frag. IP
Bose Speaker	✓	✗	✗	✓
Sonos One (Gen2)	✗	✗	✗	✓
Echo Dot 5	✗	✗	✗	✓
Ring Chime Pro	✗	✗	✗	✓
Ring Doorbell	✗	✗	✓	✓
Google Nest Cam	✗	✗	✗	✓
SimpliSafe Cam	✗	✗	✗	✓
Furbo Camera	✗	✗	✗	✓
WeeKett Kettle	✗	✓	✓	✓
Govee Lights	✗	✗	✓	✗
Sensibo Sky	✗	✓	✓	✓

servers are assigned by the DHCP server of the gateway). Their network traffic is captured by the gateway.

3.4 Threat Scripts

We simulate security threats programmatically using *threat simulation scripts*, which, depending on the type of threat, are run on the attacker (threats originating targeting the IoT devices under test). We generate three threats involving Denial of Service (DoS), port scanning, and OS scanning. We make our threat scripts available at `https://github.com/SafeNetIoT/spices`.

4 Methodology

In this section, we report the methodology we use for answering our research questions. We propose an experimental setup that detects the IoT devices' traffic and semi-automatically classifies our attacks as successful or not.

We evaluate devices' defensive measures against various attacks and compare within and across categories.

4.1 Assessing Device Vulnerability

4.1.1 Attacks Definition

We define a list of attacks that can be simulated in a testing environment to assess devices' vulnerabilities. The attacks include SYN (port 80), UDP, DNS, and fragmented IP flooding, as well as port scanning and OS scanning [15]. The flooding attacks and scanning attacks are implemented in separate and configurable scripts. We run the scripts on an RPi 4, connected to the same LAN as the IoT devices under attack. Our threat script uses Nmap [16] to launch port and OS scan attacks and tcpreplay [17] for flooding attacks. Each type of flooding attack is repeated continuously ten times to allow sufficient time for the device to detect and mitigate such an attack. We also check for inconsistency in device behaviours across repeated attacks.

4.1.2 Attacks Validation

In order to assess whether the attacks are able to reach their targets (IoT devices) as expected, we set up a second RPi 4 connected to the same LAN as the first one. We conduct attacks targeting the second RPi and perform packet capture on it. We then verify that simulated attacks reach the second RPi thus validating our experiment.

4.2 Assessing Device Reaction

We run tcpdump continuously on the gateway to capture network packets for all devices connected. All flooding and scanning network traffic is also captured and separated into different folders per device.

We then use Tshark [18] to analyse the packet captures. We implement detection scripts for filtering all the packets coming from the attacker. By applying the filter, we are able to intercept the corresponding replies to the simulated attacks.

We determine whether an attack is successful by analysing the target device's reaction. If the device implements countermeasures that detect ongoing attacks and mitigate the consequences of the attacks, the attack is considered to be unsuccessful, even if the device's normal functionality is interrupted (e.g. interrupted video streaming to its companion application). However, the attack is considered successful if no defensive measures can be observed on the target device's captured traffic and the device's normal activity is halted.

5 Evaluation

We now answer our research questions by identifying and characterizing the reaction of IoT devices to security threats.

5.1 Flooding

Table 2 shows the (un)successful rate of attacks for each device. During SYN flooding, Bose Smart Speaker 500 replies to the SYN packets with SYN/ACK packets. Other devices, except Echo Dot 5, reply to every SYN packet with RST/ACK. Among all devices, Bose Smart Speaker 500 performs the worst in SYN DoS attacks as it replies to inbound SYN with SYN/ACK packets, which would hold the corresponding communication ports half-open, consuming the most resources and making the device stop working. On the contrary, other devices, excluding Echo Dot 5, defend themselves against SYN flooding by resetting those half-open connections, reducing unnecessary resource consumption caused by the attack.

In UDP flooding, Sensibo Sky Smart AC and the Weekett kettle reply with ICMP port unreachable packets with significant delay. Other devices, excluding Echo Dot 5, only reply to a fraction of messages with significant delay. The ICMP port unreachable messages are error messages indicating that the requested UDP port is unavailable or closed [19]. Due to the connectionless nature of the UDP protocol, UDP flooding can successfully render a device unusable without estab-

Table 3. Devices and identified open ports (filtered ports are reported in green).

Device	Identified Open Ports
Bose Speaker	80/7000/8082/8083/8085 /8091/8200/30030/40002 /40031/40035
Sonos One	1400/1410/1443/1843/7000
Echo Dot 5	1080/4070/8888/55442/55443
Ring Chime Pro	847/1003/1020/1393/3736/7240 /8173/12302/15986/16891 /17704/17944/17993/18682/20307 /21257/23825/24669/25781/25958 /25997/26757/27234/28363/29161 /32466/33377/33544/33616/33862 /35470/38657/44100/46108/46194 /47199/50852/51212/52663/54739 /55524/55530/56621/65488
Ring Doorbell	Blocking ping probes & none found
Google Nest Cam	8012/10101/11095
SimpliSafe Cam	19531
Furbo Camera	None found
WeeKett Kettle	6668
Govee Lights	None found
Sensibo Sky	None found

Table 4. Devices and identified Operating Systems (OS).

Device	Operating System
Bose Speaker	Linux 3.2 - 4.9
Sonos One (Gen2)	Linux 3.2 - 4.9
Echo Dot 5	No exact match, can be Linux
Ring Chime Pro	Too many fingerprints match
Ring Doorbell	2N Helios IP VoIP doorbell (95%)
Google Nest Cam	Too many fingerprints match
SimpliSafe Cam	Too many fingerprints match
Furbo Camera	Too many fingerprints match
WeeKett Kettle	No exact OS matches
Govee Lights	Espressif esp8266 firmware (lwIP stack), NodeMCU firmware (lwIP stack)
Sensibo Sky	Philips Hue Bridge (lwIP 1.4.1), Philips Hue Bridge (lwIP stack)

lishing two-way conversations. Hence, all devices perform more or less the same against UDP flooding attacks, as their normal communications are reduced or halted when attacks are in progress.

During DNS flooding, Ring Video Doorbell (2nd Gen), the Weekett kettle, Govee Alexa LED Strip Lights, and Sensibo Sky Smart AC also reply with ICMP port unreachable messages. The rest of the devices reply sparsely, not including the Echo Dot 5. This can be due to their limited resources or designed defensive mechanisms to mitigate the effects of DNS flooding attacks. We conclude that it is challenging to assess devices' performance under DNS DoS attacks without having access to the devices' source code.

Under the IP fragmentation attack, none of the tested devices responds, except the Govee Alexa LED Strip Lights, which replies with an ICMP message stating Time-to-live exceeded (Fragment assembly time exceeded). This indicates that the Strip Light is designed to discard or drop the fragmented packets when it takes too long to assemble them into a complete IP packet. Other devices might have different defensive designs that do not involve sending such ICMP packets.

The Bose speaker still sends application data during SYN, UDP, and DNS flooding but stops working during the IP fragmentation attack. The Ring Chime Pro pings its server during those flooding but also stops working during the IP fragmentation attack. The devices could be sending those messages to potentially report the ongoing attacks or seek assistance during flooding events. Other devices' normal communications with their server are interrupted during the flooding. After the attack, they resume communicating with their servers.

Echo Dot 5 does not respond to any of the attacking pack-

ets, which may indicate better security practices. No inconsistency can be identified between repeated attacks.

5.2 Port Scanning

The port scanning results identify no open ports on Furbo Camera, Govee Lights, and Sensibo Sky. The Ring Doorbell blocks the ping probes, so in this case, the attack is not successful. Various numbers of open ports are identified on the rest of the devices, as shown in Table 3. Ports 80 to 8200 on the Bose speaker are associated with known services, contrarily to ports 30030-40035. Although the Ring Chime Pro has the largest number of open ports, they are all shown in *filtered* state, meaning Nmap cannot determine whether they are open. The identified open ports on other devices are in *open* state. An open port is actively listening for incoming connections and suggests that a service or application is running on that port. A filtered port indicates that some form of filtering or blocking mechanism is in place, which could indicate the presence of a firewall. It is worth noting that a port that is closed during scanning could open up if an application uses it.

5.3 OS Scanning

All tested smart speaker devices have Linux as OS, as shown in Table 4. Interestingly, the scanning results show that the Ring Video Doorbell is likely to have a similar OS to the 2N Helios IP VoIP doorbell. There are devices whose OSes cannot be identified. Those devices could have defensive mechanisms like network filtering (like the Ring Chime Pro) or obfuscation. If not, it could be due to the limitations of the scanning tool or too many similarities between OSes. The rest two appliances all utilise lightweight IP stacks as they are open-source and resource-efficient.

6 Discussion

Our findings demonstrate vulnerabilities across a range of consumer technologies. We now turn to consider the impact this may have on the user in their lived environment, the limitations of our methodology and ethical considerations.

User Implications. The harm posed by a security threat is contextual to the role of the device in the environment. For example, malfunctioning smart heating systems may be more consequential than a compromised kettle. The security

flaws we demonstrate in lighting systems (LED Strip Lights) and sound systems (e.g. smart speakers) illustrate that adversarial attacks may significantly impact the sensory experience of an occupant in the home. Flooding attacks that result in DoS may render a system unresponsive, for example preventing an occupant from activating their lighting system (an outcome that could be particularly distressing at night and if imposed for criminal intent, e.g. burglary). Further research is needed to explore whether these attacks pose a risk to smart lock systems, which, if successful, could prevent an individual from entering/exiting their property. The manipulation of lighting systems is a heightened concern for photosensitive individuals, such as epilepsy sufferers, who have been identified as at risk of seizures from security attacks on smart home lighting systems [20]. Furthermore, domestic violence researchers have exposed concerning trends in interpersonal abuse, reporting that perpetrators have exploited smart light and sound systems to inflict physical and psychological harm on victims [21, 22]. The additional success of flooding attacks on appliances, such as kettles, illustrated that these simple methods could disrupt the ability of an occupant to use the equipment within their home.

Beyond flooding attacks, our work exposes open ports present within connected systems, raising the question of possible exploits that may be enacted through attacks on these pathways. It is possible that with these ports being open, they may be accessed remotely, allowing an adversary to take control of a device. Unfortunately, we are unable to determine the current use of these ports and the means by which they may be manipulated. We leave this as future work. However, their open state allows us to question the harms that could result from attacks aimed at these targets. In particular, the open ports present in Ring doorbells and smart speakers raise the question of whether adversaries could transmit audio into a living environment and impose incessant sound signals. While we found no open ports in the Smart Air Conditioning (AC) machinery, further research is required to explore the risks of exploitation in the range of these devices.

Privacy and Security Implications. Attacks on sensory systems are likely to be immediately apparent to the occupant who is disturbed by these manipulations of the environment. Other exploits, such as privacy attacks, maybe more surreptitious. For example, we have demonstrated vulnerabilities in smart security cameras. If the video footage from these devices is inconspicuously obtained, the data may be shared elsewhere, resulting in a significant breach of occupant privacy and regulation. The implications of our research should therefore be considered through the General Data Protection Regulation (GDPR) framework, which sets the standard for data protection and privacy in the EU and the European Economic Area. For developers, there is an extent of literature that explores the application of GDPR's governing principles and provisions to IoT infrastructure [23].

The European Guidelines developed for IoT security are relevant to our findings. The framework states that if a port is not being used, that port should be closed, yet our findings demonstrate a plethora of open ports in consumer home technologies for unclear reasons. While ENISA and NIST guidelines [24, 25] have been developed to improve design practices and secure the supply chain of IoT, they are currently not mandatory, and we need a methodology for understanding their compliance. Additional research has proposed solutions at the edge for protecting the user from IoT attacks [6].

Limitations. Our exploration of security vulnerabilities in the smart home infrastructure is constrained by a number of limitations. Firstly, these devices have been examined as black-boxes, in which no attempt has been made to reverse engineer their code or response strategies (as these resources are often unavailable or undocumented). Furthermore, our experiments are limited to 11 devices which form only a small proportion of the vast and ever-growing consumer IoT market. In the evolving IoT space, new devices (with potentially new vulnerabilities) are constantly appearing on the market.

Ethical Considerations. In our experiments, we do not cause any real threat on the Internet. All experiments are contained within our own testbed. When conducting the experiments, we fully respected the ethical guidelines defined by our affiliated organization.

7 Related Work

Many works have assessed the security and privacy risks of consumer IoT devices. Approaches used during assessments include running simulated attacks [26, 27, 15], static source code analysis [28], network traffic interception [11, 29], and binary code reverse engineering [30]. However, their methodology does not allow them to run experiments that assess security threat reactions automatically. Running simulated attacks and network traffic interception were chosen for their scalability regarding the number of devices that can be tested simultaneously and the black-box nature of many devices.

Babun et al. [31] perform an analysis of popular smart home platforms. The authors focus on commercial and open-source platforms, pointing out their limitations when dealing with IoT data and apps. In contrast, our study is about IoT security threats, offering a comprehensive tool for assessing their reactions to common attacks.

8 Conclusion

Detecting security threats on smart home IoT devices is an important ongoing challenge. Commercial IoT devices are appearing in the market and being offered by different vendors, but there has been no insight into how they react to security threats.

In this paper, we took a quantitative approach in auditing some of the IoT devices available in the market, as well as analyzed their reaction to common security threats. We developed a scalable and automated methodology for evaluating the effectiveness of these attacks against known IoT devices. Our evaluations using 3 security threats on 4 device categories on an advanced IoT testbed indicate underwhelming protection for commercially available IoT devices. They often are vulnerable to common security attacks; further, they do not include any security protection or user alerting system.

Based on our findings, we argue there is a need for IoT security and privacy systems deployed specifically for IoT devices and developed at the edge. To assist with such efforts, we make our datasets (IoT devices packet captures) and software public to encourage the creation of such systems and better security compliance from IoT vendors at `https://github.com/SafeNetIoT/spices`. We will maintain the codebase regularly to keep it up-to-date.

Acknowledgements

9 References

[1] R Shirley. A systematic content review of artificial intelligence and the internet of things applications in smart home. volume 10, page 3074, 2000.

[2] Hasanen Alyasiri, John Clark, Ali Malik, and Ruairí de Fréin. Grammatical evolution for detecting cyberattacks in internet of things environments. 07 2021.

[3] H Touqeer. Smart home security: Challenges, issues and solutions at different iot layers. volume 77, page 14053–89, 2021.

[4] R Shirley. Internet security glossary. May 2000.

[5] Eryk Schiller, Andy Aidoo, Jara Fuhrer, Jonathan Stahl, Michael Ziörjen, and Burkhard Stiller. Landscape of iot security. *Computer Science Review*, 44:100467, 2022.

[6] Anna Maria Mandalari, Daniel J Dubois, Roman Kolcun, Muhammad Talha Paracha, Hamed Haddadi, and David Choffnes. Blocking without breaking: Identification and mitigation of non-essential iot traffic. *Proceedings on Privacy Enhancing Technologies*, 4:369–388, 2021.

[7] Constantinos Kolias, Georgios Kambourakis, Angelos Stavrou, and Jeffrey Voas. Ddos in the iot: Mirai and other botnets. *Computer*, 50(7):80–84, 2017.

[8] Badis Hammi, Sherali Zeadally, Rida Khatoun, and Jamel Nebhen. Survey on smart homes: vulnerabilities, risks, and countermeasures. *Computers & Security*, 117:102677, 2022.

[9] Mikail Mohammed Salim, Shailendra Rathore, and Jong Hyuk Park. Distributed denial of service attacks and its defenses in iot: a survey. volume 76, pages 5320–5363, Jul 2020.

[10] Jerry John Kponyo, Justice Owusu Agyemang, Griffith Selorm Klogo, and Joshua Ofori Boateng. Lightweight and host-based denial of service (dos) detection and defense mechanism for resource-constrained iot devices. *Internet of Things*, 12:100319, 2020.

[11] Jingjing Ren, Daniel J Dubois, David Choffnes, Anna Maria Mandalari, Roman Kolcun, and Hamed Haddadi. Information exposure from consumer iot devices: A multidimensional, network-informed measurement approach. In *Proceedings of the Internet Measurement Conference*, pages 267–279, 2019.

[12] Quoc-Dung Ngo, Huy-Trung Nguyen, Van-Hoang Le, and Doan-Hieu Nguyen. A survey of iot malware and detection methods based on static features. *ICT Express*, 6(4):280–286, 2020.

[13] Rohan Doshi, Noah Apthorpe, and Nick Feamster. Machine learning ddos detection for consumer internet of things devices. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 29–35. IEEE, 2018.

[14] Manos Antonakakis, Tim April, Michael Bailey, Matt Bernhard, Elie Bursztein, Jaime Cochran, Zakir Durumeric, J Alex Halderman, Luca Invernizzi, Michalis Kallitsis, et al. Understanding the mirai botnet. In *26th {USENIX} security symposium ({USENIX} Security 17)*, pages 1093–1110, 2017.

[15] Anna Maria Mandalari, Hamed Haddadi, Daniel J Dubois, and David Choffnes. Protected or porous: A comparative analysis of threat detection capability of iot safeguards. In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 3061–3078. IEEE Computer Society, 2023.

[16] Gordon Fyodor Lyon. *Nmap network scanning: The official Nmap project guide to network discovery and security scanning.* Insecure. Com LLC (US), 2008.

[17] Tcpreplay - pcap editing and replaying utilities. `https://tcpreplay.appneta.com/`. Accessed: 2022-11-29.

[18] Mihalis Tsoukalos. Using tshark to watch and inspect network traffic. *Linux Journal*, 2015(254):1, 2015.

[19] Jon Postel. Internet control message protocol. Technical report, 1981.

[20] Laura South and Michelle Borkin. Ethical considerations of photosensitive epilepsy in mixed reality. *OSF Preprints*, 10 2020.

[21] Isabel Straw and Leonie Tanczer. Safeguarding patients from technology-facilitated abuse in clinical settings: A narrative review. *PLOS Digital Health*, 4(2):1, 2023.

[22] Leonie Tanczer, Isabel Lopez-Neira, and Simon Parkin. I feel like we're really behind the game': Perspectives of the united kingdom's intimate partner violence support sector on the rise of technology-facilitated abuse. *Journal of Gender Based Violence*, 5(3):431–450, 2021.

[23] Sandra Wachter. The gdpr and the internet of things: a three-step transparency model. *Law, Innovation and Technology*, 10(2):266–294, 2018.

[24] European Union Agency for Cybersecurity (ENISA). Baseline security recommendations for iot, 2021. Accessed: June 14, 2023.

[25] NIST. Cybersecurity for iot program, 2021. Accessed: June 14, 2023.

[26] Ali Tekeoglu and Ali Saman Tosun. A testbed for security and privacy analysis of iot devices. In *2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pages 343–348, 2016.

[27] Arunan Sivanathan, Franco Loi, Hassan Habibi Gharakheili, and Vijay Sivaraman. Experimental evaluation of cybersecurity threats to the smart-home. In *2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pages 1–6. IEEE, 2017.

[28] Earlence Fernandes, Jaeyeon Jung, and Atul Prakash. Security analysis of emerging smart home applications. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 636–654, 2016.

[29] Ali Tekeoglu and Ali Şaman Tosun. A testbed for security and privacy analysis of iot devices. In *2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pages 343–348, 2016.

[30] Hui Liu, Changyu Li, Xuancheng Jin, Juanru Li, Yuanyuan Zhang, and Dawu Gu. Smart solution, poor protection: An empirical study of security and privacy issues in developing and deploying smart home devices. In *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy*, IoTS & amp;P '17, page 13–18, New York, NY, USA, 2017. Association for Computing Machinery.

[31] Leonardo Babun, Kyle Denney, Z Berkay Celik, Patrick McDaniel, and A Selcuk Uluagac. A survey on iot platforms: Communication, security, and privacy perspectives. *Computer Networks*, 192:108040, 2021.