IoT Malware Ecosystem in the Wild: A Glimpse into Analysis and Exposures

Jinchun Choi*
jc.choi@knights.ucf.edu
University of Central Florida

Afsah Anwar*
afsahanwar@knights.ucf.edu
University of Central Florida

Hisham Alasmary
hisham@knights.ucf.edu
University of Central Florida

Jeffrey Spaulding
spauldi6@canisius.edu
Canisius College

DaeHun Nyang
nyang@inha.ac.kr
Inha University

Aziz Mohaisen
mohaisen@ucf.edu
University of Central Florida

ABSTRACT

The lack of security measures among the Internet of Things (IoT) devices and their persistent online connection give adversaries a prime opportunity to target them or even abuse them as intermediary targets in larger attacks such as distributed denial-of-service (DDoS) campaigns. In this paper, we analyze IoT malware and focus on the endpoints reachable on the public Internet, and play an essential part in the IoT malware ecosystem. Namely, we analyze endpoints acting as dropzones and their targets to gain insights into the underlying dynamics in this ecosystem, such as the affinity between the dropzones and their target IP addresses, and the different patterns among endpoints. Towards this goal, we reverse-engineer 2,423 IoT malware samples and extract strings from them to obtain IP addresses. We further gather information about these endpoints from public Internet-wide scanners, such as Shodan and Censys. For the masked IP addresses, we examine the Classless Inter-Domain Routing (CIDR) networks accumulating to more than 100 million (≈78.2% of total active public IPv4 addresses) endpoints.

CCS CONCEPTS

• **Security and privacy** → *Malware and its mitigation.*

KEYWORDS

Internet of Things, Endpoints, Malware

ACM Reference Format:

Jinchun Choi, Afsah Anwar, Hisham Alasmary, Jeffrey Spaulding, DaeHun Nyang, and Aziz Mohaisen. 2019. IoT Malware Ecosystem in the Wild: A Glimpse into Analysis and Exposures. In *The Fourth ACM/IEEE Symposium on Edge Computing (SEC 2019), November 7–9, 2019, Arlington, VA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3318216.3363379

*Both authors contributed equally to this research.

SEC 2019, November 7–9, 2019, Arlington, VA, USA

ACM ISBN 978-1-4503-6733-2/19/11…$15.00
https://doi.org/10.1145/3318216.3363379

1 INTRODUCTION

With the number of IoT devices soaring into the tens of billions [21], the potential adversaries have set their sights on these devices acknowledging their persistent connectivity. To this end, malicious code that targets IoT devices is on the rise that not only infects the device itself but also receives code updates from dropzones around the world. Acting as intermediate nodes, these infected devices have the potential to launch attacks on other targets to form an enormous distributed denial-of-service (DDoS) attack [20, 22, 23].

Moreover, the majority of these IoT devices are at a high risk to the new threats due to the lack of security awareness among consumers and the lack of consensus on security standards among the IoT industry [24]. Bastys *et al.* [9] demonstrate that popular IoT app platforms are susceptible to attacks by malicious applet makers. With less than half of consumers changing the default password on their IoT devices [13], it is no surprise that malware like the Mirai worm has been able to amass a large botnet to launch massive DDoS attacks by simply using a dictionary of common IoT login credentials [8]. Compared to traditional hardware with operating systems with automated updates, IoT devices tend to have slower patch times and insecure communication [10]. It makes them "ideal targets" for additional attacks like the **K**ey **R**einstallation **A**tta**ck** (KRACK) exploit [25]. It abuses design flaws in cryptographic Wi-Fi handshakes to reinstall existing keys which allows attackers to eavesdrop on network traffic or even inject malicious content [35]. With the proliferation of IoT devices in today's world, we even see decades-old attacks resurface to take advantage of vulnerable IoT devices. For example, the SSHowDowN Proxy attack discovered by Akamai [12] utilizes a 12-year old vulnerability in OpenSSH to effectively take over the device to remotely generate attack traffic. In the rest of this paper, we present the analysis and results of examining malicious IoT code focusing on the endpoints for dropzones and targets. Given the low computational power of the IoT devices, a single device would not be sufficient to carry out an attack [8]. However, exploiting IoT's scale, adversaries form a network of bots or intermediary targets, large enough to launch an attack of substantial magnitude. For example, recently, 13,000 compromised IoT devices were used to generate persistent traffic of 30 Gbps, targeting numerous financial institutions, with significantly low intensity than the recorded Mirai botnet attack that generated devastating attack traffic of 620 Gbps [32]. Similarly, a service provider in the US survived the largest DDoS attack with attack traffic staggering

to 1.7 Tbps [34]. Considering the landscape and the risks IoT devices possess, the malware authors can exploit the vulnerability to either attack them or hire them as intermediary targets for a future large scale attack. Reckoning that the malware sources, command and control (C2) servers, the intermediary targets, and the victim must be connected to the Internet in the attack scenario, which makes it important to study these endpoints. In this work, we extract endpoints from IoT malware samples by reverse-engineering those samples and perform a data-driven study to analyze their different traces, such as geographical affinities, open ports, and their susceptibility to attacks. We also try to understand the pattern shared among victims by different malware. Additionally, our work is important to understand the Indicators of Compromise (IoCs) and the behavioral aspects of the targets can be used for threat intelligence or threat hunting. For the masked IP addresses in the malware, we analyze the Classless Inter-Domain Routing (CIDR) addresses that accumulate to more than 100.7 million IP addresses accumulation to ≈78.2% of total IPv4 addresses. We calculate the ratio with respect to the total responsive public IPv4 addresses as observed using Censys [14].

Contributions. In this paper, the overarching goal is to analyze the dynamics exposed by the affinities between IoT malware endpoints. To this end, we make the following contributions:

(1) We analyze the dropzone-target inter-relationships. Specifically, we investigate the target IP addresses among different dropzone IP addresses.
(2) We perform a geographical analysis of the dropzones and targets. Towards this, we analyze the locations of dropzones and their target IP addresses.
(3) We analyze the attack exposure of networks and IP addresses. For masked target endpoints, we examine the entire network and study the network devices and their exposure to risk.

Organization. We describe our dataset, data augmentation and pipeline, and outline our goals and objectives in section 2. In section 3, we perform IP address-centric analysis of the endpoints followed by a network-centric analysis in section 4. We review the related work in section 5. Finally, we present conclusions and future work in section 6.

2 DATASET AND GOALS

We describe our dataset and its augmentation towards our goal. We then describe the aims and the objectives of this work. Specifically, we use an IoT malware dataset, perform static analysis on them, and finally use the strings to extract endpoints from them.

2.1 Dataset

We obtain our dataset from IoTPOT [27], a honeypot that emulates the Telnet services (later improved to include other services). We obtained a total of 2,423 IoT malware samples, which were graciously given to us by the authors of IoTPOT. The dataset represents four different malware families, labelled by augmenting the results from VirusTotal (VT) and by using AVClass [29]. For malware samples that do not have a decisive family label from the VT results, those malware samples are labeled as SINGLETON. The distribution of malware families can be seen in Table. 1.

Table 1: Distribution of malware by family. DZ - Dropzone.

Target family	Count	Pct.	DZ family	Count	Pct.
Gafgyt	930	95.58	Gafgyt	2,294	98.96
Tsunami	39	4.01	Tsunami	24	1.04
SINGLETON	3	0.31	-	-	-
Lightaidra	1	0.10	-	-	-

We reverse-engineer and analyze the malware samples using Radare2 [2], an open-source malware analysis framework. We find strings in the malware binary, especially IP addresses, and classify those addresses by their association with special keywords into two classes: `dropzone` and `target` IP addresses, defined as follows:

- **Dropzone IP.** Adversaries often keep malware binaries in remote servers to distribute them after gaining access to victim devices. These remote servers are identified by `dropzone` IP addresses, controlled and managed by the adversary and used for malware propagation and management. As such, the dropzone IP addresses are associated with wget, HTTP, TFTP, GET or FTP in the residual strings obtained from the malware analysis.
- **Target IP.** To infect victim hosts, malware uses a list of IP addresses, including target devices. We refer to these IP addresses as `targets`. We note that a large number of those target addresses in our analysis are masked. For example, 123.17.*.* is one of the target IP address that is masked at /16; the attacker can utilize this address targeting all IPs in the network address space.

We find the internal network addresses (*e.g.,* 192.168.*.*), loopback address (*e.g.,* 127.0.0.1) from our target dataset and remove them, since they are irrelevant to our analysis. Also, we note that the Mirai source code contained a list of "don't scan" addresses, including various U.S. DoD address blocks, as well as internal addresses [18], which we exclude.

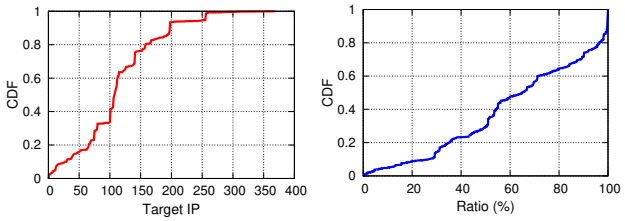
Data Augmentation. We group the target and dropzone addresses by malware. Since a dropzone can be used by multiple malware, and to help analyze the overall sample-space a dropzone caters to, we cluster the target IPs by each dropzone.

Using *UltraTools* [4], a free DNS and domain lookup tool, and *Censys* [1], a search engine for Internet-connected devices, each of the targets and dropzones is augmented with the following information: country, ASN (Autonomous System Number), and location (*e.g.,* latitude and longitude), open ports, etc.

We observe some dropzone addresses have no current information, *e.g.,* they are no longer connected to the Internet. This confirms that the dropzones are short-lived—long enough to carry out an attack and short not to be detected. As such, we leverage historical data of those IP addresses from Shodan [3] to determine the necessary data points associated with them.

Table 2: Top 5 dropzone IPs per the number of targets. Countries include: France (FR), United Kingdom (GB), Canada (CA) and United States (US).

Rk.	Dropzone IP	Country	#Malware	#Targets
1	163.172.104.150	FR	35	9,529
2	145.239.72.250	FR	22	5,632
3	45.76.131.35	GB	17	4,352
4	64.137.253.50	CA	26	3,066
5	198.175.126.89	US	11	2,816



(a) CDF graph of the number of overlapped target IPs. (b) Ratio of overlapped target IPs to total target IPs in a dropzones.

Figure 1: CDF graphs showing the distribution of the number of overlapped target IP addresses and their ratio.

2.2 Objectives

We used the dataset described earlier to address the following questions, which make up the specific objectives of this study:

- **Dropzone-Target Inter-Relationships.** Since malware associated with certain dropzones point to specific target IP addresses, could these IP addresses be similar or identical to the addresses of targets in other dropzones? To answer this, we reverse-engineered and analyzed the malware's disassembly to extract all target IP addresses for each dropzone.
- **Geographical Analysis.** What are the characteristics of the areas where the dropzones are located? How does this affect the distribution of dropzones and targets? For that, we analyze the distributions of the distance between the dropzones and their targets, and look at these distributions from various perspectives at the country and state level.
- **Attack Exposure.** How exposed are the IP addresses in the target's network address space? Towards this, we analyze the targets and look for vulnerabilities in the services that they use. For the masked targets, we analyze the network space and examine their up-to-date susceptibility.

To answer these questions, we divide our data-driven analysis into (i) IP centric analysis and (ii) network centric analysis. We cover those directions in the two following sections.

3 IP CENTRIC ANALYSIS

3.1 Dropzone-Target Inter-relationship

We inspect the dropzone-target relationship, we examine the affinity between the dropzone and the target IP addresses. While ≈77% of the unique target IPs received less than 10 attacks, one unique target IP received 72 attacks.

We found one dropzone IP (`50.115.166.193`) that was only associated with 1 malware sample. This malware sample pointed to 1,265 network addresses, which was significantly larger than the average of 121 target IP addresses for a typical malware sample. Also, they are masked network addresses (most of them are /16 masked, as mentioned in Table. 4), which means that one target network address can be larger dynamically. Conversely, the dropzone IP (`5.189.171.210`) has 86 associated malware samples, but each of those point to a single target IP address.

Dropzones can be found distributed mainly in North America and Europe. Moreover, through our further analysis we found that the first IP address (`163.172.104.150`) (Table 2) is associated with 35 malware samples affecting 9,529 target IPs.

Shared Targets Between Dropzones. To inspect the shared targets between dropzone IP addresses, we group the dropzone IP addresses and capture the common (overlapping) targets among the dropzones. Since dropzones can be associated with multiple instances of malware, each malware can have its own list of target IP addresses. If we assume that a dropzone has a union of target IPs for each malware belonging to that particular dropzone, we can aggregate all of their target IPs into a larger set of target IPs. This dataset of 18,158 dropzone IP pairs is a combination of only 247 unique dropzone IP addresses, from the dataset of 877 unique dropzone IP addresses.

We found 71 cases that had more than 300 overlapped target IPs in Fig. 1(a). Fig. 1(b) shows that there were 2,199 cases (12.11%) which are 100% overlapped between dropzones. Overall, we found 6,451 cases (35.53%) in which the overlap was more than 80%.

Summary. It is evident from the results of the above analysis that a large number of targets are being shared between dropzones. If the target IP addresses between different dropzones are matched 100%, it is possible that the attacker obtained the same targets through similar vulnerability analysis (*i.e.,* Shodan) or shared the target list from other attackers through underground communities.

3.2 Geographical Analysis

In this section, we focus on the distribution of the distances between the dropzones and their target IPs. **Distance Between Dropzone and Target.** As mentioned previously, a dropzone IP can be associated with several malware instances where each malware can point to one or more target IPs. Knowing the locations of these IPs, we calculate the distance between the dropzone and its target if they are related to the same malware instance. Each distance shows the locality of the attack. The total number of calculated distance cases is 111,480.

Our result of the majority of the distance shows the 8K-10K km range had the most frequent number of cases totaling 34,479 (30.93% of all dropzone-target distance cases). In this range, countries with the most target IPs are Brazil, Vietnam, and China, in order; while the dropzones are in European countries, including Italy, France, and the Netherlands. According to Table 3, a large

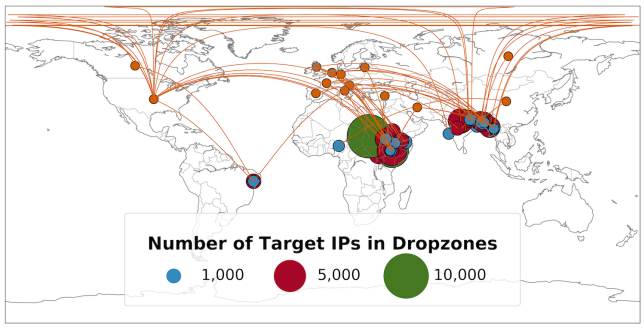


Figure 2: Attack trends between dropzones and target IPs. We only plot attacks that have over 500 target IPs. The orange circle represents dropzones, and blue, red, and green circles stand for target areas.

Table 3: Top 5 countries by the number of target and dropzone IPs. Countries include: United States (US), Netherlands (NL), France (FR), United Kingdom (GB), Italy (IT), Vietnam (VN), Brazil (BR), China (CN), India (IN) and Pakistan (PK).

Rk.	Country	Dropzones	Pct.	Rk.	Country	Targets	Pct.
1	US	1,041	43.25	1	VN	26,290	24.70
2	NL	278	11.55	2	BR	20,572	19.33
3	FR	188	7.81	3	CN	15,799	14.84
4	GB	183	7.60	4	IN	5,598	5.26
5	IT	177	7.35	5	PK	5,076	4.77

number of dropzones exist in the US, but they also have target IPs in Brazil, Vietnam, and China, with distance between dropzone and target in the range of 12K-14K km and 10K-12K km.

Country-level Analysis. In this part, we look at the overall attack trend between dropzones and their targets on a world-scale. For each dropzone, we collect all of the target IP addresses and extract location information (*e.g.,* latitude, longitude) to display the *average* position of the target area (not the exact position). The target areas are scaled according to the number of target IP addresses they contain. Fig. 2 shows the results of our country-level analysis, where we limit to only plotting dropzones with more than 500 target IP addresses. The locations of the dropzones (depicted in orange) are spread around various countries, but we highlight that there is a large concentration of target areas focused in Central Asia.

Table. 3 lists top 5 countries by the number of dropzone and target IPs. Note that the US has a large distribution of dropzones pointing to targets in Asian countries such as Vietnam. Additionally, China and Brazil contain a large number of target IP addresses originating from European countries. Imperva Incapsula (a global content delivery network and DDoS mitigation company) confirms that Vietnam (12.8%), Brazil (11.8%) and China (8.8%) were the countries with the most-infected devices (from the Mirai botnet) [18].

Summary. We observe that the US has a large distribution of dropzones targeting Asian countries such as Vietnam. We also see that China and Brazil are victims of attacks originating from European countries. Imperva Incapsula [18] back our findings, confirming that the Mirai botnet most targets Vietnam, Brazil, and China.

Table 4: Composition of Target IPs for masked and not-masked networks. "In Total" means the total number of target IPs, "In Unique" means the composition of non-duplicated target IPs.

Address	In Total	Pct.	In Unique	Pct.
/24	137	0.13	27	1.22
/16	104,369	98.07	1,869	84.53
/8	776	0.73	126	5.70
Not-masked	1,146	1.08	189	8.55
Total	106,428	100.00	2,211	100.00

4 NETWORK CENTRIC ANALYSIS

Malware specifically aimed at IoT devices tend to recruit a large number of intermediary targets to launch attacks on high-profile targets ultimately. To do this, the malware typically identify the intermediary targets using their IP addresses which are either mentioned in their source code or downloaded via dropzone. Additionally, these IP addresses could be masked IP addresses, showing only a prefix (*e.g.,* `123.17.%d.%d`).

In the previous sections, we analyzed the IP addresses explicitly stated in the malware code base. For the masked IP addresses, malware typically uses functions to hide the targets from the malware analysts and determine the targets dynamically. This functions invoked during run time to determine the remaining of the masked octets. Malware authors seldom obfuscate these functions – we, therefore, in this section, examine the entire /16, /24, or /8 network to probe their susceptibility.

Using CIDR notation, Table. 4 shows that 98.92% of the target endpoints are masked, mapping to 126 unique /8 networks and 1,869 unique /16 networks and 27 unique /24 networks. Removing the /16 networks covered in /8 and /24 networks, we have 125 /8 networks and 435 /16 networks. These 560 networks are then searched on Censys [14] which map to 100,793,403 active IP addresses, which also allows us to analyze their active ports. As different devices use different services to operate, we clustered the IP addresses by their device types and studied which ports were being used by the devices. Considering that open ports lead to increased security risks, we look for ports that are necessary for a device to operate without any misfire. Taking a conservative approach, we suggest that if a port is being used by less than 10% of devices in a given device type, it should be closed to reduce its exposure to risk. We observe that except for *VoIP phone* (over 77% of them used 5 ports), more than 75% of the devices among all the other device types have only two or less port being used. Fig. 3 shows the number of devices within a device type in log scale and the number of ports being used by less than 10% of the devices. In this figure, the two graphs show a similar pattern. We speculate this result is due to more attack taking place on the popular devices (*e.g.,* target devices of the Mirai consist of security cameras, DVRs, and consumer routers [8]).

Summary. The division of the endpoints by devices and then determining their exposure to the attackers represent the chances of an

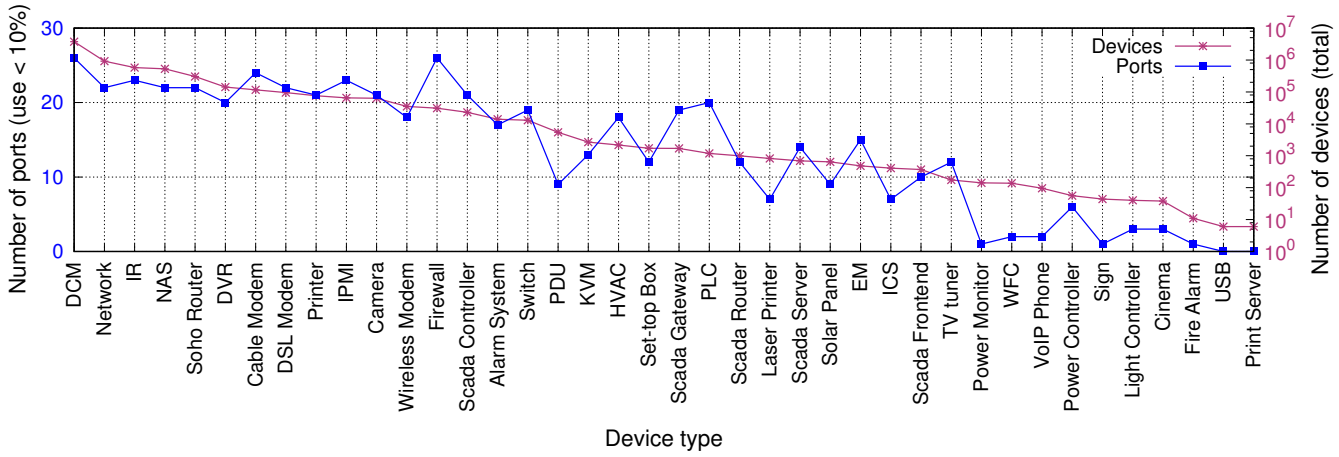


Figure 3: Total number of devices and the number of ports that used less than 10% of devices. The left Y-axis belongs to the number of ports (blue), the right Y-axis belongs to the total number of devices (red), and the X-axis is device types. Device types include: DSL/cable Modem (DCM), Infrastructure Router (IR), Network Attached Storage (NAS), Digital Video Recorder (DVR), Intelligent Platform Management Interface, (IPMI) Power Distribution Unit (PDU), Kernel-based Virtual Machine (KVM), Heating, ventilation, and Air Conditioning (HVAC), Programmable Logic Controller (PLC), Environment Monitor (EM), Industrial Control System (ICS), and Water Flow Controller (WFC).

endpoint being compromised. Based on our analysis we suggest the users close the ports that aren't necessary for the uninterrupted execution of their devices. These endpoints need to be further examined in-depth to understand the pattern that could predict an endpoints chances of being compromised. The suggestions could be finally narrowed, with specific device centred recommendations, and by probing them individually by performing an offensive penetration testing. However, in this work, we understand the data-centric landscape and put forward the suggestions with a conservative approach, and without carrying out any offensive analysis undermining ethics.

5 RELATED WORK

Recent studies related to IoT malware in the past few years have primarily focused on classifying IoT Malware. Su *et al.* [33] proposed a light-weight classification system based on image recognition that was tested on real IoT malware samples collected by IoTPOT [27], one of the first honeypots specifically for IoT threats. Using a dataset of 500 malware samples comprised of multiple families (including the Mirai botnet and Linux.Gafgyt) and benign samples from Ubuntu 16.04.3 system files, they converted each sample into 64x64 gray-scale images that were fed into a convolutional neural network achieving an average accuracy of 94%. Abusnaina *et al.* [5] presented Graph Embedding and Augmentation (GEA), a method to generate adversarial IoT software. With their approach, they successfully achieved a high misclassification rate in Control Flow Graph (CFG)-based features and deep learning network detection method, while ensuring that the generated software is executable. Studies have also utilized a combination of Shodan, an Internet-wide search engine for IoT devices [3], and known vulnerability databases to realize the potential risks inherent to Internet-connected devices. For example, Genge and Enachescu[17] proposed ShoVAT (Shodan-based Vulnerability Assessment Tool) and collected IoT device information such as open ports, when they were scanned, banner data, and their operating system through the Shodan API. They then used this information to confirm their identities in the NVD and revealed 3,922 known vulnerabilities among 1,501 services in 12 different institutions. Formby *et al.* [16] security challenges in the existing Industrial Control Systems (ICS) and address them by leveraging fingerprinting methods. Feng *et al.* [15] proposed a rule to discover and annotate IoT devices.

Cozzi *et al.* [11] investigated the different patterns and trends among the Linux malware in depth. Sivanathan *et al.* [31] analyzed the traffic of smart IoT environments gathered over a period of 3 weeks to characterize different traffic attributes. They differentiate IoT traffic from other traffic as well as identifying IoT devices with an accuracy of 95%. Alasmary *et al.* [6, 7] studied methods of malware detection based on graph-based features from CFGs. They show that CFGs, even with smaller size than similar software, can be powerful in identifying IoT applications, including distinguishing between benign and malicious ones. Related work on analyzing malware of other systems and evaluating the accuracy of their detection using various modalities are explored in [22, 23, 30].

To the best of our knowledge, there is no recent work that analyzes the relationships between the endpoints of IoT malware dropzones and their target devices. With that said, the closest study to our work is by Holz *et al.* [19] who presented one of the first empirical studies of malware and dropzones. Specifically, they focused on keyloggers and harvested data from dropzones which contained stolen credentials. Since keyloggers typically contact dropzones upon execution (to obtain a configuration file), the authors managed to successfully obtain the locations of several dropzones from several Autonomous Systems and countries, shown to be Russia and the US, among others.

West and Mohaisen [36] used 28,000 expert-labeled endpoints extracted from ≈100K malware binaries for binary threat classification with an accuracy of 99.4%. The endpoints were extracted using dynamic execution of malware in a sandboxed environment. Ouellette *et al.* [26] used deep learning to detect malicious endpoints. They used the features from obfuscated malware samples to feed to the classifier that performing classification on the cloud. Rafique and Caballero [28] used the network signatures from executing malware binaries to cluster them into families. Although limited, prior works have looked into investigating Linux malware, and the malware endpoints have not received the attention. Antonakakis *et al.* [8] analyzed the relationship between domains that were extracted by reverse-engineering the Mirai malware. With this work, we push towards filling the gap.

6 CONCLUDING REMARKS

In this paper, we analyze the ≈78.2% of total responsive public IPv4 endpoints among dropzones and their targets as extracted from IoT malware and spread across the globe from diverse perspectives. First, we analyze the dropzone-target inter-relationship and their affinity. We observe that the list of targets is shared between attackers, or are compiled by abusing shared susceptibilities. We visualize the target areas representing dropzone locations and their size scaled by the number of associated targets.

Our distributed analysis shows the exposure of endpoints which we correlate to the risk they possess. These endpoints need to be carefully and individually analyzed to extract patterns for predicting the chances of them being compromised. This, along with de-obfuscating functions to understand dynamic IP generation by malware will be our future work.

ACKNOWLEDGMENTS

This research was supported by Korea National Research Foundation under grant 2016K1A1A2912757 and a collaborative seed research grant from Cyber Florida.

REFERENCES

[1] 2018. Censys Landing Page. https://censys.io.
[2] 2018. Radare2. https://rada.re/r/.
[3] 2018. Shodan Landing Page. https://www.shodan.io.
[4] 2018. UltraTools Free IP Tools. https://bit.ly/2v2cLk4.
[5] Ahmed Abusnaina, Aminollah Khormali, Hisham Alasmary, Jeman Park, Afsah Anwar, and Aziz Mohaisen. 2019. Adversarial Learning Attacks on Graph-based IoT Malware Detection Systems. In *39th IEEE International Conference on Distributed Computing Systems, ICDCS 2019, Dallas, TX, USA, July 7*, Vol. 10. 2019.
[6] Hisham Alasmary, Afsah Anwar, Jeman Park, Jinchun Choi, DaeHun Nyang, and Aziz Mohaisen. 2018. Graph-based comparison of IoT and android malware. In *International Conference on Computational Social Networks*. Springer, 259–272.
[7] Hisham Alasmary, Aminollah Khormali, Afsah Anwar, Jeman Park, Jinchun Choi, Ahmed Abusnaina, Amro Awad, Dae Hun Nyang, and Aziz Mohaisen. 2019. Analyzing and Detecting Emerging Internet of Things Malware: A Graph-based Approach. *IEEE Internet of Things Journal* (2019).
[8] Manos Antonakakis, Tim April, Michael Bailey, Matt Bernhard, Elie Bursztein, Jaime Cochran, Zakir Durumeric, J. Alex Halderman, Luca Invernizzi, Michalis Kallitsis, Deepak Kumar, Chaz Lever, Zane Ma, Joshua Mason, Damian Menscher, Chad Seaman, Nick Sullivan, Kurt Thomas, and Yi Zhou. 2017. Understanding the Mirai Botnet. In *Proceedings of the 26th USENIX Security Symposium*. 1093–1110.
[9] Iulia Bastys, Musard Balliu, and Andrei Sabelfeld. 2018. If This Then What?: Controlling Flows in IoT Apps. In *Proceedings of the 25th ACM Conference on Computer and Communications Security, CCS*. 1102–1119.
[10] Elisa Bertino and Nayeem Islam. 2017. Botnets and internet of things security. *Computer* 2 (2017), 76–79.
[11] Emanuele Cozzi, Mariano Graziano, Yanick Fratantonio, and Davide Balzarotti. 2018. Understanding Linux Malware. In *IEEE Symposium on Security and Privacy, S&P*. 161–175.
[12] Developer. 2016. Akamai Threat Research Team Identifies New Abuses Of OpenSSH Vulnerability. https://goo.gl/cVr51i.
[13] Developer. 2018. IoT Consumer Insights. https://bit.ly/2NV34e8.
[14] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J. Alex Halderman. 2015. A Search Engine Backed by Internet-Wide Scanning. In *Proceedings of the 22nd ACM Conference on Computer and Communications Security, CCS*. 542–553.
[15] Xuan Feng, Qiang Li, Haining Wang, and Limin Sun. 2018. Acquisitional rule-based engine for discovering Internet-of-Things devices. In *Proceedings of the 27th USENIX Security Symposium*. 327–341.
[16] David Formby, Preethi Srinivasan, Andrew Leonard, Jonathan Rogers, and Raheem A Beyah. 2016. Who's in Control of Your Control System? Device Fingerprinting for Cyber-Physical Systems.. In *Network and Distributed System Security Symposium, NDSS*.
[17] Béla Genge and Calin Enachescu. 2016. ShoVAT: Shodan-based vulnerability assessment tool for Internet-facing services. *Security and Communication Networks* 9, 15 (2016), 2696–2714. https://doi.org/10.1002/sec.1262
[18] Ben Herzberg, Dima Bekerman, and Igal Zeifman. 2016. Breaking Down Mirai: An IoT DDoS Botnet Analysis. https://bit.ly/2dQbvYo.
[19] Thorsten Holz, Markus Engelberth, and Felix Freiling. 2009. Learning More About the Underground Economy: A Case-study of Keyloggers and Dropzones. In *Proceedings of the 14th European Conference on Research in Computer Security (ESORICS'09)*. 1–18.
[20] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas. 2017. DDoS in the IoT: Mirai and Other Botnets. *Computer* 50, 7 (2017), 80–84.
[21] Peter Middleton. 2016. Forecast Analysis: Internet of Things–Endpoints, Worldwide, 2016 Update. http://gtnr.it/2oRo4aN.
[22] Aziz Mohaisen and Omar Alrawi. 2014. AMAL: High-Fidelity, Behavior-Based Automated Malware Analysis and Classification. In *Proc. of WISA*.
[23] Aziz Mohaisen and Omar Alrawi. 2014. AV-Meter: An Evaluation of Antivirus Scans and Labels. In *Proc. of DIMVA*.
[24] Janakiram MSV. 2016. Security Is Fast Becoming The Achilles Heel of Consumer Internet of Things. https://bit.ly/2Ovco9B.
[25] Alfred Ng. 2017. Why KRACK could hit your smart home's Wi-Fi the hardest. https://cnet.co/2ze2JvM.
[26] Jacob Ouellette, Avi Pfeffer, and Arun Lakhotia. 2013. Countering malware evolution using cloud-based learning. In *Proceedings of the 8th International Conference on Malicious and Unwanted Software: "The Americas", MALWARE*. 85–94.
[27] Yin Minn Pa Pa, Shogo Suzuki, Katsunari Yoshioka, Tsutomu Matsumoto, Takahiro Kasama, and Christian Rossow. 2016. IoTPOT: A Novel Honeypot for Revealing Current IoT Threats. *Journal of Information Processing* 24, 3 (2016), 522–533. https://doi.org/10.2197/ipsjjip.24.522
[28] M. Zubair Rafique and Juan Caballero. 2013. FIRMA: Malware Clustering and Network Signature Generation with Mixed Network Behaviors. In *Proceedings of the 16th International Symposium on Research in Attacks, Intrusions, and Defenses, RAID*. 144–163.
[29] Marcos Sebastián, Richard Rivera, Platon Kotzias, and Juan Caballero. 2016. AVclass: A Tool for Massive Malware Labeling. In *Research in Attacks, Intrusions, and Defenses - 19th International Symposium, RAID 2016, Paris, France, September 19-21, 2016, Proceedings*. 230–253. https://doi.org/10.1007/978-3-319-45719-2_11
[30] Feng Shen, Justin Del Vecchio, Aziz Mohaisen, Steven Y. Ko, and Lukasz Ziarek. 2017. Android Malware Detection Using Complex-Flows. In *37th IEEE International Conference on Distributed Computing Systems, ICDCS 2017, Atlanta, GA, USA, June 5-8, 2017*. 2430–2437.
[31] Arunan Sivanathan, Daniel Sherratt, Hassan Habibi Gharakheili, Adam Radford, Chamith Wijenayake, Arun Vishwanath, and Vijay Sivaraman. 2017. Characterizing and classifying IoT traffic in smart cities and campuses. In *Proceedings of the 2017 IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS)*. 559–564.
[32] Tom Spring. 2019. Mirai Variant Targets Financial Sector With IoT DDoS Attacks. https://tinyurl.com/yaecazap.
[33] Jiawei Su, Danilo Vasconcellos Vargas, Sanjiva Prasad, Daniele Sgandurra, Yaokai Feng, and Kouichi Sakurai. 2018. Lightweight Classification of IoT Malware based on Image Recognition. *arXiv preprint arXiv:1802.03714* (2018).
[34] Liam Tung. 2019. New world record DDoS attack hits 1.7Tbps days after landmark GitHub outage. https://tinyurl.com/yb7zka8y.
[35] Mathy Vanhoef and Frank Piessens. 2017. Key Reinstallation Attacks: Forcing Nonce Reuse in WPA2. In *Proceedings of the 24th ACM Conference on Computer and Communications Security, CCS*. 1313–1328.
[36] Andrew G. West and Aziz Mohaisen. 2014. Metadata-Driven Threat Classification of Network Endpoints Appearing in Malware. In *Proceedings of the 11th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, DIMVA*. 152–171.