A Scalable Platform for Enabling the Forensic Investigation of Exploited IoT Devices and Their Generated Unsolicited Activities

Article *in* Digital Investigation · December 2019

DOI: 10.1016/j.fsidi.2020.300922

CITATIONS
0

READS
209

4 authors, including:

Elias Bou-Harb
University of Texas at San Antonio
99 PUBLICATIONS **1,313** CITATIONS

SEE PROFILE

Chadi Assi
Concordia University Montreal
416 PUBLICATIONS **7,412** CITATIONS

SEE PROFILE

Mourad Debbabi
Concordia University Montreal
420 PUBLICATIONS **5,373** CITATIONS

SEE PROFILE

A Scalable Platform for Enabling the Forensic Investigation of Exploited IoT Devices and their Generated Unsolicited Activities

Sadegh Torabi[a,*], Elias Bou-Harb[b], Chadi Assi[a], Mourad Debbabi[a]

[a]*Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada*
[b]*The Cyber Center For Security and Analytics, University of Texas at San Antonio, San Antonio, United States*

Abstract

The analysis of large-scale cyber attacks, which utilized millions of exploited Internet of Things (IoT) devices to perform malicious activities, highlights the significant role of compromised IoT devices in enabling evasive and effective attacks at scale. Motivated by the shortage of empirical data related to the deployment of IoT devices, and the lack of understanding about compromised devices and their unsolicited activities, in this paper, we leverage a big data analytics framework (`Apache Spark`) to design and develop a scalable system for automated detection of compromised IoT devices and characterization of their unsolicited activities. The system utilizes IoT device information and passive network measurements obtained from a large network telescope, while implementing an array of data-driven methodologies rooted in data mining and machine learning techniques, to provide a macroscopic view of IoT-generated malicious activities. We evaluate the system with more than 4TB of passive network measurements and demonstrate its effectiveness in the network forensic investigation of compromised devices and their activities, in near real-time. In addition, we empirically analyze and elaborate on the capabilities of the developed system as a scalable infrastructure, which can support a number of applications that enable IoT-centric forensics.

Keywords: Network telescope (darknet), compromised IoT devices, big data analytics, scanning campaigns, IoT botnet, network forensics

1. Introduction

Internet of Things (IoT) devices are being used to facilitate efficient data collection, monitoring, and information sharing. Despite their benefits and wide spread adoption, the insecurity of the IoT paradigm turns such devices into attractive targets for adversaries. More importantly, recent large-scale attacks unveiled an important role of compromised IoT devices as effective attack enablers, which can be utilized to generate unsolicited activities within well-coordinated botnets (Antonakakis et al., 2017; Cimpanu, 2018; Safaei Pour et al., 2019b). For instance, the `Mirai` botnet utilized millions of compromised IoT devices to execute one of the largest targeted Denial-of-Service (DoS) attacks (Antonakakis et al., 2017). On the other hand, while the `Hajime` botnet was not used to perform such attacks yet, the in-depth analysis of the botnet reveals its sophisticated design and extended capabilities, which makes it more powerful than previously detected botnets in terms of infiltrating IoT devices at scale (Herwig et al., 2019).

In order to mitigate and prevent large-scale IoT-driven cyber attacks, there is a need to possess an Internet-scale perspective of the exploited IoT devices and their unsolicited activities over a period of time. This however, is challenging due to the shortage of empirical data on the deployment of IoT devices, and the lack of scalable cyber-threat intelligence reporting and analysis capabilities that can trigger informed decisions for in-depth forensic investigations in near real-time (Neshenko et al., 2019). Furthermore, given that IoT-tailored malware heavily rely on large-scale Internet reconnaissance activities to propagate by exploiting vulnerable IoT devices at scale (Antonakakis et al., 2017; Cimpanu, 2018), detecting and analyzing these scanning activities can provide useful insights on the compromised IoT devices and the characteristics of their underlying malicious operations and infrastructure (e.g., IoT botnets).

In this paper, we address these challenges by developing a system that facilitates effective, efficient, and cyber forensic research in the context of IoT devices by providing an infrastructure for enabling a number of operations for detecting exploited IoT devices and fingerprinting their unsolicited activities. The automated system leverages a multi-stage, data-driven methodology by utilizing passive network telescope data (darknet) along with IoT device information obtained from an online IoT device search engine (`Shodan` (SHODAN, 2019)). Furthermore, the system leverages `Apache Spark`, a big data analytics framework that supports distributed computing to achieve scalable and near-real time operations.

The system is evaluated using 4TB (120 hours) of IoT-generated unsolicited traffic captured "in the wild," to identify 27,849 exploited IoT devices that generated over 308M packets, among which, the majority were scanning packets (about 300M). Moreover, while the system supports various views for macroscopic and fine-grained monitoring and analysis of the

*Corresponding author. Concordia Institute for Information Systems Engineering, Concordia University, Montreal, Canada.
Email address: `sa_tora@encs.concordia.ca` (Sadegh Torabi)

detected activities, it utilizes behavioral characteristics of IoT devices in terms of aggregated flow features to support the implementation of a number of network forensic applications such as detecting and fingerprinting scanning campaigns, investigating campaign persistence and evolution, inferring IoT botnets, and identifying IoT DDoS victims.

Along this line of thoughts, we frame the contributions of this paper as follows:

- We implement a scalable system that enables network forensic investigations through inferring compromised IoT devices and characterizing their unsolicited activities. The system, which utilizes IoT device information and passive network traffic captured at a large network telescope, leverages the capabilities of a big data analytics framework (`Apache Spark`) to implement multi-level data-driven methodologies rooted in data mining and unsupervised machine learning.
- We discuss the network forensic capabilities of the implemented system to support several operations including but not limited to: monitoring and fingerprinting unsolicited IoT-generated activities, inferring compromised IoT devices and characterizing the generated scanning campaigns, identifying IoT devices that have fallen victims of DDoS attacks, inferring IoT botnets, and performing temporal network forensic analysis.
- We evaluate the effectiveness of the system by analyzing over 4TB IoT-generated traffic over 5 days and identifying more than 27,000 IoT devices that generated about 300 million unsolicited packets. More importantly, the results of our performance evaluation affirm the scalability of the system with respect to large amount of analyzed network traffic, while generating results in near real-time.

The remainder of the paper is organized as follows. Section 2 reviews the recent literature on IoT threats and vulnerabilities. Detailed information on the design and implementation of the system is presented in Section 3. In Section 4, we present an experimental setup for evaluating the proposed system using real-world IoT traffic, while presenting empirical and performance analysis results. Finally, we conclude the paper by summarizing the findings in Section 5.

2. Related Work

In this section, we discuss the literature on a number of related topics to the IoT paradigm, including IoT device categories, data collection, traffic monitoring, and analysis.

IoT Device and Protocol Vulnerabilities. IoT device vulnerabilities have been discussed in the literature from different angles. For instance, Cui and Stolfo (2010) provided quantitative evidence on the vulnerable devices that are configured with factory default root passwords. This vulnerability was in fact one of the main reasons behind the large-scale outbreak of the Mirai botnet in late 2016 (Antonakakis et al., 2017). Considering the impact of vulnerability analysis in identifying and addressing IoT malware/botnets, a number of studies focused on developing tools and test beds for extensive assessment of IoT devices and their firmware images (Sachidananda et al., 2017; Costin et al., 2014; Chen et al., 2016). Apart from executing device and firmware vulnerability analyses, a number of IoT security research work has been dedicated to securing IoT context-aware permission models and program-flow operations (Yu et al., 2015; Jia et al., 2017; Fernandes et al., 2016). IoT protocol vulnerabilities were also studied for numerous types of home automation IoT devices and unveiled various insights with regards to the security, privacy, and usability of the implemented access control models (Ur et al., 2013; Ronen and Shamir, 2016).

IoT Data Capturing Initiatives. A number of ongoing projects have been implemented to perform active scanning of the Internet in order to locate and profile Internet connected devices on frequent basis. For instance, `Censys` was created by security researchers at the University of Michigan as an online tool for discovering devices, networks, and infrastructure on the Internet while monitoring changes over time (Durumeric et al., 2015). `Shodan` on the other hand (SHODAN, 2019), performs IP banner analysis to provide a more specialized online IoT device search engine that indexes different types of IoT devices. In line with the same approach, Feng et al. (2018) proposed a rule-based IoT device detection model that addresses the limitations of conventional banner grabbing/analysis techniques (e.g., insufficient device information) by utilizing device information from multiple online resources.

IoT Honeypots. Given the rareness of IoT-relevant empirical data, passive network traffic analysis has been introduced as an effective approach towards studying Internet-wide cyber threats associated with IoT devices. For instance, IoTPOT, was deployed by Pa et al. (2016) as a honeypot that emulates Telnet services of various IoT devices running on different CPU architectures. In alternative work, Guarnizo et al. (2017) presented the Scalable High-Interaction Physical Honeypot platform for IoT devices (SIPHON). The authors demonstrated an approach for imitating various IoT devices on the Internet to attract significant malicious traffic by leveraging worldwide wormholes and a few physical devices. Luo et al. (2017) implemented a machine learning approach to create an intelligent honeypot that automatically learns the behavioral responses of IoT devices through active scanning in order to mimic realistic interactions with attackers. Vervier and Shen (2018) deployed a honeypot that captured a wider range of emerging IoT threats as compared to previous honeypots (e.g., IoTPOT). They used 6 months of collected data along with multiple sources of cyber-intelligence to explore current IoT malware and their emerging behavioral characteristics.

Passive Network Measurements. In addition to IoT-tailored honeypots, passive network telescope or darknet data, which represents one-way network traffic collected at unused IP addresses over the Internet, has been adopted to analyze cyber activities and obtain cyber-intelligence (Labovitz et al., 2001).

More importantly, with the rise of IoT-driven cyber attack
sive network telescope data was leveraged to capture ar
lyze unsolicited IoT scanning activities. For instance, F
et al. (2017) presented a probabilistic model for sanitizi
work telescope data and inferring orchestrated probing
paigns towards cyber-physical systems (CPS). Furthe
Antonakakis et al. (2017) used unique Mirai traffic sig
to capture Mirai-related scans at the network telescope
ther analysis of the botnet. Torabi et al. (2018) proposed
driven methodology to infer compromised IoT devices by
aging IoT device information and darknet data through
ecution of correlation algorithms on IP header informat
line with the same line of research, Safaei Pour et al. (
proposed a data dimensionality reduction technique to in
characterize Internet-scale IoT probing campaigns by ana
passive network measurements collected from the dark
addition, Safaei Pour et al. (2019b) utilized several shall
deep learning models to sanitize telescope data and infer probing activities generated by compromised IoT devices based on a number of flow features.

Despite the promising work done towards inferring and characterizing compromised IoT devices and their unsolicited activities, this paper complements previous contributions by extending network telescope research to address the problems of detecting compromised IoT devices and characterizing their underlying unsolicited activities. In addition, considering the lack of scalable tools/systems for monitoring and investigating unsolicited IoT-generated activities in the wild, in this paper, we implement a near real-time threat detection system by utilizing passive network measurements while providing an infrastructure for investigating and fingerprinting activities generated by compromised IoT devices in near real-time. More importantly, the system enables the development of several applications for executing IoT-centric research and generating threat intelligence with respect to unsolicited IoT behaviors.

3. Design and Implementation

In this section, we present the design and implementation of our proposed system, which consists of four main components, a shown in Figure 1.

3.1. IoT Data Collection Module

The aim of this module is to obtain IoT device information and process it for further use in the system. A common approach for detecting IoT devices is to perform active scanning of the Internet address space and subsequent banner analysis. Indeed, we leverage `Shodan` (SHODAN, 2019), which is one of the largest online IoT device search engines that utilizes a similar approach to infer information on different types of Internet-connected hosts, including IoT devices. `Shodan` provides an API for searching and accessing information related to connected devices. In this paper, we focus on information such as device IP address, type, operator, and location information, to name a few.

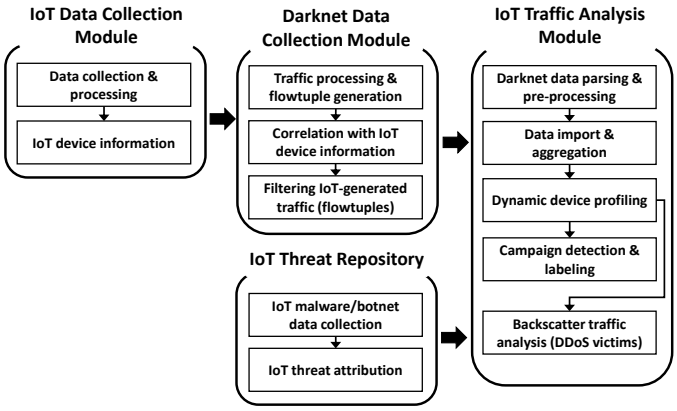


Figure 1: Overall architecture of the implemented system.

3.2. Darknet Data Collection Module

The system utilizes the UCSD real-time network telescope (darknet), which is one of largest available sources of passive traffic with about 16.7 million IPv4 addresses that receive over a billion packets per hour (CAIDA, 2019). Darknet traffic represents one-way packets captured at unused, yet routable IP addresses that belong to the darknet operators. Given that traffic received at the darknet is likely to be unsolicited, the module aims at correlating the obtained IoT device information (i.e., device IP address) with darknet traffic to identify suspicious IoT-generated activities. Furthermore, depending on the implementation of the darknet, these packets undergo several pre-processing and filtering operations to eliminate noise (e.g., unnecessary traffic/information) and classify traffic categories (e.g., Internet scanning and backscatter packets). The system processes the obtained IoT-generated traffic as flowtuples, which illustrate incoming packets from a source IP to a darknet IP address during one minute time intervals. A flowtuple consists of the following nine information fields: source/destination IP addresses, source/destination ports, used protocol, time to live (TTL), TCP flags, IP length, and total number of packets sent from a source IP to a destination IP address (per minute).

3.3. IoT Traffic Analysis Module

The IoT traffic analysis module, which utilizes `Apache Spark`, consists of the following main components:

3.3.1. Darknet Data Parsing and Pre-Processing

The IoT-generated flowtuples obtained from the darknet are pre-processed using the darknet traffic parser to identify different types of traffic according to the protocol and used flags. We identify backscatter traffic (Blenn et al., 2017), which represent reply packets (e.g., SYNACK) generated by IoT devices as a result of denial of service (DoS) attacks using spoofed IP addresses that belong to the darknet address space. Indeed, the analysis of backscatter packets can reveal information on benign IoT devices that were victims of DoS attacks. Moreover, we identify scanning traffic, which represents a significant portion of the darknet traffic. Given that benign IoT de-

vices have no justifiable reason to continuously send scanning packets towards the darknet, we label these devices as compromised or exploited. The scanning traffic contains mainly TCP-SYN scanning packets, followed by a relatively smaller number of ICMP Echo requests (Ping). We also identify UDP traffic, which is less commonly used for scanning the Internet due to the stateless nature of the packets (Moore et al., 2003; Durumeric et al., 2014). Finally, the parsed/processed data (flowtuples) will be fed into the aggregation module for further analysis.

3.3.2. Data Import and Aggregation

The system utilizes *Apache Spark*'s *DataFrame* API to import processed darknet flowtuples into distributed collections of data organized into named columns (DataFrame) (Spark, 2019). Given the imported flowtuples, the data aggregation module is implemented by utilizing a set of methods to group IoT generated traffic per source IP address, while aggregating IoT-generated traffic over specified discrete time interval(s) to obtain different views of the compromised IoT devices and their behaviors over various analysis periods. For instance, a macroscopic view of the data is presented through summarizing IoT-generated traffic over the analysis intervals (i.e., generated packets, number of compromised IoT devices, etc.). In addition, IoT traffic is combined to identify aggregated flow features per IoT device with different levels of interval granularity (e.g., per minute or per hour), which is utilized to infer temporal characteristics of IoT devices. This feature can be handy when analyzing scanning campaigns and their evolution over time, as described in Section 4.2.3.

3.3.3. Dynamic Device Profiling

The systems utilizes the data aggregation outcomes to create a dynamic profile for every active IoT device over accumulative analysis intervals. These profiles contain a list of IoT device information including but not limited to: source IP, targeted destination ports and IP addresses, aggregated flow features, traffic statistics and summaries, and device info (e.g., type location, ISP). The device profiles are dynamically updated after processing every input file over the accumulative time intervals. However, in order to maintain scalability and avoid accumulating unnecessary data, the system maintains a last seen flag to clean out IoT devices after a number of inactive intervals. These device profiles, which consist of device-specific measurements and information, are stored in JSON files in order to be used for further analysis when necessary.

Traffic aggregation and device profiling can result in several outcomes. In terms of backscatter traffic, the aggregated traffic reveals the intensity and duration of inferred DoS attacks towards the IoT devices. On the other hand, given that adversaries leverage controlled botnets to perform Internet-scale probing activities to identify hosts that run certain vulnerable services, the outcome of the module is used to profile IoT devices based on their scanning objectives (targeted ports) and overall scanning behaviors (aggregate flow features) over a period of time.

3.3.4. Campaign Detection and Labeling

Given that compromised IoT devices are utilized to scan certain vulnerable services/ports, the system groups the identified devices into correlated scanning campaigns according to their scanning objectives. Furthermore, given that orchestrated scanning campaigns performed by botnets tend to generate similar behavioral characteristics over a period of time, the system implements subsequent clustering using unsupervised learning techniques to identify IoT botnets. These botnets are then labeled for use in further investigations. It is also important to note that data aggregation and campaign detection/labeling processes are performed continuously over specified time intervals, and therefore, the inferred campaigns and botnet labels are updated periodically to account for any changes in the involved IoT devices and their behavioral characteristics. This is an important feature of our implemented system as it enables detecting temporal changes in the behaviors of the compromised IoT devices acting within a coordinated botnet.

3.4. IoT Threat Repository

The system maintains a local threat repository, which is built by compiling various publicly available information about recently discovered IoT malware/botnets such as malicious devices' IP addresses, targeted vulnerabilities, exploited services/ports, targeted device types, and botnet/malware family, to name a few. These information will be utilized by the system to create partial labels for the identified IoT devices and their malicious activities (e.g., scanning campaigns). In addition, the system will automatically update the created threat repository with information related to new, previously undetected malicious IoT behaviors and/or exploitations using feedback loops to adapt with the evolving nature of IoT threats.

4. Experimental Results and Evaluation

The system is built by deploying `Apache Spark` using *PySpark* in a standalone mode on a single node, with Debian Operation System (Ubuntu 18.04 version), 8 CPU cores (`Intel® Xeon(R) CPU E3-1240 v5 @ 3.50GHz`), 64GB memory, and 5TB storage space. In what follows, we describe details of the data collection, analysis results, and performance evaluation results.

4.1. Data Collection and Sampling

In this study, we obtain information about more than 400,000 IoT devices from `Shodan` (SHODAN, 2019). The collected data belongs to different types of IoT devices deployed in the consumer realm such as routers, IP cameras, printers, and DVRs, to name a few. Furthermore, we processed more than 4TB of passive darknet data, which represents traffic generated by millions of IoT and non-IoT hosts towards the darknet. We correlate the collected IoT device IP addresses from `Shodan` with the processed darknet traffic to obtain traffic generated by 27,940 unsolicited devices towards the darknet. Note that while the implemented system is generic and can be fed with hourly traffic from the darknet at any time frame, for the

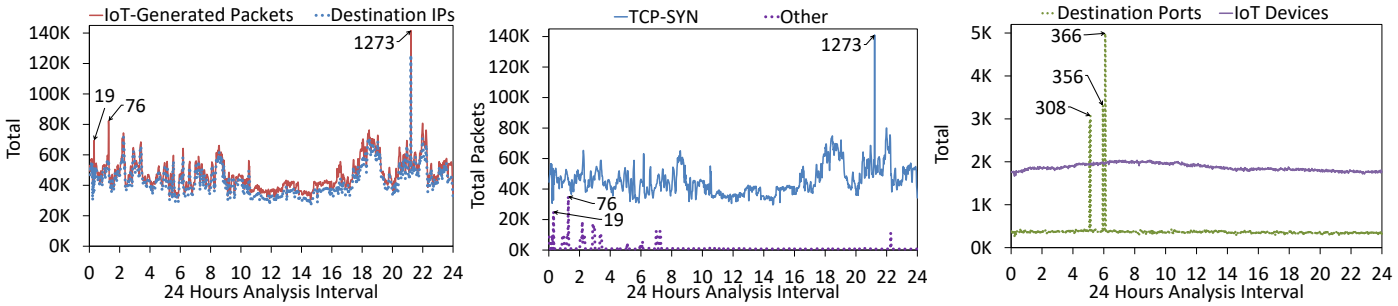


Figure 2: Macroscopic views of the various IoT-generated packets towards the darknet over 24 hours of analysis interval (1,440 minutes).

sake of experimentation, we analyzed a large sample of darknet traffic representing 120 hours (5 days) of traffic that was captured in November, 2018. We obtained about 324.6M IoT-generated packets (308M flows), with a mean of about 2.7M packets per hour. These packets represent mainly TCP-SYN traffic (87.7%), followed by UDP (10.9%), ICMP Echo requests (0.5%), and backscatter (0.3%) traffic. Other packets such as misconfiguration, account for about 0.6% of the IoT-generated traffic.

4.2. Results (Applications)

In what follows, we present experimental results with respect to leveraging the developed system to analyze data and enable a number of network forensic applications and investigations.

4.2.1. Monitoring Unsolicited Activities: A Macroscopic View

The system outputs multiple high-level macroscopic views of IoT-generated traffic over the analysis intervals. For instance, Figures 2(a–c) provide an Internet-scale perspective of the IoT devices and their online behaviors over a 24-hour analysis interval. These views are useful for enabling early threat detection through monitoring the overall IoT activities on the Internet, while highlighting trends and temporal changes in the overall activities of IoT devices in near real-time. For instance, we found a strong correlation between the number of IoT-generated packets and the targeted destination IP addresses in the darknet (Figure 2a), which reflects typical Internet reconnaissance activities. In fact, over 97% of the IoT-generated traffic at the majority of the observed time intervals were TCP-SYN packets (Figure 2b), which are commonly used for scanning the Internet.

Moreover, by looking at the abrupt increases in the total number of IoT-generated packets (e.g., minutes 19, 76, and 1273 in Figure 2a) and comparing them to the detailed distribution of the packets as illustrated in Figure 2b, we note that TCP-SYN packets contributed towards the majority of packets at minute 1273, while other packets such as UDP, ICMP-REQ, and backscatter, contributed towards the majority of packets at minutes 19 and 76. The system can also be used to find the number of active IoT devices that generate packets towards the darknet, which could be useful for estimating the magnitude of IoT exploitations over time. Finally, by looking at the sudden increase in the number of targeted destination ports (e.g., minutes 308, 356, and 366), we detect traces of intensive port scanning activities related to the behaviors of compromised IoT devices (Figure 2c).

Table 1: Compromised IoT devices and their generated scanning traffic type(s).

Scanning Traffic	Devices		Packets	
	Count	(%)	Count (M)	(%)
UDP	**14,314**	**51.40**	33.21	10.32
TCP-SYN	3,770	13.54	**167.88**	**52.19**
ICMP-REQ	23	0.08	0.71	0.22
TCP-SYN/UDP	9,728	34.93	118.38	36.80
UDP/ICMP-REQ	40	0.14	1.83	0.57
TCP-SYN/ICMP-REQ	36	0.13	0.97	0.30
All types	31	0.11	1.05	0.32

4.2.2. Detecting Compromised IoT Devices

We leveraged the proposed system presented in Section 3 to identify 27,849 compromised IoT devices that were sending scanning packets (TCP-SYN, UDP, and ICMP-REQ) towards the darknet during the 5 days analysis interval. As summarized in Table 1, slightly over half of these devices (51.4%) were sending only UDP packets (32.21M packets). Furthermore, while a relatively smaller number of devices (13.54%) were generating only TCP-SYN packets, they account for significantly more scanning traffic, with about 167.8M TCP-SYN packets (52.19% of total scans). In addition, about 35% of all devices generated both UDP and TCP-SYN scanning packets, with a total of about 118.4M scanning packets, representing about 36.8% all scanning packets. On the other hand, only 68 IoT devices were generating ICMP-REQ packets (about 0.2% of the scanning traffic).

Given the identified IoT devices, we utilize our system along with device information collected from Shodan to shed light on a number of properties associated with the exploited IoT devices such as device type, model, and location (hosting countries). These properties can be used to infer large-scale exploitations affecting vulnerable devices over the Internet. Indeed, the distribution of the compromised IoT devices per device type (Figure 3) shows about 33.7% of the devices to be

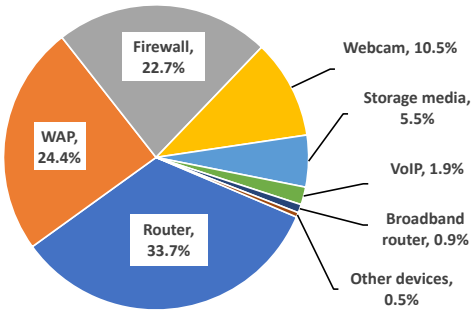


Figure 3: Compromised IoT device types.

Table 2: Compromised IoT device models (scanning).

Device Model	Count	%
MikroTik router	**8,035**	**28.9**
SonicWALL firewall	4,654	16.7
Linksys wireless-G WAP	1,944	7.0
DD-WRT supported routers	1,380	5.0
TP-LINK WR740N WAP	1,238	4.4
Cisco router	923	3.3
Talk Talk YouView box	751	2.7
TP-LINK WR841N WAP	681	2.4
Avtech AVN801 network camera	671	2.4
ZyXEL ZyWALL	618	2.2

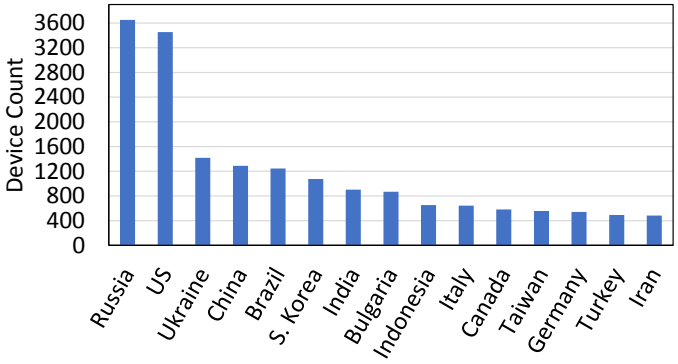


Figure 4: Countries with the largest number of compromised devices.

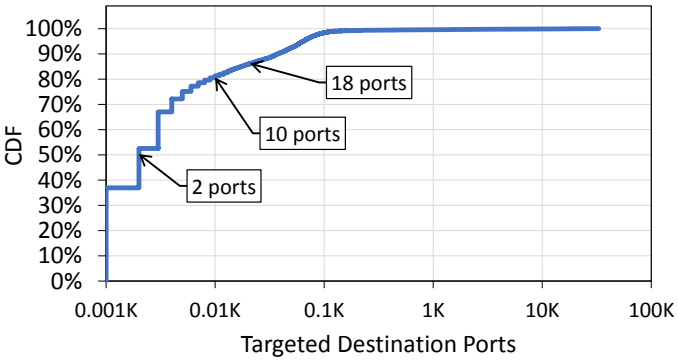


Figure 5: The commutative distribution of the total number of scanned destination ports by the exploited IoT devices.

routers, followed by WAP (24.4%), Firewalls (22.7%), and Webcams (10.5%), respectively. In addition, as summarized in Table 2, about 29% of the exploited devices were MikroTik routers, followed by a relatively smaller number of SonicWALL firewalls (16.7%), and Linksys WAPs (7%). Moreover, these devices were hosted across 192 countries (Figure 4), with the largest number of devices to be found in Russia (3,650), the U.S. (3,454), Ukraine (1,417), and China (1,288), respectively. The distribution of compromised IoT devices per device type, model, and country can reveal information about the overall threat landscape that targets vulnerable IoT devices.

4.2.3. Inferring and Monitoring Scanning Campaigns

Our analysis showed that the majority of IoT-generated traffic towards the darknet consists of scanning packets (99.1%), among which about 88.5% were TCP-SYN scans, followed by UDP (11%), and ICMP Echo requests (0.5%). To identify scanning campaigns, we explored orchestrated scanning activities generated by compromised IoT devices that targeted similar destination ports/services, which represent unique scanning objectives (S_i). Prior to analyzing the scanning objectives, we filtered IoT devices that generated scanning packets less than a pre-determined threshold q. As shown in Figure 5, about 85% of IoT devices were found to scan less than 18 destination ports. Therefore, given the fact that the majority of IoT devices tend to scan a small number of destination ports over the analysis interval, we set the threshold $q = 20$ packets. We leveraged our system to analyze the targeted ports and identified $S' = 9,523$ unique scanned destination port sets (scanning objectives) that were targeted by 14,731 compromised IoT devices.

As shown in the top 10 most common scanning objectives (Table 3), 932 devices (6.3%) were targeting UDP ports 28183, 32124, and 37547, while 835 devices targeted TCP port 445. Moreover, the majority of scanning packets (> 99.5%) sent to ports 28183, 32124, and 37547, were UDP packets, while on the other hand, the remaining ports were almost entirely scanned by TCP packets (e.g., 23, 80, and 5555). From a different perspective, while S_1 was scanned by the largest number of IoT devices, scanning objectives associated with Telnet (e.g., S_7, S_3, and S_4) were scanned by a significantly larger number of packets. This is justified by the fact that Telnet is the most targeted service, especially in the context of compromised IoT devices.

The identified scanning campaigns highlight an important characteristics of the underlying compromised IoT devices, which targeted TCP/UDP ports that might be associated with known vulnerable services. In fact, the identified scanning objectives, which consist of a handful of common TCP services such as Telnet (23/2323), HTTP(80/8080), and HTTPS (443), are reported to be associated with known IoT malware/botnets (e.g., `Mirai`). We also observed other targeted ports that are associated with emerging IoT malware/botnets (e.g., port 5555/`ADB.Miner` (360Netlab, 2018) and port 445/MS-DS and SMB (Seaman, 2018)). Similarly, the remaining TCP ports in Table 3 are all associated with an array of known exploits

Table 3: Top 10 identified scanning objectives (S').

S_i	TCP/UDP Ports	Devices (%)	Packets (M)
1	28183, 32124, 37547	**932 (6.33)**	0.300
2	445	835 (5.67)	7.687
3	23, 80, 8080	735 (4.99)	11.200
4	23, 80, 8080, 37547	403 (2.74)	15.809
5	28183, 32124	209 (1.42)	0.007
6	37547	182 (1.24)	0.015
7	23, 2323	180 (1.22)	**16.849**
8	80, 8080	118 (0.80)	1.122
9	80	100 (0.68)	1.607
10	80, 443, 8080	89 (0.60)	0.019

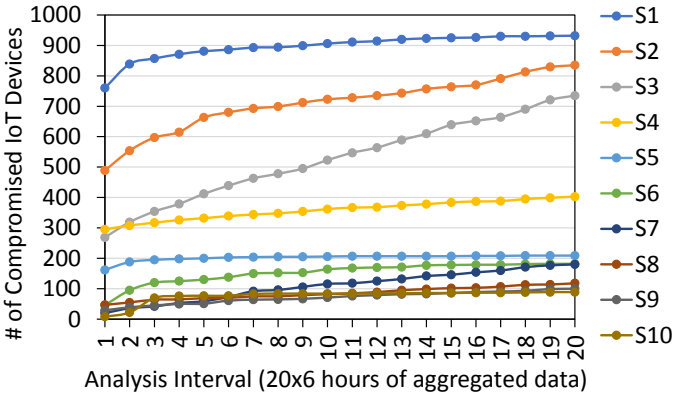
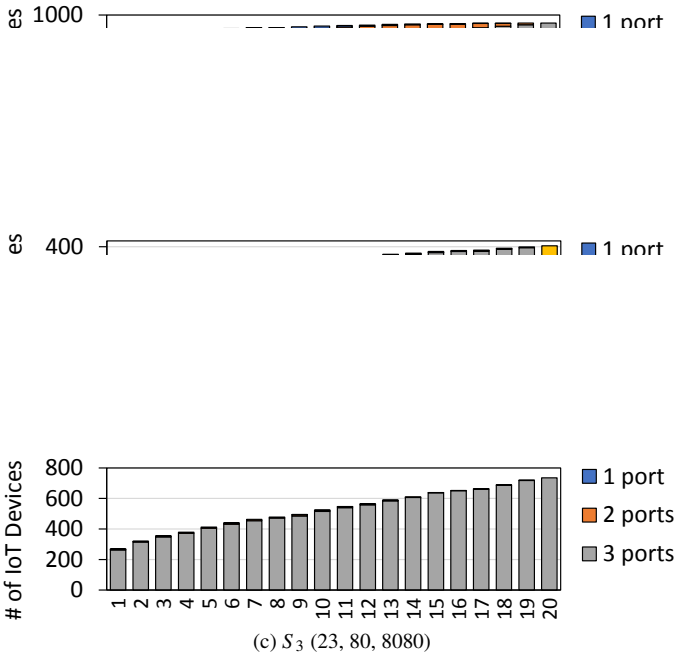


Figure 6: Cumulative number of exploited IoT devices within the top 10 scanning campaigns targeting S_1–S_{10}.

that have been associated with orchestrated scanning activities generated by IoT botnets (Safaei Pour et al., 2019b). On the other hand, a considerable number of IoT devices generated scanning campaigns towards UDP ports (28183, 32124, and 37547), which to the best of our knowledge, are not associated/registered with known services. This however, implies suspicious activities that require further investigation to determine the underlying services and associated exploits (if any).

4.2.4. Temporal Analysis and Campaign Evolution

An important feature of the developed system is to provide the ability to monitor compromised IoT devices and their unsolicited activities over a long period of time. This feature can be used to support operational cyber security research through the identification and inferences of behavioral patterns, while enabling the analysis of temporal changes with respect to the detected scanning campaigns and their evolution over time. For instance, we leveraged the developed system to analyze campaign evolution by finding the cumulative number of compromised IoT devices within the campaigns targeting the top 10 scanning objectives over the analysis interval, as illustrated in Figure 6. While these findings highlight the evolving nature of the campaigns, we also notice variable rates in terms of the number of newly detected IoT devices within the campaigns. For instance, the campaign targeting S_5, reached a steady stage early during the analysis, while the evolution of other campaigns (e.g., S_3) indicates an increasing device discovery trend.



(c) S_3 (23, 80, 8080)

Figure 7: Examples of scanning campaign evolution over the analysis interval (20x6 hours of aggregated data). The campaigns target ports specified in (a) S_1, (b) S_3, and (c) S_4.

The increasing trend is likely to be justified by: (i) the spread of an infection, which exploits further devices over time, and/or (ii) the fact that adversaries may distribute a scanning campaign over controlled IoT botnets, which tend to be active in disperse time intervals while performing partial scanning tasks as part of the bigger campaign.

To investigate the latter, we looked at a sample of scanning campaigns and explored the campaign evolution in terms of the number of scanned ports by the involved IoT devices during each interval. As illustrated in Figure 7a, while the number of exploited IoT devices that scanned all 3 destination ports within S_1 increased gradually by time, a considerable number of them were scanning a subset of the destination ports from S_1 throughout the analysis intervals. Similarly, the majority of devices targeting S_4 were scanning 3–4 ports after each analysis interval (Figure 7b). This indicates that the devices within these two scanning campaigns did not target all specified destination ports at every time interval. Instead, they distributed the task by scanning subsets of the final scanning objective over time, resulting in an evolving scanning campaign that targeted a fixed set of destination ports over a longer period of time.

In contrary, as shown in Figure 7c, almost all devices within the identified scanning campaign were targeting the entire destination ports within S_3 at every interval. This however, reflects the used scanning strategy, which resulted in targeting all three ports during every interval (6 aggregated hours). After all, our implemented data-driven methodology, which is based on identifying scanning campaigns by finding unique scanning objectives over aggregated time periods, was indeed successful in uncovering the campaign intentions, even when the tasks were distributed among multiple devices and/or over several in-

Table 4: Aggregated flow features for device d_i within interval I.

f_i	**Selected Features**
1–3	$U_{i,m}$: number of scanning packets from each type (m)
4	$S_P = \sum_m U_{i,m}$: combined scanning packets
5–7	$\alpha_{i,m}$: discrete prob. dist. representing the fraction of each scanning packet to scans
8	$N^{'}$: number of active intervals (minutes)
9	$A_R = \frac{b_i - a_i}{N_i^{'}}$: activity rate
10	$S_R = \frac{S_P}{N_i^{'}}$: scan rate
11	$\overline{TTL}$: average TTL value
12	$\overline{P}_{size}$: average packet size
13	Src_{Ports}: number of source ports
14	Dst_{IPs}: number of destination IP addresses
15	$Dst_R = \frac{S_P}{Dst_{IPs}}$: per destination packet rate
16	Dst_{Ports}: number of scanned destination ports

Table 5: Clustering results for the top 5 scanning campaigns.

S_i	ϵ	**#Devices**	**#Clusters**	**Clusters' Size (#Outliers)**
1	0.1	932	7	753, 45, 53, 9, 6, 3, 3 (60)
2	0.15	835	7	677, 57, 5, 13, 7, 3, 5 (68)
3	0.15	735	8	659, 3, 3, 3, 3, 3, 3, 3 (55)
4	0.15	403	7	301, 34, 15, 6, 5, 3, 3 (36)
5	0.15	209	2	179, 5 (25)

tervals.

4.2.5. Inferring IoT Botnets

It is important to realize that the identified scanning campaigns in Section 4.2.3 may reflect the behaviors of compromised IoT devices as a part of co-opted botnets, which are utilized to scan a set of predefined ports for vulnerabilities. The assumption is that different exploited IoT devices will produce similar scanning behaviors when infected by the same malware. Moreover, given that IoT malware target specific vulnerable devices, it is likely that these devices share device and firmware-specific features (e.g., TTL values). Therefore, to correlate these devices, the system is utilized to extract aggregated flow features, which represent the overall behaviors of the IoT devices over time. The system also leverages these features towards subsequent clustering of IoT devices within the scanning campaigns to infer groups of correlated IoT devices with similar behavioral characteristics (i.e., IoT botnets).

We leveraged the system to extract 16 features (Table 4), which consist of raw flow information from the IoT-generated packets, along with features related to the aggregated traffic over time. Note that the extracted features can always be modified to add or remove features, if necessary. The system leverages these features in a number of ways to cluster/classify compromised IoT devices into correlated groups. For instance, we leveraged the system to perform sub-space clustering within the identified scanning campaigns to detect IoT devices that produced similar flow features over the entire analysis period. The system utilizes the density-based spatial clustering of applications with noise (DBSCAN) (Ester et al., 1996), which is widely adopted due to the fact that it does not require a priori knowledge about the number of clusters, while it can detect arbitrary shaped clusters and outliers by grouping sufficiently dense regions into clusters in a spatial database (Shah et al., 2012).

The clustering analysis results for the campaigns targeting the top 5 scanning objectives (Table 5) highlight 7 clusters within S_1, with cluster #1 to have the largest number of members (753 out of 932). Similarly, while the analysis revealed variable number of clusters within the remaining groups (S_2–S_5), with each group to contain a main cluster with the largest number of IoT devices. This is not surprising as the majority of devices within the identified groups had similar types and models. Furthermore, given that an IoT malware might in fact target specific types/models of IoT devices, the clustering results will indeed shed light on similarities among the exploited devices based on their correlated behavioral characteristics and aggregated flow features.

4.2.6. Identifying DDoS Victims

Another aspect of monitoring IoT-generated traffic is to identify devices that send backscatter packets towards the darknet. These devices are likely to be victims of DDoS attacks using spoofed IP addresses (Blenn et al., 2017). As summarized in Figure 9, the analysis of backscatter traffic identified 437 IoT devices, among which, the majority were routers (68%). Furthermore, slightly over half of these routers were MikroTik routers, followed by a significantly smaller number of devices from other models. This might be justified by the fact that a considerable number of the routers within the identified DDoS victims were in fact MikroTik routers (59%), as summarized in Table 6. Moreover, the distribution of DDoS victims over the hosting countries, as illustrated in Figure 8, shows that Iran was hosting the largest number of targeted devices in our data, with the majority of these devices to be MikroTik routers (102 out of 106). Considering the fact that our data contained significantly less number of IoT device that were located in Iran, this finding highlights a period of targeted DDoS attacks towards an increasing number of devices located in Iran, as perceived from the darknet.

Overall, the IoT devices (DDoS victims) generated different amount of backscatter packets towards the darknet, with the top 40 victim devices to account for about 93% of all generated backscatter packets. As illustrated in Figure 10, these DDoS victims are found at the high spikes, with device #120 (Radware firewall located in China) to be responsible for generating the largest number of backscatter packets (246K). On the other hand, other DDoS victims, such as device #265 (MikroTik router from Iran), generated relatively fewer number of backscatter packets (<65K). In addition to backscatter packets, about 68% (298/437) of these IoT devices were also generating scanning packets during the analysis intervals. We suspect that these devices were targeted by DDoS attacks while already being involved in scanning activities due to existing exploitations. However, confirming this phenomena is considered for future work.

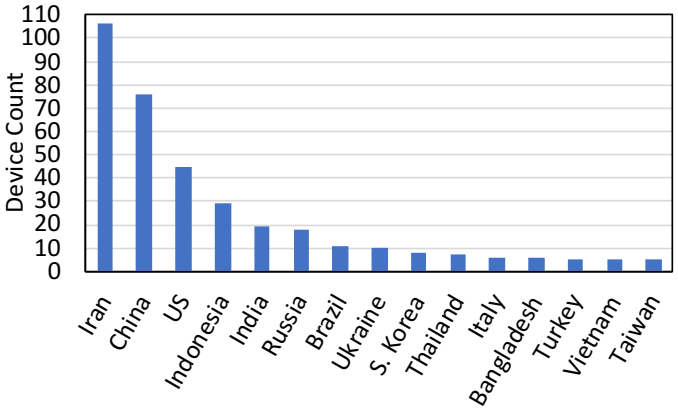


Figure 8: Top 15 countries with the highest number of DDoS victims.

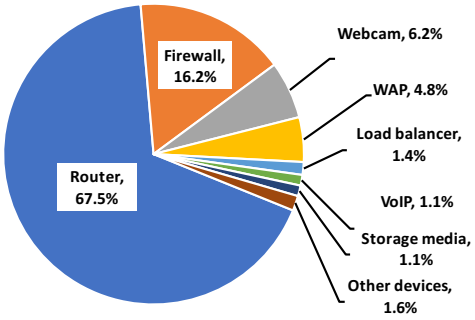


Figure 9: Targeted device types (DDoS victims).

Table 6: DDoS Victims' device models.

Device Model	Count	%
MikroTik router	**258**	**59.0**
SonicWALL firewall	33	7.4
Cisco router	19	4.3
Radware load balancer and ADC	16	3.6
Avtech AVN801 camera	15	3.4
Huawei VRP	14	3.2
WatchGuard firewall	12	2.7
Linksys wireless-G WAP	9	2.0
D-Link DCS webcam	8	1.8
Haproxy load balancer	6	1.4

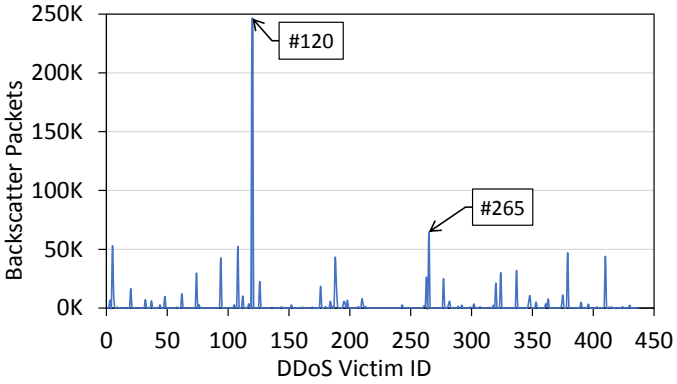


Figure 10: Backscatter packets generated by DDoS victims.

4.3. Performance Evaluation

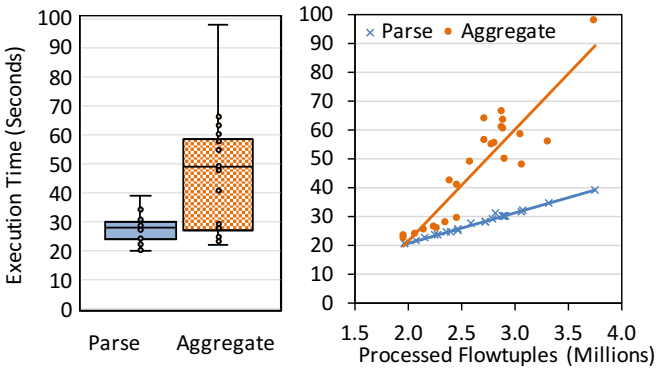
To evaluate the performance and scalability of the system using real life data, we sampled 24 hours of IoT-generated traffic from the collected darknet data, representing a total of 63.5M flows ($mean$ = 2.65M and σ = 4.3M) generated by 13,603 IoT devices ($mean$ = 4061.6 and σ = 183.9). The performance of the system is measured during darknet data parsing, data aggregation, and device profiling processes, as described throughout Sections 3.3.1–3.3.3. In what follows, we provide further information on the performance analysis in terms of execution time and CPU/Memory usage.

4.3.1. Execution Time

The overall execution times required to perform darknet data parsing and IoT data aggregation (Figure 11a) shows that hourly darknet data files were parsed in less than 40 seconds each, with an average of about 27.6 seconds to prepare formatted flowtuple files (min = 20.4s, max = 38.9s, and σ = 4.5s). Moreover, we observe a strong positive correlation ($r \approx 1$) between the required execution time and the number of processed flowtuples in every file, as illustrated by the Least-Squared regression lines in Figure 11b. The regression analysis indicates high accuracy of the model in predicting over 99% of the variance observed in the analyzed data (R^2 = 0.999). This indeed can be used to predict the execution time for parsing a given data file by knowing the number of flowtuples.

Meanwhile, aggregating the parsed flowtuple files required relatively more time (Figure 11a), with an average of 46.7s per file (min = 22.5s, max = 97.7s, and σ = 18.35s). Interestingly, while we also observe a strong positive correlation between the execution time and the number of flowtuples per aggregated file (r = 0.90), the regression analysis indicates that the linear model can describe about 82.5% of the variance in the data (R^2 = 0.825). In other words, the required execution time for the aggregation processes cannot be accurately predicted by the number of flowtuples only as it depends on other factors such as the number of identified IoT devices and their associated flowtuples per analysis interval. These factors can indeed invoke a series of Spark operations (e.g., `groupBy()` and `agg()`) on subsets of data with variable length, resulting in further processing and execution overhead, respectively.

In addition, we analyze the execution time required for creating the dynamic device profiles at the end of every hourly analysis interval (recall Section 3.3.3). Device profiles are expected to grow in terms of the number of records (IoT devices) over an accumulative period of time as they depend on merging the aggregated IoT device information at any interval with previously obtained device profiles. This result in increasing the required execution time by a range between 1–59 minutes for intervals 1 to 24, as shown in Figure 12, respectively. In fact, the correlation analysis indicate a strong positive correlation that is modeled almost accurately by an exponential linear regression line (R^2 = 0.99). Given that we performed our experiments on



(a) (b)

Figure 11: Execution time analysis for (a) data parsing and aggregation, and (b) the correlation of execution times to the number of flowtuples in parsed/aggregated data files.

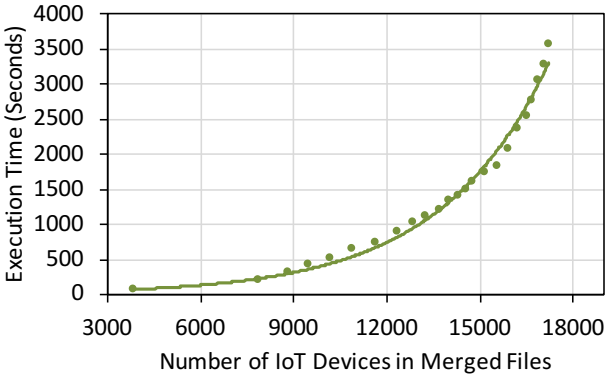


Figure 12: Correlation of execution time with the accumulative number of IoT devices in the merged data files.

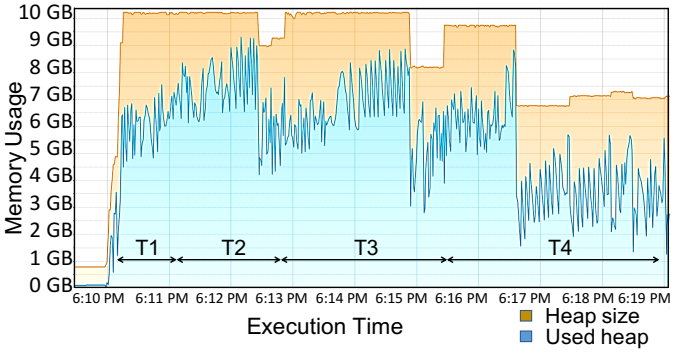
a single node implementation of `Apache Spark`, we anticipate to produce a linear correlation between the execution times and the number of devices in the merged files by implementing the system on a cluster of nodes, which will result in shorter execution times over the accumulated IoT devices.

4.3.2. CPU and Memory Usage

We analyzed the CPU and memory usage for different parts of the system. The darknet data parser is used for reading flowtuples from input files, parsing them and writing parsed flowtuples back into output files. These I/O operations tend to be CPU intensive and can usually use maximum CPU power. On the other hand, the memory usage for the darknet data parser stayed almost constant, with about 88MB of needed memory throughout the operations.

Moreover, we summarize the analysis of the CPU and memory usage for the data aggregation and profiling processes by illustrating the results for a sample of four consecutive aggregated and then profiled hourly darknet data files, as shown in Figures 13 (a–b). At every hourly time interval (T1–T4), the operations start by reading large amount of data from input files, followed by aggregation and merging (profiling) operations, which result in intensive CPU usage (Figure 13a). At the

(a) CPU usage



(b) Memory usage

Figure 13: CPU and memory usage for a sample of four consecutive hours of aggregated/profiled darknet data.

end of every interval, the data/results are written back to output files (JSON), which justifies the noticeable drop in the CPU usage (< 30%) during the write operations. The sequence of operations is repeated at every hourly interval, which explains the recurring high/low CPU usage throughout the analysis. More importantly, due to the accumulated number of detected IoT devices after every analysis interval, the device profiling/merging operations required more time to process the data, as observed by the extended intervals of intensive CPU usage in Figure 13a. In addition, while the heap size was set to 10GB for this experiment, the memory usage stayed below 9.5GB, with an average of about 6.7GB of used memory over the analysis interval (Figure 13b). This is in fact very reasonable due to the size of the input data files and the number of processed flowtuples during the aggregation and device profiling operations.

4.4. Limitations and Future Work

The generalizability of our findings might be hampered by the size of the IoT device sample and the collected darknet data over the analysis period (5 days). In addition, we rely on external resources of IoT device information and passive network measurements. Nevertheless, both `Shodan` and the UCSD network telescope are considered among the largest and most reliable sources of data available for research purposes. In addition, we can overcome these limitations by performing long term data collection and analysis experiments, which can produce results in near real-time. Furthermore, we have implemented an Internet-scale IoT device scanning and banner anal-

ysis tool that mimics Shodan, while providing similar IoT device information that could be used for validating and extending our knowledge of deployed IoT devices. Additionally, we have already started collaborating with other darknet data providers in order to expand our data, while providing means for comparing the compromised IoT device behaviors over an extended subset of the IPv4 address space. Finally, while we provide information on the design of the IoT threat repository in Section 3.4, the integration and evaluation of this module is considered for future work, with the aim to make such repository publicly available at large, to strongly support IoT-centric forensics and Internet-scale remediations.

5. Conclusion

In this paper, we contribute towards empirical IoT forensics by designing, developing, and thoroughly evaluating a scalable infrastructure to enable the development of supporting technologies that help in building a better understanding of compromised IoT devices and their unsolicited activities. The developed system, which leverages the power of big data analytics frameworks, was utilized to process more than 4TB of passive network traffic collected at a large-scale network telescope (darknet) to identify 27,849 compromised IoT devices that generated more than 300 million unsolicited packets. Furthermore, we demonstrate the effectiveness of the system through a number of applied security operations to infer and fingerprint IoT-generated activities, which enable future work towards IoT-centric remediation, cyber-situational awareness, malware detection and evolution, to name a few. Finally, while the performance evaluation shows that the system can indeed execute large-scale data analysis effectively and efficiently, the implemented system is also scalable by design, as it can be extended through the implementation of Apache Spark on a multi-node cluster architecture.

Acknowledgments

The authors would like to express their sincere gratitude to the anonymous reviewers for their constructive feedback. The work has been supported by Natural Sciences and Engineering Research Council of Canada (NSERC) and Concordia University. The work was also partially supported by a grant from the U.S. National Science Foundation (NSF), Office of Advanced Cyberinfrastructure (OAC) #1907821.

References

360Netlab, 2018. ADB.Miner: More Information [Blog post]. Retrieved from https://blog.netlab.360.com/adb-miner-more-information-en/.

Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J.A., Invernizzi, L., Kallitsis, M., Kumar, D., Lever, C., Ma, Z., Mason, J., Menscher, D., Seaman, C., Sullivan, N., Thomas, K., Zhou, Y., 2017. Understanding the Mirai Botnet, in: 26th USENIX Security Symp., Vancouver, BC. pp. 1093–1110.

Blenn, N., Ghiëtte, V., Doerr, C., 2017. Quantifying the Spectrum of Denial-of-Service Attacks Through Internet Backscatter, in: Proc. of the 12th Int. Conf. on Availability, Reliability and Security, Reggio Calabria, Italy. pp. 21:1–21:10.

CAIDA, 2019. The CAIDA UCSD Real-Time Network Telescope Data. UCSD - Center for Applied Internet Data Analysis. Retrieved from http://www.caida.org/data/passive/telescope-near-real-time_dataset.xml.

Chen, D.D., Woo, M., Brumley, D., Egele, M., 2016. Towards Automated Dynamic Analysis for Linux-based Embedded Firmware, in: Proc. of the Network and Distributed Syst. Security Symp. (NDSS).

Cimpanu, C., 2018. Hajime Botnet Makes a Comeback With Massive Scan for MikroTik Routers. Retrieved from https://www.bleepingcomputer.com/news/security/hajime-botnet-makes-a-comeback-with-massive-scan-for-mikrotik-routers/.

Costin, A., Zaddach, J., Francillon, A.é., Balzarotti, D., Antipolis, S., 2014. A large-scale analysis of the security of embedded firmwares, in: In 23rd USENIX Security Symp., pp. 95–110.

Cui, A., Stolfo, S.J., 2010. A quantitative analysis of the insecurity of embedded network devices: results of a wide-area scan, in: Proc. of the 26th Annual Comput. Security Applicat. Conf., ACM. pp. 97–106.

Durumeric, Z., Adrian, D., Mirian, A., Bailey, M., Halderman, J.A., 2015. A Search Engine Backed by Internet-Wide Scanning, in: 22nd ACM Conf. on Computer and Commun. Security.

Durumeric, Z., Bailey, M., Halderman, J.A., 2014. An Internet-Wide View of Internet-Wide Scanning, in: Proc. of the 23rd USENIX Security Symp., San Diego, CA. pp. 65–78.

Ester, M., Kriegel, H.P., Sander, J., Xu, X., et al., 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, in: Proc. of KDD, pp. 226–231.

Fachkha, C., Bou-Harb, E., Keliris, A., Memon, N., Ahamad, M., 2017. Internet-scale Probing of CPS: Inference, Characterization and Orchestration Analysis, in: Proc. of the Network and Distributed Syst. Security Symp. (NDSS'17), San Diego, California.

Feng, X., Li, Q., Wang, H., Sun, L., 2018. Acquisitional Rule-based Engine for Discovering Internet-of-Things Devices, in: 27th USENIX Security Symp., pp. 327–341.

Fernandes, E., Paupore, J., Rahmati, A., Simionato, D., Conti, M., Prakash, A., 2016. FlowFence: Practical Data Protection for Emerging IoT Application Frameworks, in: 25th USENIX Security Symp.

Guarnizo, J.D., Tambe, A., Bhunia, S.S., Ochoa, M., Tippenhauer, N.O., Shabtai, A., Elovici, Y., 2017. Siphon: Towards Scalable High-Interaction Physical Honeypots, in: Proc. of the 3rd ACM Workshop on Cyber-Physical Syst. Security, ACM. pp. 57–68.

Herwig, S., Harvey, K., Hughey, G., Roberts, R., Levin, D., 2019. Measurement and Analysis of Hajime, a Peer-to-peer IoT Botnet, in: Proceedings of the Network and Distributed System Security Symposium (NDSS).

Jia, Y.J., Chen, Q.A., Wang, S., Rahmati, A., Fernandes, E., Mao, Z.M., Prakash, A., Unviersity, S.J., 2017. ContexIoT: Towards Providing Contextual Integrity to Appified IoT Platforms, in: Proc. of the Network and Distributed Syst. Security Symp. (NDSS'17).

Labovitz, C., Ahuja, A., Bailey, M., 2001. Shining Light on Dark Address Space. Arbor Networks Inc.

Luo, T., Xu, Z., Jin, X., Jia, Y., Ouyang, X., 2017. IoTCandyJar: Towards an Intelligent-Interaction Honeypot for IoT Devices, in: Blackhat.

Moore, D., Paxson, V., Savage, S., Shannon, C., Staniford, S., Weaver, N., 2003. Inside the Slammer Worm. IEEE Security & Privacy , 33–39.

Neshenko, N., Bou-Harb, E., Crichigno, J., Kaddoum, G., Ghani, N., 2019. Demystifying IoT Security: An Exhaustive Survey on IoT Vulnerabilities and a First Empirical Look on Internet-Scale IoT Exploitations. IEEE Communications Surveys & Tutorials 21, 2702–2733.

Pa, Y.M.P., Suzuki, S., Yoshioka, K., Matsumoto, T., Kasama, T., Rossow, C., 2016. IoTPOT: A Novel Honeypot for Revealing Current IoT Threats. J. of Inform. Process. 24, 522–533.

Ronen, E., Shamir, A., 2016. Extended Functionality Attacks on IoT Devices: The Case of Smart Lights, in: IEEE European Symp. on Security and Privacy (EuroS&P), IEEE. pp. 3–12.

Sachidananda, V., Siboni, S., Shabtai, A., Toh, J., Bhairav, S., Elovici, Y., 2017. Let the Cat Out of the Bag: A Holistic Approach Towards Security Analysis of the Internet of Things, in: Proc. of the 3rd ACM Int. Workshop on IoT

Privacy, Trust, and Security, pp. 3–10.

Safaei Pour, M., Bou-Harb, E., Varma, K., Neshenko, N., Pados, D.A., Choo, K.K.R., 2019a. Comprehending the IoT cyber threat landscape: A data dimensionality reduction technique to infer and characterize Internet-scale IoT probing campaigns. Digital Investigation 28, S40–S49.

Safaei Pour, M., Mangino, A., Friday, K., Rathbun, M., Bou-Harb, E., Iqbal, F., Shaban, K., Erradi, A., 2019b. Data-driven Curation, Learning and Analysis for Inferring Evolving IoT Botnets in the Wild, in: Proc. of the 14th Int. Conf. on Availability, Reliability and Security (ARES 2019), pp. 6:1–6:10.

Seaman, C., 2018. UPNPROXY: ETERNALSILENCE. Retrieved from `https://blogs.akamai.com/sitr/2018/11/upnproxy-eternalsilence.html`.

Shah, G.H., Bhensdadia, C., Ganatra, A.P., 2012. An Empirical Evaluation of Density-Based Clustering Techniques. Int. J. of Soft Comput. and Eng. (IJSCE) 22312307, 216–223.

SHODAN, 2019. Shodan. Retrieved from `https://www.shodan.io/`.

Spark, 2019. Spark DataFrame API. Retrieved from `https://spark.apache.org/docs/latest/api/python/pyspark.sql.html#pyspark.sql.DataFrame`.

Torabi, S., Bou-Harb, E., Assi, C., Galluscio, M., Boukhtouta, A., Debbabi, M., 2018. Inferring, Characterizing, and Investigating Internet-Scale Malicious IoT Device Activities: A Network Telescope Perspective, in: Proc. of the 48th Annual IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN), pp. 562–573.

Ur, B., Jung, J., Schechter, S., 2013. The Current State of Access Control for Smart Devices in Homes, in: Workshop on Home Usable Privacy and Security (HUPS).

Vervier, P.A., Shen, Y., 2018. Before Toasters Rise Up: A View into the Emerging IoT Threat Landscape, in: Int. Symp. on Research in Attacks, Intrusions, and Defenses, Springer. pp. 556–576.

Yu, T., Sekar, V., Seshan, S., Agarwal, Y., Xu, C., 2015. Handling a trillion (unfixable) flaws on a billion devices: Rethinking network security for the Internet-of-Things, in: Proc. of the 14th ACM Workshop on Hot Topics in Networks, ACM. p. 5.