SIEM-based detection and mitigation of IoT-botnet DDoS attacks

Basheer Al-Duwairi, Wafaa Al-Kahla, Mhd Ammar AlRefai, Yazid Abdelqader, Abdullah Rawash, Rana Fahmawi

Department of Network Engineering and Security, Jordan University of Science and Technology, Irbid, Jordan

Article Info	ABSTRACT
	The Internet of Things (IoT) is becoming an integral part of our daily life including health, environment, homes, military, etc. The enormous growth of IoT in recent years has attracted hackers to take advantage of their computation and communication capabilities to perform different types of attacks. The major concern is that IoT devices have several vulnerabilities that can be easily exploited to form IoT botnets consisting of millions of IoT devices and posing significant threats to Internet security. In this context, DDoS attacks originating from IoT botnets is a major problem in today's Internet that requires immediate attention. In this paper, we propose a Security Information and Event Management-based IoT botnet DDoS attack detection and mitigation system. This system detects and blocks DDoS attack traffic from compromised IoT devices by monitoring specific packet types including TCP SYN, ICMP and DNS packets originating from these devices. We discuss a prototype implementation of the proposed system and we demonstrate that SIEM based solutions can be configured to accurately identify and block malicious traffic originating from compromised IoT devices.

Corresponding Author:

Basheer Al-Duwairi,
Jordan University of Science and Technology,
Irbid 22110, Jordan.
Tel: +962-02-7201000 (Ext. 24000)
Email: basheer@just.edu.jo

1. INTRODUCTION

Internet of things (IoT) have witnessed enormous growth in recent years. This growth is due to the significant development in semiconductor industry, wireless communication technologies and the emergence of IoT applications in different fields in our daily life including: smart homes [1], health care sector [2], industrial control systems [3] and military applications [4]. According to Statista [5], the number of IoT devices is estimated to be about 23.4 billion and is expected to increase to 31 billion devices in 2020.

Motivated by the significant growth of IoT, hackers are eager to take advantage of the computation and communication capabilities of IoT devices to perform different types of attacks. Botnets represent the primary source of many attacks targeting the Internet. A botnet is a network of compromised machines (e.g. computers, smartphones, IoT devices, etc.) which are controlled by an attacker (botmaster) and used to perform a variety of activities such as distributed denial of service (DDoS) attacks, spamming, click fraud, and phishing attack. IoT devices are creating a huge risk to security compared to the ordinary Internet connected devices because of their large number, resource limitation and protocol diversity, which makes IoT a valuable target for attackers to create IoT botnet. For example, an attacker can compromise victim's IoT smart home devices [1] to obtain sensitive information, or to use IoT devices as IoT Botnets [6] in order to execute a distributed denial-of-

service attacks (DDos) against critical cyber-physical systems. On October 21, 2016, Attackers launched a DDoS attack against Dyn DNS service [7] by utilizing Mirari IoT botnet causing many services and internet platforms to be unavailable in Europe and North America.

Previous research efforts have focused on devising mechanisms to detect vulnerable or compromised IoT devices or to secure IoT devices despite of their computing, storage and power limitations. IoT devices usually exchange different types of messages and generate huge amount of data making it difficult to monitor and correlate events and detect malicious activities originating from them. In this paper, we address the problem of DDoS attacks originating from IoT botnets by using Security Information and Event Management (SIEM) solution in such a way to detect and block attack traffic. The main contributions of this paper are as follows:

(a) A SIEM-based IoT botnet detection system. The proposed system adopts Splunk SIEM solution [8] to identify and block traffic from compromised IoT devices by monitoring specific packet types including TCP SYN, ICMP and DNS flooding.

(b) The implementation of a prototype of an IoT botnet detection system based on the proposed Security Information and Event Management (SIEM) solution, and application of the proposed system to detect three common DDoS attack types.

(c) Review of different aspects of IoT botnets focusing mainly on IoT botnet architectures, main types, countermeasures, and discusses recent research efforts in this field.

The rest of this paper is organized as follows: background information about IoT botnets and presents main features and limitations of IoT devices are presented in Section 2. Related work is discussed in Section 3. The proposed system and the prototype implementation of the SIEM-based IoT Botnet DDoS attack detection and mitigation is discussed in Section 4. Evaluation of the proposed system is presented in Section 5. Finally, conclusions are presented in Section 6.

2.   BACKGROUND ABOUT IOT BOTNETS

Recently, IoT botnets have emerged as a new type of botnets where hackers control millions of compromised IoT devices and utilize them in their malicious attacks. Generally, an IoT botnet consists of the following components:

(a) Bots: compromised IoT devices used to perform the different types of attacks (e.g., denial of service).
(b) Command and Control Server: Server used to control the Bots.
(c) Scanners: used to scan for vulnerable IoT devices.
(d) Reporting Server: Server used to collect scanning reports from scanner or Bots.
(e) Loader: instruct IoT devices to download the malware code after logging at the vulnerable IoT devices.
(f) Malware Distribution Server: A server responsible for malware distribution.

IoT devices have been widely utilized in many large-scale DDoS attacks. In these attacks, IoT devices are instructed by the botmaster to flood a target system with huge amount of packets (e.g. SYN, DNS, ICMP, etc.). Zhou et al [9], discussed several features of IoT devices that contributes to the problem of IoT botnets. These features include:

(a) **Constrained** Compared to traditional PCs and mobile devices, IoT devices usually have limited computational capabilities and limited storage resources in order reduce manufacturing costs and to save battery power as much as possible. For example, IoT devices deployed in a military battle field or an earth quick area are required to work for a long duration. Therefore, power consumption of these devices should be managed effectively and power-consuming applications can not run on such devices in order to increase battery life time. On the other hand, IoT devices used in the medical sector are time-constrained. Therefore, time-consuming applications can not run on such devices in order to avoid delay in the response. Because of such limitations, we cannot deploy the required defense systems on these devices such as complex encryption and authentication techniques that consume computing and storage resources, and thus create many vulnerabilities that can be exploited by hackers.

(b) **Diversity** IoT technology is increasingly used in many real life applications. As a result, there are many IoT vendors with large number of heterogeneous IoT products that have their own network and communication protocol implementations. In most cases, many of these products do not have security included in their design . According to a recent HP report [10], there are several problems with IoT devices deployed in today's Internet. This includes lack of encryption, failing to change default passwords,

and lack of software updates. The heterogeneity and diversity of IoT networking and communication protocols makes it difficult to secure IoT devices manufactured by different vendors or to eliminate their vulnerabilities.

(c) **Myriad** This feature is related to the substantial increase of IoT devices and the proliferation of data generated by them. This feature combined with the lack of security protection on IoT devices and their limited computational power allowed attackers to compromise these devices easily and to create armyies of IoT zombies that could be used to conduct several attack types. Yin et al [11] found that most recent large-scale DDoS attacks are mainly caused by IoT botnets.

(d) **Unattended** In many cases, IoT devices are deployed in harsh environments and remain unattended for long period of time. Usually, there is no mechanism to remotely access these devices and check their security level or perform regular updates on them. This has provided adversaries with additional reason to target these devices and control them with minimal effort. For example, Ronen et al [12] implemented an attack on Philips's smart light bulb by exploiting vulnerability in the communication protocol and showed how fast the developed worm can spread to other bulbs.

(e) **Mobility** Wearable devices and smartphones are examples of a mobile IoT devices that join and leave networks dynamically according to the human movement. The mobility of IoT devices imposes additional security threats because the chances of malware infection increase whenever a device becomes within the vicinity of another malicious device. Also, it increase hackers' chances to compromise IoT devices.

3. RELATED WORK

There have been substantial amount of work in the area of IoT security. In this section, we provide a review of the recent advances in this field focusing on detecting IoT devices' vulnerabilities, detecting IoT network vulnerabilities, and detecting IoT botnets.

3.1. Detecting IoT devices' vulnerabilities

Static analysis of IoT devices' frameworks is one of the techniques used to detect IoT devices' vulnerabilities. Fore example, Costin et al. [13] performed a large-scale security analysis of 32 thousand of firmware images of embedded devices, and found 38 unknown vulnerabilities in more than 693 firmware images which extend in 123 different products. Fernandes et al. [14] performed static source code analysis and crafted tests on 123 smart Home platforms and 499 smart things applications. They discovered two flaws in the design of these platforms. These design flaws lead to grant Smart Things Application over privileged rather than a separated privilege as designed, beside the full access of these applications to the Smart Home Device rather than the limited access as designed. They also proved that the asynchronous communication (event subsystem) between devices and Smart Applications is not secure and could leak sensitive information.

In [12], Ronen et al. described how to exploit a vulnerability in the implementation part of Zigbee light protocol of Philips Hue smart lamps to perform a remote firm-ware update. Once the vulnerability is exploited, the key used by Philips smart lamps was extracted by performing side channel attack. This key is used to encrypt and authenticate the update. After that, the worm rapidly spread from the infected lamp to the other lamps using their ZigBee wireless connectivity. Such attack enables an attacker to control city lighting or to exploit the lamps to perform DDoS attacks. Several research efforts have been made to develop Internet Wide Scan technologies to search for vulnerable IoT devices. Kim et al [15] proposed a model to improve the performance of the Internet Wide Scanner Zmap [16].

3.2. Detecting IoT network vulnerabilities

There are several approaches for detecting IoT network vulnerabilities. For example, IoT network protocol vulnerabilities can be detected using Fuzzing-based approach. In this approach, software implementation of a given protocol is tested by observing any resulting exception when false data is submitted intentionally to test the software. Fuzzing-based approach includes two main Fuzzing techniques: (i) Mutation-based: In this technique, false data is injected randomly in testing messages (ii) Generating-based: test files are mainly generated by constructing messages that adhere to the protocol specifications. However, it contains random and false data. The main problem of this method is that it requires large number of test files, especially that each protocol has its own message format. Therefore, consuming considerable amount of time. Luo et al [17], proposed a technique that performs reverse engineering of IoT protocols, to identify the message format of

protocols and create test file messages with certain faults according to the message format. This method reduces the size of test files used in the Fuzzing approach.

Jia et al [18] verified the effectiveness graph-based analysis on a smart home system prototype. The prototype consisted of a smartphone controlling a smart light pulp and a Google home speaker. The main idea of their approach is to construct a traffic graph based on certain input files, and to identify correlated sub-graphs. This allowed them to identify the vulnerabilities based on the sensitivity level of different keywords. They also demonstrated how to exploit the vulnerable sup-graphs to conduct different attacks. Li1 et al [19] analyzed the traffic of encrypted video stream for Video Surveillance Systems and observed that the traffic pattern is different for different user activity. This means that it is possible for an attacker to infer user's information by analyzing the traffic size and rate even in case the traffic is encrypted. Apthorpe et al [20] examined the network traffic rate of several IoT devices and found that passive network observers, could analyze the network traffic and infer sensitive information.

3.3. Detecting IoT botnets

Previous work on the detection of IoT botnets can be categorized into: anomaly-based, signature-based, specification-based, and hybrid-based detection. What follows is a discussion of each approach.

3.3.1. Anomaly-based

This approach detects IoT botnet by recognizing abnormal behavior in the network. To achieve this goal, it is required to profile the normal behavior of the IoT network in advance. Summerville et al, [21] developed an ultralight packet anomaly-based method to detect abnormal payload in the packet, using efficient matching technique for bit pattern requires only ADD operation followed by incremental counter, and implemented as a look up table for fast and flexible packet evaluation. Sagirlar et al, [22] proposed AutoBotCatcher, which utilizes Block chain concept to detect decentralized P2P Botnets. Based on the fact that IoT bots within the same botnet usually communicate with each other, AutoBotCatcher is designed to detect botnet devices and label them as one community by analysing network traffic exchanges between different devices. AutoBot-Catcher uses Agents to monitor traffic exchanged between IoT devices. These agents report the information they collect as a block chain transaction to a big trusted entity in the network called the block generator, which models mutual contact information of IoT device as a mutual contact graph. Then it uses Louvain method [23] to detect botnet community based on the graph.

3.3.2. Signature-based

P. Ioulianou et al [24] proposed a signature-based IoT botnet detection system that utilizes Intrusion Detection System (IDS) technology that is typically used to monitor networks for known malicious activities and policy violations based on matching attack signatures. In their system, the IDS modules are configured to work in a hybrid mode. The detection and firewall module called Router IDS and the monitoring lightweight module called Detector IDS. These modules are distributed in the network close to IoT devices which does not require any software modification on sensors or devices. Detector IDS logs network traffic and sends it to the Router IDS that will detect malicious node behavior if it resembles to a known attack. Khoshhalpour et al, [25] proposed a host-based approach called BotRevealer to detect IoT botnet in the early infection step, using botnet life cycle as a general signature for detection. They analyze the running process and network activities on the host based on statistical features of packet sequence and compare it with the behavior pattern of botnet traffic.

3.3.3. Specification-based

This approach is similar to anomaly-based approach but it takes into account system specifications. Carli et al, [26] proposed an automatic interference technique for the specifications of malware network protocol using samples of malware communication and malware binaries. Since each malware has its own custom binary format and each C&C protocol has its own malware family, this will provide a fingerprint for the malware structure and intent. They proposed a type system field of the message that describes all field types in the message, and then use type interference algorithm to interfere message structure. However, most C&C network traffic is encrypted so they apply dynamic traffic analysis to extract C&C system keys. Prokofiev et al [27] proposed a detection technique for IoT botnets during malware propagation stage where infected devices starts to exploit other devices in the network using brute force attack strategy.

3.3.4. Hybrid-based

Hybrid-based IoT botnet detection usually employs two detection approaches. For exampe, signature-based IoT botnet detection can be combined with anaomaly-based IoT botnet detection or with specification-based approach. This has the advantage of minimizing false positive and false negative rates of the detection system. Sedjelmaci et al [28] proposed a low energy consumption anomaly and signature based IoT botnet detection system and using game theory techniques to decide whether an IDS agent is required to activate anomaly detection or no. Therefore, increasing detection accurcy while reducing the power consumption in IoT device. Another hybrid-based detection system was proposed by Bostani et al [29]. This system combines anomaly-based and specification- based intrusion detection models to detect attacks in IoT. The specification-based detection agent will be located on routers nodes; it will analyze the host node behaviour and send the results to the root node where the anomaly-based detection agent is located. This agent is based on the Map reduce architecture and employs optimum path algorithm using the data sent by routers to project clustering model and detect malicious behaviour using voting mechanism.

4.   PROPOSED WORK: SIEM-BASED DETECTION AND MITIGATION OF IOT BOTNET DDOS ATTACKS

In this section, we present the proposed SIEM-based detection and mitigation of IoT botnet DDoS attacks, focusing on the system overall architecture and a prototype implementation.

4.1.   System architecture

SIEM systems are primarily used in the security field to correlate events reported by various network security defense technologies (e.g., intrusion detection systems, firewalls, bring your own device solutions, operating systems syslogs, etc.) deployed within an Enterprise network. The results of event correlation indicate weather there is a security incident or no. There are few recent research studies about the use of SIEM solutions for IoT security (e.g., [30, 31]). These studies focused mainly on efficient delivery of IoT data to the SIEM system for analysis and correlation of events.

Figure 1 depicts the basic architecture of the proposed system. First, IoT traffic logs are forwarded by the default gateway to the SIEM system. These traffic logs are obtained from various IoT devices in the monitored network including IP cameras, fingerprint readers, building management system sensors, etc. The SIEM solution performs a sequence of data processing tasks that include parsing, indexing, and storing these logs in a highly available secure database. The logs are then analyzed and in case there is any abnormal behavior compared to the traffic profile of the device in question, it detects an attack and alerts the network administrator.

DDoS attacks are generally characterized by a high packet volume. Therefore, the detection of DDoS attacks in our system is based on comparing the number of packets of certain type (e.g., SYN, ICMP, or DNS) that are destined to a certain machine to a predefined threshold value. Once an attack is detected, SIEM mitigates the ongoing attack by automatically configuring the firewall application installed on the default gateway such that new rules are added to block attack traffic. There are multiple SIEM platforms that can gather machine data.
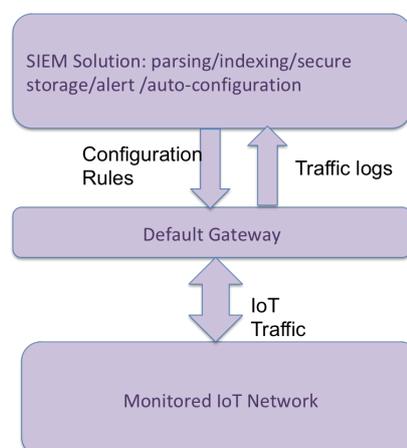

Figure 1. System architecture

4.2. Prototype implementation

An enterprise network (e.g., a campus network) usually has different types of IoT devices such as IP cameras, temperature sensors, fingerprint attendance system, etc. Monitoring these devices individually would be difficult because of traffic heterogeneity. In this paper, we have implemented a prototype of an IoT botnet detection system based SIEM solution. Our goal is to show that it is possible to detect different types of malicious traffic originating from various IoT devices. A prototype implementation of the proposed systems is shown in Figure 2. This prototype consists of the following main components:
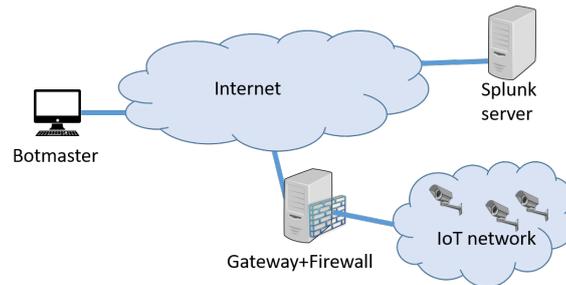
Figure 2. Network topology of the SIEM-based IoT botnet detection system prototype

(a) **IoT Botnet:** In any organizational network, there are different types of IoT devices that are vulnerable to botnet infection. In our prototype, we represented the IoT botnet by Raspberry Pi v1 which is an open source hardware platform that can be used for special purpose IoT devices. We installed a bot code written in Python scripting language on these devices to generate different types of attack traffic that include SYN flooding, DNS flooding, and ICMP flooding. In addition, we used a Cisco 2520V camera with framework Cisco Video Surveillance 2421 IP Dome camera in order to generate background IoT traffic. Bots receive commands in the format: (*type, count, IP, data*), where type represent attack packet type (e.g., SYN, DNS, or ICMP), count specifies the number of packets to be sent in case of flooding attacks, IP represents the IP address of the targeted system, and data specifies the port number if in case of SYN flooding attack, or the domain in case of DNS attack, not used in ping scan. Each bot runs a Python script to conduct the attack based on the parameters assigned in the command received from the botmaster.

(b) **Gateway:** We configured a Linux machine to work as the default gateway of the IoT devices. The machine has two network interfaces. One Interface is facing the Internet and the other one is facing the local network where IoT devices are located. We run tcpdum on this machine in order to capture IoT traffic. We used the command:

bash $tcpdump- -n- -e- -i\backslash interface" > logfile.log$

to capture all outgoing traffic and saving it in a logfile to be forwarded to Splunk server periodically. Here we used the options -n and -e in order not to convert the IP address and to include MAC addresses in the traffic capture, respectively. We installed and configured the Splunk forwarder on the gateway to forward the generated traffic log file to the Splunk server. We specified the source type and index on the forwarder as the same source type and index we defined on the Splunk server. The server was configured as shown in Figure 3, where port 9997 was used to receive traffic logs and port 8089 for management. In addition, we used IPTABLES to add specific rules preventing abnormal traffic generated from a specific IoT device and targeting certain machine.
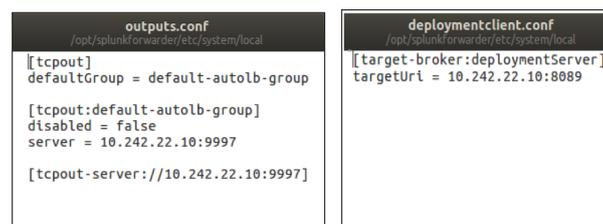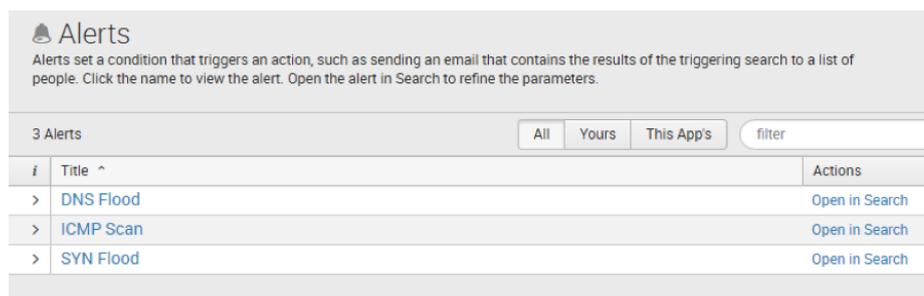

Figure 3. outputs.conf file and deploymentclient.conf file

2188 □ ISSN: 2088-8708

(c) **SIEM Solution:** We used the well known Splunk SIEM solution [8] to analyze IoT traffic collected through the gateway. Splunk was installed on a standalone server and was configured to present the collected traffic logs in a readable and searchable way. This required us to extract certain fields from the IoT traffic logs and present it in Splunk readable format. The Splunk system represents the core of our IoT botnet detection prototype. In this regard, collected traffic logs were parsed, indexed, and stored in a secure database designed only for high availability and real-time analysis. Analyzing this traffic allowed us to understand the behavior of the monitored device. Moreover, Splunk was configured to alarm the network administrator about suspicious events and to automatically add defensive rules to the firewall in order to block attack traffic originating from infected IoT devices.

(d) **Firewall:** We used IPTABLES to implement the firewall where the SIEM is configured to add specific rules to block certain traffic types in a fully automated fashion based on the IoT traffic log analysis. Adding/removing rules is done through an SSH connection between the SIEM and the firewall.

4.3. Attack alert generation

Splunk SIEM platform was configured to generate three types of alerts in order to notify network administrator once an attack is detected. As shown in Figure 4, these alerts are mainly for SYN-flood attack, DNS-flood attack and alert for ping scan. An alert was configured by defining a specific search that is based on packet headers found in the periodically received traffic logs.

Figure 4. Main alerts defined in Splunk platform

For example, in the case of SYN-flooding attack, an alert is generated whenever the SIEM platform detects that there there is large number of SYN packets targeting certain system. SIEM reads the traffic log continuously and search about SYN packets originating from a given source and targeting a given destination (i.e., a one to one relationship) and count them in a specific period of time, if the number of packets exceeds a predefined threshold value, the alert will send a notification email for the administrator about the ongoing attack. The search of this alert is shown in Figure 5. This search command is mainly to search about packets with SYN flag set in the traffic logs that came from the index `traffic_idx` and sort them based on (Src_MAC, Src_IP and Dst_IP) and count them. Matching results are inserted in a special table, and all entries with count less than 100 are removed from the table. Finally, the fields (Src_MAC, Src_IP and Dst_IP) are extracted to be included in the attack notification email as shown in Figure 6.
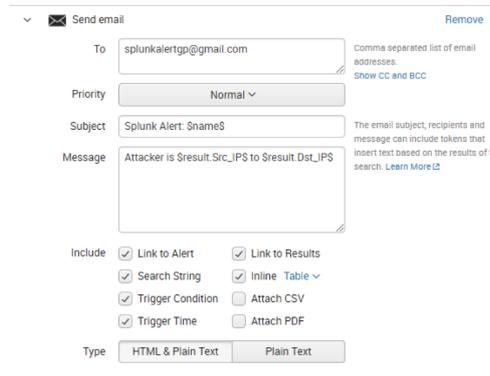

Figure 5. The search command of SYN flood alert

Figure 6. Configuration of attack notification email details

5.   EVALUATION

The prototype implementation described in Subsection 4.2. was used to test the functionality of the proposed system. In this prototype, IoT bots were instructed to flood a targeted system with different types of attack packets. Then, IoT traffic logs were forwarded periodically to the Splunk server. While Splunk platform comes with a predefined source types (e.g., syslog, apachelog, etc.), the IoT traffic log captured by tcpdum could not be recognized by Splunk. Therefore, we defined a new source type called "tcpdump_traffic". Defining a new source type can be done by creating a new file in the configuration folder in SIEM deployment this file located in "$SPLUNK_HOME/etc/system/local" the source type helps the SIEM server in determine how can the server reacts with this kind of log. Also, we added a special field called *stamp* in each forwarded packet such that Splunk identified the source type of the received log.

In Splunk, it is required to write specific regular expressions to extract certain packet fields from the traffic logs. For IoT botnet detection, we instructed Splunk to extract: Source Mac address (Src_MAC), Destination Mac address (Dst_MAC), Source IP address (Src_IP), Destination IP address (Dst_IP), Source port (Src_port), Destination port (Dst_port). Figure 7 shows an example of the extracted fields from one of the packets. For each of the attack types mentioned above, we set a threshold value for the number of packets originating from IoT devices. Once this number exceeds the threshold value a notification email is sent to the network administrator and a filtering rule is automatically added to the firewall to block attack traffic.
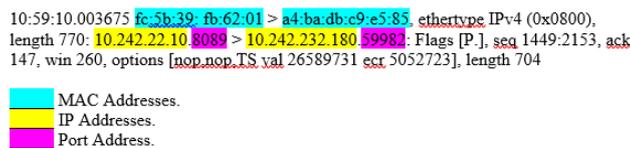

Figure 7. Main packet header fields extracted by Splunk

We tested the prototype by conducting diffrent types of attacks including SYN flooding, DNS flooding, and ICMP flooding. For example, in the case of SYN flooding attack the botmaster instructed IoT device to flood the target machine (IP address: 10.242.232.144) with SYN packets. Figure 8 shows the wireshark traffic capture on the two network interfaces of the Gateway side by side. Splunk alerts the administrator about this attack as shown in Figure 9 and a filtering rule is added automatically to IPTABLES in order to block attack traffic as shown in Figure 10. Dealing with other attacks was done in a similar way.
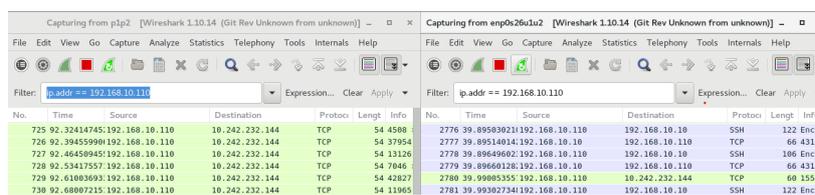

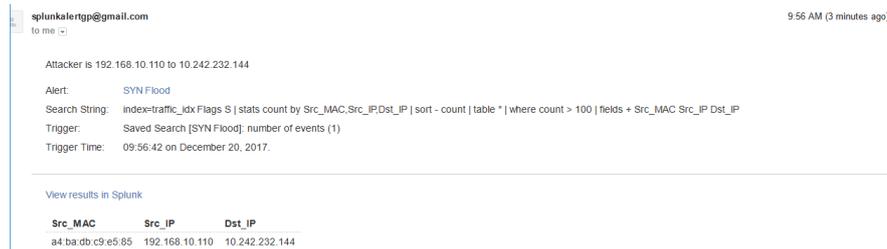Figure 8. A wireshark traffic capture of SYN flooding attack traffic from Internet side and IoT network side

Figure 9. A Splunk generated email alert about ongoing SYN flooding attack

Figure 10. A filtering rule is added automatically to iptables in order to block attack traffic.

6. CONCLUSIONS

With the rapid adoption of IoT devices in our daily life, there is a growing concern from exploiting vulnerabilities of these devices to form IoT botnets and perform different types of attacks. DDoS attacks originating from IoT botnets represent an imminent threat for today's Internet because of the attackers ability to generate high packet volume from millions of compromised IoT devices. In this paper, we proposed a SIEM based system to detect and mitigate this type of attacks. The proposed system detects and blocks DDoS attack traffic from compromised IoT devices by monitoring specific packet types including TCP SYN, ICMP and DNS packets originating from these devices. Also, We discussed a prototype implementation of the proposed system showing how the SIEM based solutions can be configured to accurately identify and block malicious traffic originating from compromised IoT devices. In addition, we discussed recent advances in the field of IoT botnets focusing mainly on main methods to discover IoT devices' vulnerabilities and main approaches to detect IoT botnets.

REFERENCES

[1] H. Lin and N. Bergmann. IoT privacy and security challenges for smart home environments. *Information*, 7(3):44, 2016.
[2] S. Baker, W. Xiang, and I. Atkinson. Internet of things for smart health care: Technologies, challenges, and opportunities. *IEEE Access*, 5:26521–26544, 2017.
[3] H. Boyes, B. Hallaq, J. Cunningham, and T. Watson. The industrial internet of things (iiot): An analysis framework. *Computers in Industry*, 101:1–12, 2018.
[4] S. Cha, S. Baek, S. Kang, and S. Kim. Security evaluation framework for military iot devices. *Security and Communication Networks*, 2018, 2018.
[5] IoT: number of connected devices worldwide 2012-2025 — statista. https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/. (Accessed on 03/09/2019).
[6] K. Angrishi. Turning Internet of things into Internet of vulnerabilities (IovV: IoI botnets. *arXiv preprint arXiv:1702.03681*, 2017.
[7] 2016 dyn cyberattack - wikipedia. https://en.wikipedia.org/wiki/2016_Dyn_cyberattack. (Accessed on 03/09/2019).
[8] Splunk SIEM solution, https://www.splunk.com/. (Accessed on 09/18/2019)
[9] Wei Zhou, Yan Jia, Anni Peng, Yuqing Zhang, and Peng Liu. The effect of iot new features on security and privacy: New threats, existing solutions, and challenges yet to be solved. *IEEE Internet of Things Journal*, 2018.

[10] E. Fernandes, J. Jung and A. Prakash. Security Analysis of Emerging Smart Home Applications. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 636–654. IEEE, 2016.

[11] Y. .M Pa Pa, S. Suzuki, K. Yoshioka, T. Matsumoto, T. Kasama, and C. Rossow. IoTPOT: Analyzing the Rise of IoT Compromises. In *9th {USENIX} Workshop on Offensive Technologies ({WOOT} 15)*, 2015.

[12] E. Ronen, A. Shamir, A. Weingarten, and C. O'Flynn. IoI Goes Nuclear: Creating a Zigbee Chain Reaction. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 195–212. IEEE, 2017.

[13] A. Costin, J. Zaddach, A. Francillon, and D. Balzarotti. A Large-Scale Analysis of the Security of Embedded Firmwares. In *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pages 95–110, 2014.

[14] E. Fernandes, J. Jung, and A. Prakash. Security Analysis of Emerging Smart Home Applications. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 636–654. IEEE, 2016.

[15] H. Kim, T. Kim, and D. Jang. An Intelligent Improvement of Internet-Wide Scan Engine for Fast Discovery of Vulnerable IoT Devices. *Symmetry*, 10(5):151, 2018.

[16] Z. Durumeric, M. Bailey, and J. A. Halderman. An Internet-Wide view of Internet-Wide Scanning. In *Proceedings of the 23rd USENIX Conference on Security Symposium*, SEC'14, pages 65–78, Berkeley, CA, USA, 2014. USENIX Association.

[17] J. Luo, C. Shan, J. Cai, and Y. Liu. IoT Application-Layer Protocol Vulnerability Detection Using Reverse Engineering. *Symmetry*, 10(11):561, 2018.

[18] Y. Jia, Y. Xiao, J. Yu, X. Cheng, Z. Liang, and Z. Wan. A Novel Graph-based Mechanism for Identifying Traffic Vulnerabilities in Smart Home IoT. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pages 1493–1501. IEEE, 2018.

[19] H. Li, Y. He, L. Sun, X. Cheng, and J. Yu. Side-Channel Information Leakage of Encrypted Video Stream in Video Surveillance Systems. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE, 2016.

[20] N. Apthorpe, D. Reisman, and Nick Feamster. A Smart Home is no Castle: Privacy Vulnerabilities of Encrypted IoT Traffic. *arXiv preprint arXiv:1705.06805*, 2017.

[21] D. Summerville, K. M. Zach, and Y. Chen. Ultra-Lightweight Deep packet Anomaly Detection for Internet of Things Devices. In *2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC)*, pages 1–8, Dec 2015.

[22] G. Sagirlar, B. Carminati, and E. Ferrari. Autobotcatcher: Blockchain-based P2P Botnet Detection for the Internet of Things. *CoRR*, abs/1809.10775, 2018.

[23] V. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, oct 2008.

[24] P. Ioulianou, V. Vasilakis, I. Moscholios, and M. Logothetis. A Signature-based Intrusion Detection System for the Internet of Things. 2018.

[25] H. R. Shahriari and E. Khoshhalpour. Botrevealer: Behavioral Detection of Botnets based on Botnet Life-Cycle. *The ISC International Journal of Information Security*, 10(1):55–61, 2018.

[26] L. De Carli, R. Torres, G. Modelo-Howard, A. Tongaonkar, and S. Jha. Botnet Protocol Inference in the Presence of Encrypted Traffic. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, May 2017.

[27] A. O. Prokofiev, Y. S. Smirnova, and V. A. Surov. A Method to Detect Internet of Things Botnets. In *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 105–108, Jan 2018.

[28] H. Sedjelmaci, S. M. Senouci, and M. Al-Bahri. A Lightweight Anomaly Detection Technique for Low-Resource IoT Devices: A Game-Theoretic Methodology. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2016.

[29] H. Bostani and M. Sheikhan. Hybrid of Anomaly-based and Specification-based IDS for Internet of Things Using Unsupervised OPF based on MapReduce Approach. *Computer Communications*, 98:52–71, jan 2017.

[30] D. S. Lavrova, "An approach to developing the SIEM system for the Internet of Things," Automatic Control and Computer Sciences, vol. 50, no. 8, pp. 673–681, 2016.

[31] P. Zegzhda, D. Zegzhda, M. Kalinin, A. Pechenkin, A. Minin, and D. Lavrova, "Safe integration of SIEM systems with Internet of Things: Data aggregation, integrity control, and bioinspired safe routing," in Proceedings of the 9th International Conference on Security of Information and Networks, SIN 2016, pp. 81–87, USA, July 2016.