MedBIoT: Generation of an IoT Botnet Dataset in a Medium-sized IoT Network

Alejandro Guerra-Manzanares[a], Jorge Medina-Galindo, Hayretdin Bahsi[b] and Sven Nõmm[c]
Department of Software Science, Tallinn University of Technology, Tallinn, Estonia

Keywords: Botnet, Internet of Things, Dataset, Intrusion Detection, Anomaly Detection, IoT.

Abstract: The exponential growth of the Internet of Things in conjunction with the traditional lack of security mechanisms and resource constraints associated with these devices have posed new risks and challenges to security in networks. IoT devices are compromised and used as amplification platforms by cyber-attackers, such as DDoS attacks. Machine learning-based intrusion detection systems aim to overcome network security limitations relying heavily on data quantity and quality. In the case of IoT networks these data are scarce and limited to small-sized networks. This research addresses this issue by providing a labelled behavioral IoT data set, which includes normal and actual botnet malicious network traffic, in a medium-sized IoT network infrastructure (83 IoT devices). Three prominent botnet malware are deployed and data from botnet infection, propagation and communication with C&C stages are collected (Mirai, BashLite and Torii). Binary and multi-class machine learning classification models are run on the acquired data demonstrating the suitability and reliability of the generated data set for machine learning-based botnet detection IDS testing, design and deployment. The generated IoT behavioral data set is released publicly available as MedBIoT data set*.

1 INTRODUCTION

The adoption of the Internet on an increasing wider scope, i.e., providing connectivity capabilities to everyday objects, is a reality. In fact, the rise of the Internet of Things (IoT) has just begun and it is expected to have a major increase in the near future. It was estimated that there would be 26.66 billion active IoT devices by 2019, a figure that may be increased up to 75 billion by 2025 (Statista, 2019). 127 new IoT devices are connected to the Internet every second (McKinsey, 2017) in a wide range of applications, from factories and smart cities sensors to healthcare and car products. The market size is calculated to grow over $212 billion by 2019 and reach $1.6 trillion by 2025 (Liu, 2019). However, the adoption of the IoT technology still poses usability concerns even to early adopters and eager customers, related to device security and data privacy issues (Bosche et al., 2018; Sklavos et al., 2017). Thus, despite its huge growth, the Internet of Things market explosion is still being limited by its main barrier: security (Bertino and Islam, 2017; Bosche et al., 2018; Pratt, 2019).

Their ubiquity will pose a major challenge to security as IoT devices have traditionally lacked of proper control measures and proactive security management (e.g., usage of default passwords, no firmware updates, no access control policy), featuring them as high vulnerable and prone to be compromised devices (Bertino and Islam, 2017). These features have been exploited by malicious actors, being able to compromise the defenseless devices by exploiting its vulnerabilities, gaining remote access and using them as magnification platforms for their massive attacks (Kolias et al., 2017). An IoT botnet is just a particular type of botnet in which the compromised devices are IoT devices, thus showing analogous scheme and dynamics to computer botnets. In this regard, when a vulnerable device is compromised it becomes a *bot*, a member of a larger community of compromised devices, called *botnet*, under the control of a malicious actor, the *botmaster*. The botmaster has remote access and control of the bot over the Internet, without the consent and awareness of the actual owner of the compromised device, using a Command&Control (C&C) server (Silva et al., 2013). Botnets have been used to perpetrate a wide range of malicious attacks, from

[a] https://orcid.org/0000-0002-3655-5804
[b] https://orcid.org/0000-0001-8882-4095
[c] https://orcid.org/0000-0001-5571-1692

*Available at https://cs.taltech.ee/research/data/medbiot

Guerra-Manzanares, A., Medina-Galindo, J., Bahsi, H. and Nõmm, S.
MedBIoT: Generation of an IoT Botnet Dataset in a Medium-sized IoT Network.
DOI: 10.5220/0009187802070218
In *Proceedings of the 6th International Conference on Information Systems Security and Privacy (ICISSP 2020)*, pages 207-218
ISBN: 978-989-758-399-5; ISSN: 2184-4356

massive SPAM and phishing campaigns to distributed denial-of-service (DDoS), the most common usage of a botnet. A DDoS attack targets the availability of online resources, such as websites or services. The main goal is to saturate the targeted server or network with more traffic than it can handle (e.g., receiving an overwhelming amount of messages, connection requests or forged packets) thus provoking the service or website to crash and become unavailable to legitimate users requests (Weisman, 2019).

1.1 Data Sets for IoT Anomaly Detection

The phenomenon of botnet detection in computer networks has been widely studied (Garcia et al., 2014; Feily et al., 2009), with many available data sets at hand (Shiravi et al., 2012), while the most recent IoT network botnet phenomenon has not received the required attention yet, showing a remarkable lack of available data sources.

Data sets for building effective IoT anomaly detection methods rely on the acquisition of both legitimate (normal) and malicious (botnet) behavioral data from IoT networks. Anomaly models are built and trained using only legitimate data to establish the so-called normality patterns. The induced models are tested using legitimate and malicious data, where the metrics related to model's detection performance are evaluated. Therefore, proper and complete data are key components for a high-performance effective intrusion detection system (IDS). Table 1 summarizes the available data sets for IoT anomaly-based intrusion detection systems. As can be observed, a small amount of data sets are available for the specific IoT botnets issue. The available data sets are focused on small-sized IoT networks, reflecting the behavior of a small set of IoT devices. Additionally, a specific and small variety of devices are used (mostly security cameras) limiting the scope of the IoT devices analyzed from the broad domain of available IoT devices. None of the available IoT data sets combine real and emulated devices, which limit the scope of their results to either real or emulated devices. In this regard, our generated data set combines real and emulated devices, using different but common types of IoT devices, not investigated by previous data sets (i.e., fans, locks, light bulbs and switches), in a medium-sized network composed of more than 80 devices. Furthermore, our data set focuses on the first stages of a botnet deployment, such as infection and propagation, while the rest of the data sets focus on the last stages of the botnet lifecycle, mainly detection of attacks (Kirubavathi and Anitha, 2014).

As already stated, the Internet of Things is a reality that will become ubiquitous in the following years. This fact combined with the lack of proper security measures and devices inherent vulnerabilities make IoT devices an easy and appealing target for cyber attackers (Bertino and Islam, 2017). Thus, proper data are in need to create machine learning-based effective detection systems that may help to overcome these limitations. In this regard, there is a remarkable lack of available data sets that might help to build effective IDSs in IoT networks. This research aims to fill this significant gap in IoT anomaly-based IDSs by providing a novel IoT data set obtained from medium size IoT network architecture (more than 80 devices), which includes normal and malicious behavior from different devices (real and emulated) and the deployment of prominent IoT botnets (Mirai, BashLite and Torii). The scale extension enables to capture malware spreading patterns that cannot be seen in small-sized networks, thus providing a more realistic environment. Additionally, this data set includes the behavior of Torii botnet malware which has not been addressed in any other data set before. Finally, this data set provides data for the first stages of botnet deployment (i.e., infection, propagation and communication with C&C server stages), thus complementing the available data sets which mainly focus on attack detection, the main outcome and part of the last stages of the botnet lifecycle (Hachem et al., 2011; Kirubavathi and Anitha, 2014).

This paper is structured as follows: Section 2 provides background information and a review of related literature, Section 3 explains the methodology followed to implement the experimental setup while Section 4 offers a detailed overview of the main outcome of this study, a novel IoT data set for botnet detection, and its verification. Finally, Section 5 concludes the study and highlights its main contributions.

2 LITERATURE REVIEW & BACKGROUND INFORMATION

2.1 Botnets & DDoS Attacks

Botnets have been used to perpetrate record-breaking DDoS attacks. In this regard, in 2016, the journalist Brian Krebs was the target of a record-breaking attack (620 Gbps) to its blog KrebsOnSecurity.com, specifically tailored to take the site offline (Krebs, 2016). A month later, the french hosting provider OVH was attacked by the same botnet (probably BashLite), reaching 1 Tbps and involving over 140.000 compromised

Table 1: Data sets for IoT Anomaly-based IDS.

Data set	Botnet	Number of devices	Device type	Real or Emulated	Network Size	Data set features	Date	Reference
N-Baiot	Mirai BashLite	9	Doorbell Webcam Thermostat Baby monitor Security Camera	Real	Small	115 - statistics	2018	(Meidan et al., 2018a) (Meidan et al., 2018b)
IoT host-based datasets for ID research	Hajime Aidra BashLite Mirai Doflo Tsunami Wroba	2	Multimedia Center Security Camera	Emulated	Small	NA - PCAP & Netflow/Host	2018	(Bezerra et al., 2018a) (Bezerra et al., 2018b)
IoT Network Intrusion Dataset	Mirai	2	Speaker Wi-Fi Camera	Real	Small	NA - PCAP	2019	(Kang et al., 2019)
Bot-IoT	No actual malware - simulated	5	Refrigerator Smart Garage door Weather Monitoring Smart Lights Smart thermostat	Emulated	Small	31+14 - flow	2019	(Moustafa, 2019) (Koroniotis et al., 2019)

cameras/dvr (Pritchard, 2018). The same year, Dyn, a domain name system provider of major websites and services such as CNN, Netflix, Paypal, Visa or Amazon was attacked by the Mirai botnet, using around 100.000 IoT devices and reaching up to 1.2 Tbps, causing the servers to be inoperative and the websites unreachable by the legitimate users for several hours (Weisman, 2019; Hilton, 2016). It is estimated that Dyn lost around 8% of its customers (i.e., 14000 domains) as a consequence of the attack and the lost of trust (Weagle, 2017). This was just the onset for the IoT botnet-based attacks. Since then, the attacks have not stopped, evolving in sophistication and capabilities as the source code of the malware behind the botnets became available to the public (Asokan, 2019). A recent report by F-secure states that cyber-attacks on IoT devices rouse 300% in 2019, reaching the 3 billion attacks, an unprecedented figure (Doffman, 2019). The threat is still alive and growing, caused mainly by the combination of the increase of the number of IoT devices deployed worldwide and the intrinsic vulnerabilities carried by such devices, which can also contain valuable data related to medical or control issues. Nevertheless, one of the major risks is the usage of the IoT endpoints (e.g., a printer or a fridge) as an easy-to-reach and vulnerable entry points to wider and secured networks (Doffman, 2019).

As a result, cyber security for IoT, in the form of early detection of threats, becomes a key issue to detect and mitigate such attacks. In this regard, intrusion detection systems are widely used network security components which aim to detect security threats where preventive security measures are not feasible to implement (Benkhelifa et al., 2018; Sun et al., 2007).

2.2 Intrusion Detection Systems

An intrusion could be defined as a set of activities or actions that compromise one or more components of the IT security model known as CIA triad (i.e., short for Confidentiality, Integrity and Availability) of a specific entity or system. These systems are not restricted to computers, network equipment, firewall, routers or networks but to any information technology system which is under the monitoring scope of an intrusion detection system (IDS) (Sun et al., 2007). Based on that, an intrusion detection system is a security tool that aims to detect and identify the unauthorized individuals willing to break into and misuse a system and also those authorized and legitimate users that abuse of their privileges within the system (Sun et al., 2007). There are four common approaches used for intrusion detection: misuse, anomaly, specification and hybrid (Benkhelifa et al., 2018; Sun et al., 2007; Butun et al., 2013; Zarpelão et al., 2017). They are briefly explained as follows:

- Misuse or signature-based detection systems use known fingerprints or signatures from attacks stored in a database. If an IDS finds a match between the current activities and a known signature it raises the alarm about the detected suspicious behavior. This systems are easily bypassed by not-known or novel attacks, when a signature is not yet available.
- Anomaly-based detection systems are based on the creation of a typical or normal activity profile. Current activities are compared against this normal behavior. If the IDS finds a significant deviation or discrepancies from the normality model it raises the alarm about the suspicious behavior.

These systems success on the detection of novel attacks but they are prone to false positives (i.e., legitimate behavior is detected as malicious behavior) as the normal behavior might not be easy to model, so that being very sensitive to the correctness of the normality model created.

- Specification-based detection systems combine features of misuse and anomaly approaches. They apply anomaly-based principle on set of human generated specifications or constraints about the normal or legitimate behavior. These systems aim to detect novel attacks based on anomalous behavior while reducing the amount of false positives.
- Hybrid detection systems involve the combination of any of the previous approaches, aiming to overcome the weaknesses of one approach using the strengths of another.

One of the most effective and widely used detection methods is the anomaly-based approach, which enables to detect novel attacks but with the inevitable trade-off of being sensitive to the correctness of the generated normality model. In this regard, statistical methods and machine learning algorithms are generally used to generate the normal behavior profile (Zarpelão et al., 2017). Therefore, valid behavioral models should be used in order to obtain the maximum benefit of this approach, depending in a direct manner on the available training data (Bolzoni, 2009). In IoT networks, where a wide variety of devices may coexist in the same network, it is likely to have different normality profiles which emphasizes the need of accurate IoT behavioral data that enable the implementation of effective anomaly-based IDS. Thus, the need of proper data encompassing such differences are highly in demand. However, there is a remarkable lack of available data sets that consider the different network behaviors, devices and architectures that can be found in IoT networks and its major threats. As a result, proper IoT behavioral data are key to train the IDS model for effective intrusion detection in IoT networks.

2.3 Machine Learning-based IDS

Machine learning has shown promising results regarding computer botnet traffic detection (Livadas et al., 2006) and more lately, in the specific IoT botnet detection issue (Zarpelão et al., 2017). As a result of the remarkable increase in IoT related security incidents, researchers have reoriented their focus to deal with the investigation of feasible and effective IoT botnet detection methods involving anomaly-based machine learning approaches. These approaches aim to overcome the intrinsic hardware and software limitations and capabilities of these devices (Zarpelão et al., 2017). In this regard, in Meidan et al. (2018b), Deep Autoencoders, Local Outlier Factor, One-Class Support Vector Machines and Isolation Forest algorithms models built and evaluated using the N-baiot dataset. The results show that all algorithms, except Isolation Forest, effectively detected all Mirai and BashLite simulated attacks. Their proposed method, based on Deep Autoencoders, showed the lowest *false alarms* ratio and required less time to detect the attacks than the other approaches. Prokofiev et al. (2018) used Logistic Regression algorithm to estimate the probability that a device was part of an IoT botnet, focusing on the connection initiation at the propagation stage. Lin et al. (2014) proposed an IoT botnet detection method which combines Support Vector Machines and Artificial Fish Swarm algorithms. McDermott et al. (2018) provided a new application for a text recognition deep learning algorithm (Bidirectional Long Short Term Memory based Recurrent Neural Network), with remarkable success on Mirai botnet attack detection. Doshi et al. (2018), used different network features to train and evaluate the accuracy of *k*-Nearest Neighbors, Support Vector Machines, Decision Tree, Random Forest and Artificial Neural Networks algorithms on the detection Mirai DDoS attacks. A novel IoT malware detection approach using network traffic is proposed in Shire et al. (2019) where Convolutional Neural Networks and binary visualisation technique were used to provide a fast detection method for zero-day malware.

As can be observed, the application of anomaly detection requires the acquisition of malicious traffic which is tested against normal or legitimate traffic in order to evaluate the goodness of the proposed detection model. For this purpose, the data sets should provide both kinds of network traffic in order to assess the effective detection of threats. In this paper we provide demonstrability of the generated data set on classification issues (i.e., supervised learning), for the easiness of interpretation of the results and comparison, but this data set may also be used to build effective anomaly detection models, considered traditionally unsupervised learning.

3 METHODOLOGY

The main outcome of this research is the generation of a labelled behavioral IoT data set, which includes normal and actual botnet malicious network traffic, in a medium-sized IoT infrastructure (composed of more than 80 devices). The focus was on the acquisition

of network data from all the endpoints and servers during the initial propagation of Mirai, BashLite and Torii botnets.

3.1 IoT Network Topology

The network topology created for the purpose of this study is provided in Figure 1. It is composed by 3 connected networks: internet network, monitoring network and IoT LAN network. Their functions and components are described as follows:

- Internet Network: this network is directly connected to the internet in order to provide internet connectivity for the initial configuration of different devices. To restrict the connectivity between networks, a different subnetwork mask is established.
- Monitoring Network: this network provides storage and processing capabilities for the data received from the switch. It is composed by a capture server and a security information and event management (SIEM) server. The capture server is responsible for the collection and storage of the acquired network packages within the whole infrastructure. *Tcpdump* is used to monitor and log the network traffic and store the data in *pcap* file format which is later used as an input by the SIEM server. The SIEM server is a *Splunk* software instance which is responsible for data indexing, filtering, analysis and data set generation (i.e., data processing and labelling).
- IoT LAN Network: this local area network (LAN) allows to spread the malware in a contained manner. This network is composed of physical and virtual IoT devices that generate the behavioral traffic collected by the monitoring network, either benign or malware generated traffic. Virtual devices are deployed using containerization software (i.e., *Docker*). The composition and capabilities of this network devices are explained as follows:
  - Router: this device is responsible for the generation of an isolated network segment allowing only communication between internal devices within this network (i.e., using firewall rules). The router provides IP addresses to this internal devices using Dynamic Host Configuration Protocol (DHCP).
  - Switch: this device is responsible for the acquisition and transfer of the network packages using the *port mirroring* technique. *Port mirroring* is used to clone and transfer network packages that flow through one port to another port, in real time, without affecting the network performance. In this scenario, all devices generated data are captured and transferred to the monitoring network.
  - IoT Management System: this device allows the management of all the IoT devices in a centralized manner. It is deployed using *Hassio* software running on a *Raspberry Pi*, which allows to simulate the same network behavior of real implementations. In this network, 4 different IoT devices were emulated: fan, lock, light bulb and switch. Each device allows the remote control of different features. For instance, the fan allows the selection of speed, oscillation state, current fan state and turning on/off capabilities.
  - Virtual IoT Devices: this device allows the virtualization of IoT devices using *Docker* containers. It is deployed using a *Raspberry Pi* which allows to emulate the behavior of an IoT device.
  - Wireless Access Point: this device allows network connection to the non-ethernet compatible devices. It is configured to allow the router the capability of assigning IP addresses (via DHCP), thus avoiding the possibility of IP address duplicates.
  - BashLite C&C Server: this server is the command and control unit of the BashLite botnet. FTP and web services are installed to allow the spreading of the malware. The server is also used to compile the malware binaries used to propagate the infection.
  - Mirai C&C Server: this server is the command and control device of the Mirai botnet. FTP and web services are installed to allow the malware propagation. The server is also used to compile the malware binaries used to spread the infection.
  - DNS Server Sinkhole: this server provides the domain name resolution for the Mirai botnet. It is also used as a sinkhole for the domains that Torii malware requests connection. The sinkhole avoids the actual connection between Torii and the domain of its C&C server, providing effective malware contention.
  - Physical Devices: this devices compose the collection of real IoT devices of this network. It is composed by 3 different devices: Sonoff tasmota smart switch, TpLink smart switch and TpLink smart bulb. All of them allow external device management and provide different features. For instance, the light bulb allows to

control light intensity, status and turn on/off capabilities.

In order to create a medium-sized network, virtual devices are created and physical devices deployed, summing up a total amount of 83 devices. Table 2 shows the composition of the IoT LAN network. As can be observed, 80 devices are emulated and 3 are actual physical devices. The virtual devices have ARM architecture as it is inherited from the *Raspberry Pi* while the physical devices have MIPS architecture. This fact conditions the malware binary used to infect the device, being architecture-dependant, and enriches the spectrum of the data, considering a wider variety of IoT devices. The *features* column provides outlines the actions that the deployed IoT devices are capable to perform.

Table 2: IoT network device composition.

Device	Type	Features	Architecture	Number of devices
Switch	Physical	Turn On Turn Off	MIPS	2
Light bulb	Physical	Turn On Turn Off Intensity	MIPS	1
Lock	Virtual	Lock Unlock	ARM	20
Fan	Virtual	Turn On Turn Off Speed Oscillation	ARM	20
Switch	Virtual	Turn On Turn Off	ARM	20
Light bulb	Virtual	Turn On Turn Off Intensity	ARM	20

3.2 IoT Behavior

The simulation of devices' behavior can be performed in several ways, ranging from the imitation of the behavior by manual usage of the devices to the automation of the execution of specific functions/tasks using *scripts*. The quality and consistency of the simulated behavior is key to create a high quality data set that provide realistic data input for effective IDS solutions. In such cases, the acquisition of real and relevant data regarding the normal usage patterns provide a realistic baseline for the simulation of the behavior. For instance, in a normal living room, the research showed that a light bulb had a mean usage of 1.7h per day while this value achieved 2.3h in the case of a light bulb in the kitchen (Gifford et al., 2012). This information provided a baseline for the simulation of benign behavior in our experimental setup. In the case of malware behavior it is simulated by the execution of the different modules within the botnet, providing a real output of the botnet behavior.

3.2.1 Legitimate Behavior

An automated execution approach is utilised for the simulation of benign behavior. This approach takes into account the architecture of the device, as stated in Table 2, performed using a python script and MQTT (MQ Telemetry Transport) protocol, which is a communication protocol used to control IoT devices. The IoT management system allows to automate this control and perform scheduled tasks on connected IoT devices. A *script* with trigger actions is configured and deployed. In this scenario, the following legitimate behavior is simulated using the following triggers:

- All devices are turned on at 8.00 AM
- Each time a device state changes, the management system starts a countdown until the next state change.
- The countdown value is randomized.
- The maximum limit of changes is established in 20 and a maximum of 3h on *ON* state is set.
- All devices are turned off at 07.00 PM
- In order to simulate a working environment, execution of the triggers is limited from Monday to Friday.

By the usage of the previous triggers, network packages are generated along the network, captured and stored. The captured network packages provide the following communication information: time, protocol used, TCP stream, TCP stream size, source IP, destination IP, MAC addresses, TCP raw message and response code.

3.2.2 Malicious Behavior

The malicious behavior is generated by the deployment of three prominent botnet malware within the controlled environment: Mirai (Antonakakis et al., 2017), BashLite (Marzano et al., 2018) and Torii (Kroustek et al., 2018). Mirai and BashLite botnets have been widely studied and malware source code is available on the Internet. Thus its deployment is fully controlled in the lab environment using a C&C server for each botnet and the source code is modified to connect with this specific C&C server. Torii source code is not yet available on the Internet, thus the samples used for its deployment within the controlled environment were obtained from Hybrid Analysis archive (Crowdstrike, 2019). In order to avoid Torii malware to connect with its actual C&C server, special contention measures are in place. Mirai, BashLite and Torii botnet propagation is performed and controlled

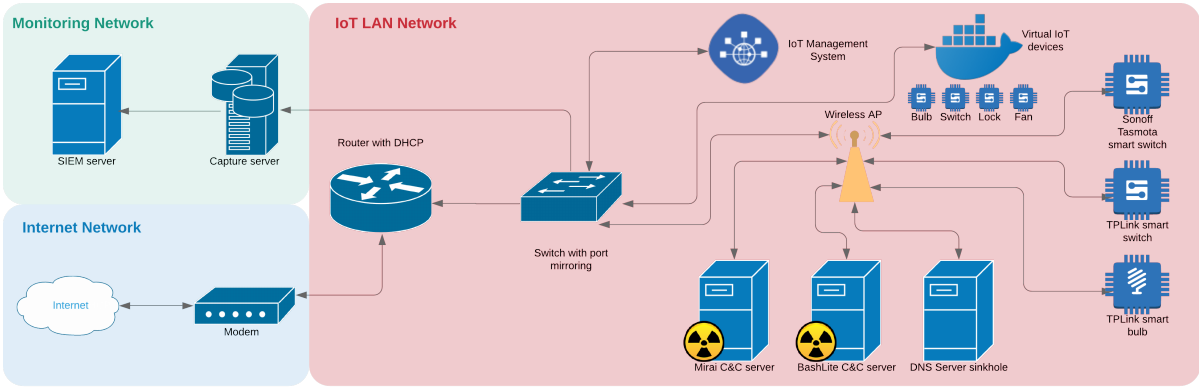

Figure 1: Medium-sized IoT network topology.

within the environment using different strategies, explained in the following paragraphs.

- **Botnet Propagation Techniques**
  - Mirai and *Yakuza* version of BashLite are configured and executed within the controlled environment, modifying the malware source codes in order to link the infection binaries with the corresponding in-lab C&C servers. Once the botnet is properly set, droppers are the medium used by these botnets to download and install the appropiate malware file according to the victim's architecture which once executed will run the bot daemon, compromising the device successfully.
  - Torii malware has not been profoundly studied yet, so the deployment of this malware in the lab environment carries further risks. In order to contain and eliminate the risks of improper use of the devices by Torii's actual botmaster, a sinkhole in the DNS server and firewall rules are used. Torii connection attempts with its C&C server are permanently denied and redirected. As a result of the lack of proper knowledge of Torii malware spreading methods and source codes, the binaries are deployed manually within the lab environment. The obtained sample is specifically tailored to target ARM devices. The malware is executed running the executable with root privileges in the target devices, allowing to spread the malware through the IoT devices.
- **Botnet Contention Methods**
  - One of the major risks within the lab is the abuse of the devices by real attackers. In this regard, Torii poses a major challenge. Contrarily to Mirai and BashLite, Torii has not been deeply analyzed and poses a risk within the lab environment that has to be addressed. Unsuccessful botnet contention may lead to unauthorized usage of the IoT devices by real attackers to perpetrate attacks or collect rellevant data. Two major risks are found within this experimental setup which are addressed and outlined as follows:
    1. Possibility of existence of hidden code in Mirai's source code to connect to the real C&C server
    2. Torii's unknown spread techniques and functionalities

    Even though Mirai spreading techniques are well-known, additional security measures are taken to ensure effective contention of the malware. To address this issues, a sinkhole and firewall rules are in place to deny possible connection attemps to the real C&C servers. The DNS sinkhole redirects the connection attempts by resolving the name resolution request with a controlled IP address. Firewall rules are set in the router to block/control the traffic based on known network masks.

Botnet malware are deployed at different times within 6 days (i.e., each let run free for 2 consecutive days) aiming to obtain relevant botnet information and eliminate undesired overlapping of information. Furthermore, Mirai malware is capable of detect malware running on a specific device and remove it in order to take the single control of it. A limited number of devices are infected in each botnet deployment. In the case of BashLite malware, 40 devices were infected, chosen in a pseudo-randomized way by limiting the scope of devices scanned and infected. Mirai botnet malware infected 25 devices, limited by the change of configuration to restrict the internal scanner to spread within the lab IP ranges. Torii botnet malware was manually deployed in 12 devices, all under the controlled scope of the DNS sinkhole.

3.3 IoT Behavior Verification

In order to verify the suitability of the IoT behavioral data set generated within the experimental setup for detection purposes, the generated data are further processed and used to build and test machine learning-based classification models. Machine learning classification models aim to correctly predict the label or category of an unknown data point based on features (also called predictors) found on the training data provided during the model training/building phase (i.e., supervised learning). Binary classification is used when the data points are split into to two mutually exclusive categories (e.g., benign and malware) while multi-class classification deals when more than two categories are present within the data (e.g., benign, Mirai, BashLite and Torii). In order to validate the outcome of the experimental setup, both approaches are used, inducing binary and multi-class machine learning classification models, which are validated using k-fold cross validation.

From the source *pcap* files captured within the lab, features are extracted and used as predictors/input for the machine learning models. The features used in this lab are computed as in Mirsky et al. (2018). A total of 100 network traffic statistical features are calculated, within different time windows. Table 3 provides a brief description of the generated features. As can be observed, statistical features are calculated in relation to 4 major categories for each of the 5 time windows (i.e, 100ms, 500ms, 1.5s, 10s and 1min).

Table 3: Feature Categories.

Categories	Features
Host-MAC&IP	Packet count, mean and variance
Channel	Packet count, mean, variance, magnitude, radius, covariance, and correlation
Network Jitter	Packet count, mean and variance of packet jitter in channel
Socket	Packet count, mean, variance, magnitude, radius, covariance and correlation

After the features are extracted, a random sample of data points are selected for each class and used to train/test machine learning models using 10-fold cross validation. Four traditional machine learning algorithms are used to induce and test classifier models. The main objective for these tests is to demonstrate the suitability of the present data set for machine learning-based anomaly and classification detection models. In this regard, there is no model hyper-parameter optimization performed on the induced models. Default *scikit_learn* library (version 0.20) configurations are used, leaving room for improvement on the classifiers performance. In this relation, k-Nearest Neighbors (k-NN), Support Vector Machines (SVM), Decision Tree (DT) and Random Forest (RF) algorithms are implemented. For each of the models, four performance metrics are reported: accuracy, precision, recall and F_1 score. They are briefly described as follows:

- Accuracy: ratio of the correctly classified test instances among all test instances.
- Precision: fraction of positive instances correctly classified among all the positive classified instances.
- Recall: fraction of positive instances correctly classified among all the actual positive instances.
- F_1 score: harmonic mean of precision and recall metrics.

All the performance metrics are bounded on the interval [0, 1]. In general, a value close to 1 may be deemed as a positive or good result for the given task while a value close to 0 as a bad performance. In this regard, for classification tasks, the higher the value the better the classifier performance on label detection and discrimination, thus inferring that the data and the classifier are suitable for that purpose. In our specific case, if the classifiers show a performance close to 1 in all metrics it may be inferred that the data is suitable for machine learning-based IoT botnet detection and that the data labels (e.g., legitimate and malware) can be discriminated effectively.

4 RESULTS

4.1 IoT Behavioral Data Set

The network packets collected in the IoT LAN network are redirected to the monitoring network using the port mirroring technique, where the SIEM software (i.e., *Splunk*) was used to process and label the data, thus allowing to create the final data set. This final data set is generated in two versions: structured (features are computed and extracted from the raw data) and non-structured format (raw *pcap* files). The total number of packets captured during the experimental setup are provided in Table 4.

Table 4: Network data captured.

Number of packets	Traffic type	Number of devices
4,143,276	BashLite	40
842,674	Mirai	25
319,139	Torii	12
12,540,478	Benign	83

As can be observed, a total amount of 17,845,567 network packets were captured within the experimental setup. Around 30% of this traffic was deemed and labelled as malicious while 70% corresponds to legitimate network traffic. Using *Splunk* it is possible to analyze and provide further details about the type of communication. Regarding the legitimate network traffic, 32% of the packages are found to be related to system updates, 53% to device communication (MQTT protocol) and 15% to other network data (e.g., TLS errors, pings, etc). Mirai and BashLite source codes are configured to convey different kind of communications on different ports, with the purpose of facilitating the posterior analysis of the data. In this relation, malicious traffic analysis shows that 68% of the data captured is related to the malware propagation activity while 32% to the communication between the C&C servers and bots. In the case of Torii, malicious traffic only includes data regarding the initial infection of the device as the containment measures did not allow the real C&C to reach the device and trigger posterior botnet events such as propagation. The generated data set is made publicly available in the following url: https://cs.taltech.ee/research/data/medbiot

4.2 IoT Behavior Verification

4.2.1 Binary Classification

Binary or two-class classification models are induced and 10-fold cross validated for four widely used machine learning classification models. In this case, the data is divided in two classes or labels: legitimate and malware (mixed data from the three malware subclasses). More specifically, the data set used is created by random selection of 3000 data points from the legitimate traffic, thus conforming the legitimate class data. The malware class is composed of 1000 random selected data points for each one of the malware botnets deployed within the lab, summing up to 3000 data points for this class. As a result, a balanced data set is created and used to perform the binary classification task. Support Vector Machines algorithm showed a poor performance in all assessed metrics, thus is not reported in the results, which are provided in Table 5.

Table 5: Binary classification results.

Model	Accuracy	Precision	Recall	F_1 score
k-NN	0.9025	0.9082	0.9025	0.9001
DT	0.9315	0.9448	0.9315	0.9293
RF	0.9532	0.9580	0.9532	0.9481

As can be observed, Random Forest algorithm is able to discriminate over 95% of the data points, thus detecting effectively the vast majority of the malware traffic. Decision Tree and *k*-NN show slightly less discriminatory performance, but over 90% in all performance metrics in both cases. The malware traffic, which is composed of a mixture of 3 different botnet malware, is effectively discriminated from legitimate traffic, as can be confirmed by the normalized confusion matrix provided in Table 6, extracted from a Random Forest model. As already stated, SVM showed bad performance, and its results are not reported. Nevertheless, this fact may suggest that the data is not linearly separable, thus being SVM a not suitable classifier model for this task unlike the other algorithms used. These results emphasize the effective capabilities of machine learning approaches to detect botnet malware traffic, even in the first stages of its deployment (i.e., infection, propagation and communication with the C&C server stages) and disregarding the botnet malware employed. Furthermore, the data set created within this lab demonstrates its suitability to be used as a medium-sized realistic IoT data set for IoT botnet detection scenarios and IDS testing.

Table 6: Confusion matrix of RF binary classification.

		Predicted	
		Malware	Legitimate
Actual	Malware	291	9
	Legitimate	7	293

4.2.2 Multi-class Classification

In this setting, multi-class classification models are induced and 10-fold cross validated for the same algorithms employed in the binary approach. In this case, the data was divided in four classes or labels: legitimate, Mirai, BashLite and Torii. The data set used is created by random selection of 2000 data points of each of the possible classes, summing up to 8000 data points, evenly distributed in 4 labels. The main aim of this configuration is not only to test the legitimate/malware discrimination, as in the binary approach, but also the discrimination of the specific malware source. As in the previous setting, Support Vector Machines algorithm showed a poor performance in all metrics, thus its performance is not reported. Table 7 provides the results for the multi-class classification task.

As can be seen, Random Forest model outperforms Decision Tree and *k*-NN algorithms in the multi-class classification task, in a similar fashion as in the binary models. More specifically, RF algorithm

Table 7: Multi-class classification results.

Model	Accuracy	Precision	Recall	F_1 score
k-NN	0.8706	0.8849	0.8706	0.8505
DT	0.9516	0.9584	0.9516	0.9499
RF	0.9766	0.9824	0.9766	0.9657

is able to discriminate more accurately the labels in the multi-class scenario than in the binary setting, being able to discriminate accurately over 97% of the data points. As shown in Table 8, extracted from the Random Forest model, the classification model is very accurate in all cases, not showing any significant bias towards any of the possible labels. The results obtained suggest that the source of network traffic can be effectively discriminated in earlier stages of botnet infection. They also demonstrate that the learning capabilities of machine learning-based detection can be accurate not only in the binary setting but also in the specific discrimination of different sources of malicious traffic in medium-sized IoT networks.

Table 8: Confusion matrix of RF multi-class classification.

		Predicted			
		Mirai	BashLite	Torii	Legitimate
Actual	Mirai	197	0	0	3
	BashLite	2	196	0	2
	Torii	0	0	198	2
	Legitimate	2	0	0	198

5 CONCLUSIONS

The exponential growth of the Internet of Things is a fact and these devices will become ubiquitous in the near future. The increasing connectivity capabilities of these devices in conjunction with their traditional lack of security features make them an appealing target for cyber-attackers. Malicious actors compromise the vulnerable IoT devices and use them as an amplification platform of their attacks, becoming part of the so-called *botnet*. Botnets have been extensively used to deliver massive spam campaigns and perpetrate record-breaking DDoS attacks that may lead to nefarious consequences. Therefore, there is an increasing need to overcome the lack of security of these devices. The proposed solutions are mainly coming from machine learning-based approaches.

The performance of machine learning algorithms heavily rely on data quality and quantity. In this relation, there is a remarkable lack of data sources in the specific IoT networks scenario. The experimental setup of this research aims to fulfill this gap by providing a novel data set with network data collected from a medium-sized IoT network architecture, which is composed of legitimate and botnet malware traffic. Three IoT botnet malware are deployed in real and emulated IoT devices and data are acquired from the first stages of botnet deployment, such as infection, propagation and communication with C&C server. These data complements the already existing data sets which mainly focus on detection of botnet attacks, part of the last stages of a botnet deployment. In this sense, by focusing on early stages of botnet deployment, the proposed data set provides the opportunity to perform early detection of the threat, previous to the perpetration of an attack, being able to prevent such attacks and botnet growth. Three prominent botnet malware are deployed in this research, one of them is a complete novelty (i.e., Torii), not being deployed before in any other available data set. The other two are well-known IoT botnet malware whose source code is publicly available and have been used in other data sets (i.e., Mirai and BashLite). The currently available data sets, summarized in Table 1, focus on small-sized networks (usually less than 10 devices), using either emulated or real devices, thus providing limited interactions between devices inside the network. The generated data set addresses these limitations by combining emulated and real devices to create a medium-sized network (i.e., 83 devices). A larger network size may provide different insights and interactions than smaller IoT networks. Finally, machine learning models are built and validated using this data to demonstrate the suitability of this data set as a reliable data source for botnet detection in general and IDS testing and deployment in particular. The data set generated within the experimental setup is made publicly available, aiming to overcome the scarcity of relevant data sources in IoT network security and limitations of the existing data sets.

REFERENCES

Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J. A., Invernizzi, L., Kallitsis, M., et al. (2017). Understanding the mirai botnet. In *26th {USENIX} Security Symposium ({USENIX} Security 17)*, pages 1093–1110.

Asokan, A. (2019). Massive botnet attack used more than 400,000 iot devices. Retrieved from: https://www.bankinfosecurity.com/massive-botnet-attack-used-more-than-400000-iot-devices-a-12841.

Benkhelifa, E., Welsh, T., and Hamouda, W. (2018). A critical review of practices and challenges in intrusion detection systems for iot: Toward universal and resilient systems. *IEEE Communications Surveys & Tutorials*, 20(4):3496–3509.

Bertino, E. and Islam, N. (2017). Botnets and internet of things security. *Computer*, (2):76–79.

Bezerra, V. H., da Costa, V. G. T., Martins, R. A., Junior, S. B., Miani, R. S., and Zarpelao, B. B. (2018a). Data set. http://www.uel.br/grupo-pesquisa/secmq/dataset-iot-security.html.

Bezerra, V. H., da Costa, V. G. T., Martins, R. A., Junior, S. B., Miani, R. S., and Zarpelao, B. B. (2018b). Providing iot host-based datasets for intrusion detection research. In *Anais do XVIII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, pages 15–28. SBC.

Bolzoni, D. (2009). *Revisiting Anomaly-based Network Intrusion Detection Systems.* University of Twente, Enschede, Netherlands.

Bosche, A., Crawford, D., Jackson, D., Schallehn, M., and Schorling, C. (2018). Unlocking opportunities in the internet of things. Retrieved from: https://www.bain.com/contentassets/5aa3a678438846289af59f62e62a3456/bain_brief_unlocking_opportunities_in_the_internet_of_things.pdf.

Butun, I., Morgera, S. D., and Sankar, R. (2013). A survey of intrusion detection systems in wireless sensor networks. *IEEE communications surveys & tutorials*, 16(1):266–282.

Crowdstrike (2019). Hybrid analysis. Retrieved from: https://www.hybrid-analysis.com/.

Doffman, Z. (2019). Cyberattacks on iot devices surge 300% in 2019, 'measured in billions', report claims. Retrieved from: https://www.forbes.com/sites/zakdoffman/2019/09/14/dangerous-cyberattacks-on-iot-devices-up-300-in-2019-now-rampant-report-claims/#574229995892.

Doshi, R., Apthorpe, N., and Feamster, N. (2018). Machine learning ddos detection for consumer internet of things devices. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 29–35. IEEE.

Feily, M., Shahrestani, A., and Ramadass, S. (2009). A survey of botnet and botnet detection. In *2009 Third International Conference on Emerging Security Information, Systems and Technologies*, pages 268–273. IEEE.

Garcia, S., Grill, M., Stiborek, J., and Zunino, A. (2014). An empirical comparison of botnet detection methods. *computers & security*, 45:100–123.

Gifford, W. R., Goldberg, M. L., Tanimoto, P. M., Celnicker, D. R., and Poplawski, M. E. (2012). Residential lighting end-use consumption study: Estimation framework and initial estimates. Retrieved from: https://www1.eere.energy.gov/buildings/publications/pdfs/ssl/2012_residential-lighting-study.pdf.

Hachem, N., Mustapha, Y. B., Granadillo, G. G., and Debar, H. (2011). Botnets: lifecycle and taxonomy. In *2011 Conference on Network and Information Systems Security*, pages 1–8. IEEE.

Hilton, S. (2016). Dyn analysis summary of friday october 21 attack. Retrieved from: https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/.

Kang, H., Ahn, D. H., Lee, G. M., Yoo, J. D., Park, K. H., and Kim, H. K. (2019). Iot network intrusion dataset. http://dx.doi.org/10.21227/q70p-q449.

Kirubavathi, G. and Anitha, R. (2014). Botnets: A study and analysis. In *Computational Intelligence, Cyber Security and Computational Models*, pages 203–214. Springer.

Kolias, C., Kambourakis, G., Stavrou, A., and Voas, J. (2017). Ddos in the iot: Mirai and other botnets. *Computer*, 50(7):80–84.

Koroniotis, N., Moustafa, N., Sitnikova, E., and Turnbull, B. (2019). Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset. *Future Generation Computer Systems*, 100:779–796.

Krebs, B. (2016). Krebsonsecurity hit with record ddos. Retrieved from: https://krebsonsecurity.com/2016/09/krebsonsecurity-hit-with-record-ddos/.

Kroustek, J., Iliushin, V., Shirokova, A., Neduchal, J., and Hron, M. (2018). Torii botnet - not another mirai variant. Retrieved from: https://blog.avast.com/new-torii-botnet-threat-research.

Lin, K.-C., Chen, S.-Y., and Hung, J. C. (2014). Botnet detection using support vector machines with artificial fish swarm algorithm. *Journal of Applied Mathematics*, 2014.

Liu, S. (2019). Global iot market size 2017-2025. Retrieved from: https://www.statista.com/statistics/976313/global-iot-market-size/.

Livadas, C., Walsh, R., Lapsley, D. E., and Strayer, W. T. (2006). Using machine learning techniques to identify botnet traffic. In *LCN*, pages 967–974. Citeseer.

Marzano, A., Alexander, D., Fonseca, O., Fazzion, E., Hoepers, C., Steding-Jessen, K., Chaves, M. H., Cunha, Í., Guedes, D., and Meira, W. (2018). The evolution of bashlite and mirai iot botnets. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 00813–00818. IEEE.

McDermott, C. D., Majdani, F., and Petrovski, A. V. (2018). Botnet detection in the internet of things using deep learning approaches. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

McKinsey (2017). What's new with the internet of things? Retrieved from: https://www.mckinsey.com/industries/semiconductors/our-insights/whats-new-with-the-internet-of-things.

Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Breitenbacher, D., Shabtai, A., and Elovici, Y. (2018a). detection_of_iot_botnet_attacks_n_baiot data set. http://archive.ics.uci.edu/ml/datasets/detection_of_IoT_botnet_attacks_N_BaIoT.

Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Breitenbacher, D., Shabtai, A., and Elovici, Y. (2018b). N-baiot—network-based detection of iot botnet attacks using deep autoencoders. *IEEE Pervasive Computing*, 17(3):12–22.

Mirsky, Y., Doitshman, T., Elovici, Y., and Shabtai, A. (2018). Kitsune: an ensemble of autoencoders for online network intrusion detection. *arXiv preprint arXiv:1802.09089*.

Moustafa, N. (2019). The bot-iot dataset. http://dx.doi.org/10.21227/r7v2-x988.

Pratt, M. K. (2019). Top challenges of iot adoption in the enterprise. Retrieved from: https://internetofthingsagenda.techtarget.com/feature/Top-challenges-of-IoT-adoption-in-the-enterprise.

Pritchard, M. (2018). Ddos attack timeline: Time to take ddos seriously. Retrieved from: https://activereach.net/newsroom/blog/time-to-take-ddos-seriously-a-recent-timeline-of-events/.

Prokofiev, A. O., Smirnova, Y. S., and Surov, V. A. (2018). 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. IEEE.

Shiravi, A., Shiravi, H., Tavallaee, M., and Ghorbani, A. A. (2012). Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *computers & security*, 31(3):357–374.

Shire, R., Shiaeles, S., Bendiab, K., Ghita, B., and Kolokotronis, N. (2019). Malware squid: A novel iot malware traffic analysis framework using convolutional neural network and binary visualisation. In *Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, pages 65–76. Springer.

Silva, S. S., Silva, R. M., Pinto, R. C., and Salles, R. M. (2013). Botnets: A survey. *Computer Networks*, 57(2):378–403.

Sklavos, N., Zaharakis, I. D., Kameas, A., and Kalapodi, A. (2017). Security & trusted devices in the context of internet of things (iot). In *2017 Euromicro Conference on Digital System Design (DSD)*, pages 502–509. IEEE.

Statista (2019). Internet of things - number of connected devices worldwide 2015-2025. Retrieved from: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/.

Sun, B., Osborne, L., Xiao, Y., and Guizani, S. (2007). Intrusion detection techniques in mobile ad hoc and wireless sensor networks. *IEEE Wireless Communications*, 14(5):56–63.

Weagle, S. (2017). Financial impact of mirai ddos attack on dyn revealed in new data. Retrieved from: https://www.corero.com/blog/797-financial-impact-of-mirai-ddos-attack-on-dyn-revealed-in-new-data.html.

Weisman, S. (2019). Emerging threats - what is a distributed denial of service attack (ddos) and what can you do about them? Retrieved from: https://us.norton.com/internetsecurity-emerging-threats-what-is-a-ddos-attack-30sectech-by-norton.html.

Zarpelão, B. B., Miani, R. S., Kawakani, C. T., and de Alvarenga, S. C. (2017). A survey of intrusion detection in internet of things. *Journal of Network and Computer Applications*, 84:25–37.