芝浦工業大學
SHIBAURA INSTITUTE OF TECHNOLOGY

Traffic Modeling and Anomaly Detection for Internet of Things

a Dissertation Submitted to the
GRADUATE SCHOOL OF ENGINEERING AND SCIENCE OF THE
SHIBAURA INSTITUTE OF TECHNOLOGY

by

NGUYEN AN HUNG
Student ID: nb17504

in Partial Fulfillment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

MARCH 2021

Acknowledgments

For the appearance of this dissertation, I would like to convey my heartfelt gratitude and sincere appreciation to all great people who have supported me during the doctoral course.

First of all, I would like to express my deepest gratitude to my supervisor, Professor **Takumi Miyoshi**, and my co-supervisor, Associate Professor **Thomas Siverston** for their guidance, support, and encouragement throughout my research at Multimedia Information Network Laboratory, Shibaura Institute of Technology. Their continued support led me to the right way in the path of research science. Their advice and passion for science will always be my guide to my life of scientific research.

I would also like to extend my appreciation to the review committee members. They contributed useful recommendations to improve the quality of my dissertation.

I would also like to thank all members of the Multimedia Information Network Laboratory and my all Vietnamese friends in Japan. They are my good friends who shared with me for studying and enjoying student life during my doctoral course. The enjoyable parties and trip camps are a part of my memories in Japan.

Finally, on this dissertation, I would like to extend my deepest gratitude to my family who always beside me during the years I stayed abroad for studying.

Thanks all for making me an unforgettable experience in my life.

Saitama, March, 2021

Nguyen An Hung

SHIBAURA INSTITUTE OF TECHNOLOGY

Abstract

Graduate School of Engineering and Science
Division of Functional Control System

Doctor of Philosophy

by Nguyen An Hung

The development of the Internet of Things (IoT) has made significant changes to people's lives now and in the future. With the development of the Internet, smartphones, and especially sensor devices, IoT is becoming the new trend of the world. IoT is defined as objects that can connect to the Internet. We enter the house, unlock the door, the lights will automatically light up where we stand, the air conditioner will automatically adjust the temperature, the music will automatically turn on to welcome us, and so on. These things are becoming familiar in everyday life with IoT technology. However, accompanied by the explosion of IoT because its utilities will increase security risks, the more connections are created, the more widely shared data, the more many security vulnerabilities. As in the past, we studied Internet traffic when it became popular, so understanding, modeling, and classifying IoT traffic is now more necessary than ever. The main objectives of this research were to solve the problem of IoT traffic understanding by using a traffic generator dedicated to the IoT environment as well as identify smart devices and detect anomaly in an IoT network. In this dissertation, I designed a novel IoT traffic generator called IoTTGen. I generate synthetic traffic for smart home and bio-medical IoT environments. Simultaneously, I also build a smart home testbed to validate and compare with generated traffic from IoTTGen. Then, I have a visual observation of IoT traffic properties by Behavior Shapes. My generator succeeds in capturing the characteristics of the IoT traffic. Additionally, I also proposed a new method to identify IoT devices based on traffic entropy. I compute the entropy values of traffic features and I rely on Machine Learning algorithms to classify the traffic. My method succeeds in identifying devices under various network conditions with performances up to 94% in all cases. My method is also robust to unpredictable network behavior with anomalies spreading into the network. In my future research, I intend to experiment with more distinct environments. I will also consider other scenarios and cybersecurity threats.

Contents

List of Figures

List of Tables

List of Abbreviations

IoT	Internet of Things
IoTTGen	IoT Traffic Generator
BS	Behavior Shape
FP	False Positive
FN	False Negative
ML	Machine Learning
NN	Neural Network
RF	Random Forest
SVM	Support Vector Machine
TP	True Positive
TN	True Negative
CM	Confusion Matrix

Chapter 1

Introduction

This chapter aims to introduce the overview and motivation so that readers could comprehend the importance of this dissertation, which studies an approach for modeling and identifying IoT traffic and detecting anomalous IoT traffic. Firstly, the existing problem will be presented in this chapter. Secondly, beneficial contributions are suggested and some examples are also provided. Finally, a summary of this chapter and the structure of this dissertation are described.

1.1 Overview

The Internet of Things (IoT) has opened up a new cyber-physical technological era. The rapid development of IoT is already impacting our daily life. The prevalence of smart homes, smart cities, and industries 4.0 are also notable. IoT devices are known as devices that can exchange data over a network without human interaction. International Data Corporation (IDC) forecasts that by 2025 about 41.6 billion IoT devices will be connected to the Internet and the total volume of generated data will reach 79.4 zettabytes (ZB) of data [61]. This growth is not expected to slow down and may increase several times over the next few years.

Smart devices are also becoming more innovative: for instance, smart homes can be equipped with several sensors that can remotely control video surveillance, lighting, and/or heating system [62]. Humans use these IoT devices in their daily lives; therefore, they become an essential part of our lives. More and more people are using IoT devices connected to the Internet and thus, the traffic generated from IoT devices is increasing year by year. The demand for using IoT devices has increased accordingly.

1.2 Motivation and goal

Just as I know about Internet traffic, it is also important to understand the specific characteristics of IoT traffic. It contributes to the control, performance evaluation, or security of IoT devices operating as intended. IoT traffic is different from Internet traffic, so I also need to understand the characteristics of IoT traffic. Thereby, I need to apply security policies as well as an individual approach to IoT traffic. Alternatively, understanding the IoT traffic model also helps us research and manage the IoT network better.

Researchers and network administrators still lack tools to generate, observe, and display the behavior of IoT devices. Having the ability to generate IoT traffic is of great help for researchers testing and modeling traffic in an IoT network to validate that devices are working as intended. Besides, with the ability to observe and display the behavior of IoT devices promptly also helps in detecting anomalies and managing the network in a better way.

The goal of this thesis is first to describe how IoT traffic can be accurately emulated. Specifically, the focus is on traffic between IoT devices and gateway, so here I have designed an IoT traffic generator that can accurately mimic the real traffic in the IoT environment. My second goal is to classify IoT devices by using a new method based Machine Learning algorithms and entropy values. Finally, I aim to detect IoT traffic anomalies with using synthetic traffic generated by IoTTGen and real traffic collected from testbed.

1.3 Challenges

With the widespread popularity of IoT devices today, there is an increase in security vulnerabilities and network attacks. The benefits from IoT devices are apparent, there is also a danger when it falls into the hands of hackers. For example, a security camera can protect our home from unauthorized intrusion, but when the security camera data is stolen, the victim's personal information images are most likely easily spread or it becomes an advantage for breaking into the house more comfortable. Therefore, securing IoT devices from attack is more important than ever and a challenge in this day and age.

Regarding security in the IoT environment, commercial devices such as laptops, PCs, or smartphones are all equipped with anti-virus software to protect and detect unusual device activities. However, IoT devices have a limited resource, so I need a way to be tracked and detected from an external source such as the network middle-box or gateway. Moreover, with the features of IoT devices such as a large scale and heterogeneity, security becomes more complicated.

1.4 Contribution and Thesis Organization

In order to resolve the difficulties as mentioned above in modeling IoT traffic and detecting anomaly. The main contributions of this dissertation can be summarized as follows:

1. I designed IoTTGen - a novel IoT traffic generator. This is a packet-level traffic generator tool dedicated to the Internet of Things traffic. I used IoTTGen to generate custom parameter traffic and extracted parameter (extracted from real dataset) traffic. It also has been used to model smart home and bio-medical use case environment. I also extracted anomalous IoT traffic from a real dataset and study the IoT traffic properties by computing the entropy value of traffic parameters. My generator succeeds in capturing the characteristics of the IoT traffic, which can be visually observed on Behavior Shape graphs.

2. In order to validating the effectiveness of IoTTGen and comparing between synthetic traffic and real traffic, I setup a testbed with 5 IoT devices emulating a smart home environment. These devices include a hub, a camera, a bulb and two plug. The trace of IoT traffic are collected for a period of one week. The results show that my generator succeeds in modeling the IoT traffic and capture its main characteristics.

3. Based on traffic traces obtained from testbed, I proposed a new method to identify and classify IoT devices. I analyzed the traffic characteristics of each IoT devices and compute the traffic entropy while applying the machine learning algorithms to classify IoT devices. The proposed approach is trained, validated and demonstrated to achieve over 94% accuracy under various network conditions.

My method is also robust to unpredictable network behavior with anomalies spreading into the network.

The organization of this thesis is described as follows:

Chapter 1: Introduction. The motivation and goal of this research were described in this chapter. In addition, the research overview was also presented. The primary contributions of this research were also concretely summarized in this chapter.

A literature review is provided in Chapter 2. The issues related to data collection and modeling are thoroughly resolved in detail from Chapter 3 to Chapter 5. Chapter 6 concretely concludes the work and figures out future work direction.

Chapter 2: Literature review. In this chapter, I discuss previous researches related to the topics of this dissertation.

Chapter 3: Generating IoT traffic: A Case Study on Anomaly Detection. This chapter proposes a novel IoT traffic generator called IoTTGen. In this chapter, I present a packet-level traffic generator tool used to study the properties of IoT traffic. IoTTGen has been also used to model different IoT use case environments such as smart home or bio-medical environments. This chapter also make a comparison between synthetic traffic and measured traffic in various condition.

Chapter 4: Entropy-based IoT Devices Identification. This chapter proposes a new method to to identify IoT devices. The proposed approach is to combine entropy value and machine learning algorithms in order to classify IoT devices under various network conditions.

Chapter 5: Conclusion and Future Work. This chapter concludes the dissertation by which advantages as well as remained difficulties were discussed. Finally, research directions of great interest for the future work were figured out.

Chapter 2

Related works

In the previous chapter, I described the introduction of this dissertation to describe what motivation of this research was and explain what problems were herein addressed, including this research's objectives and contributions. This chapter aims to provide an idea of the state of art of the corresponding areas. I will review the existing works and thoroughly discuss their solution and problems. Corresponding to the stated challenges, this chapter is organized into six parts: traffic generator, IoT traffic characterization, identify and classify IoT devices, smart home testbed, entropy-based method, and anomaly detection.

2.1 Traffic generator

In order to evaluate the performance of the network and characterize traffic, traffic generators are known as important and powerful tools. Many traffic generators have been developed in different forms, such as open-source software, research projects, or even commercial products. We can mention a few popular traffic generators as follows:

- D-ITG (Distributed Internet Traffic Generator) [1]: is a platform that can generate both IPv4 and IPv6 traffic.
- PackETH [2]: is a GUI and CLI packet generator tool that can create and send any Ethernet packet or sequence of packets.

- PktGen [3]: is a traffic generator that can generate packets with a fixed delay and sequence numbers
- Iperf [4]: is a traffic generation tool that allows user to experiment TCP and UDP parameters such as delay, bandwidth, window size and packet loss.
- Ostinato [5]: is an open-source network traffic generator with a friendly GUI.
- IP-Traffic [6]: is a commercial traffic generator for IP networks using UDP, TCP and ICMP protocols. It is developed by ZTI-Telecom.

Table 2.1: References for traffic generators

Traffic generators	Transport protocol	Interface	Open source	IoT traffic
D-ITG [1]	UDP, TCP, DCCP, SCTP, ICMP	Command line	✓	x
PackETH [2]	UDP, TCP, ICMP, IGMP	GUI, Command line	x	x
PktGen [3]	UDP, TCP, ARP, ICMP, GRE, MPLS	Command line	✓	x
Iperf [4]	UDP, TCP, SCTP	Command line	✓	x
Ostinato [5]	TCP, UDP, ICMP, IGMP, MLD	GUI	✓	x
IP-Traffic [6]	UDP, TCP, ICMP	GUI	x	x
IoTTGen [7]	UDP, TCP, ICMP	Command line	✓	✓

We cannot deny that the traffic generator is one of the best ways to inject traffic into the network for utilization by other devices. Furthermore, it is useful in evaluating the performance of devices under test. Most previous traffic generators focus on the Internet traffic, but I just consider IoT traffic for my research. Therefore, I used IoTTGen as a traffic generator dedicated to the IoT network.

2.2 IoT Traffic Characterization

The Internet of Things has become a tremendous topic but there is still only a few studies investigating the IoT traffic and its impact on networks. Shahid et al. [28] collect traffic from smart home sensor devices. They visualize IoT traffic with t-SNE method in order to classify network traffic for each device. Furthermore, Koroniotis et al. [20] deploy a Bot-IoT testbed and they made their data publicly available. They analyzed their dataset through machine learning methods for forensics purposes. Ferrando et al. [26] rely on streaming data analytics to detect abnormal IoT traffic. They visually observe the traffic from heterogeneous sensor devices with Behavior Shape. My works differ as I aim at characterizing IoT traffic, and I design an IoT traffic generator for modeling traffic in various environments. I also apply my generator to the case study of IoT traffic anomaly.

2.3 Identify and classify IoT devices

Machine Learning algorithms have already been used to identify and classify IoT devices. In [46], Shahid et al. use t-Distributed Stochastic Neighbor Embedding (t-SNE) to recognize the type of four kinds of IoT devices. Feng et al. [47] rely on a hybrid IoT device classification framework by combining Empirical models with advanced machine learning models to classify IoT devices. Bezawada et al. [48] build a static and dynamic behavioral model based on packet header and payload features. By using multiple machine learning classifiers, they fingerprint IoT device types with high accuracy. In order to classify and distinguish IoT devices from other devices, Ortiz et al. [49] rely on a Long Short Term Memory (LSTM) neural network, which automatically learns features from the device traffic. This work shows that it is feasible to identify devices after automatically learning a few TCP-flow samples with high accuracy.

Table 2.2: References for methodology

Scheme	Methodology	Goal
Shahid et al. [46]	t-Distributed Stochastic Neighbor Embedding	Recognize type of devices
Feng et al. [47]	A hybrid IoT device classification framework (Empirical models and Advanced ML models)	Classify IoT devices
Bezawada et al. [48]	Use multiple machine learning classifiers	Fingerprint IoT device types
Ortiz et al. [49]	Long Short Term Memory neural network	Classify IoT device
My research	Combine traffic entropy value and ML algorithm	Identify and classify IoT device

2.4 Entropy-based method

Regarding Internet traffic, there have been several studies relying on the entropy-based method as a good candidate to detect anomalies [41] [42] [52]. Among them, Bereziński et al. [42] show the ability to detect a broad spectrum of anomalies by using supervised learning with parameterized entropy. Shukla et al. [53] computed entropy values for a features vector and a list of legitimate traffic is then provided for filtering the traffic. Callegari et al. [54] propose an intrusion detection system by measuring the entropy associated with the traffic descriptors. They identify traffic features and detect anomalies with different network scenarios.

2.5 Anomaly detection

For modeling attacks, Arnaboldi et al. [30] propose an IoT system model for generating synthetic DoS. Erlacher et al. [31] propose an automated system for generating attack traffic for network intrusion detection system. Huang et al. [32] implement attack models in Omnet++ simulator tool and they evaluate the performances of their intrusion detection system for sensor networks. Salem et al. [34] propose a framework to detect anomalous changes in the medical wireless sensor network. Along with Cassas et al. [35] and Papadopoulos et al. [36], several studies rely on Machine Learning methods for classification and anomaly detection.

Furthermore, there have been other studies for detecting anomalies in IoT traffic. Ozcelik et al. [55] proposed a model aiming to detect and mitigate IoT-based DDoS attack by investigating SDN's capabilities in edge IoT networks. The use case of Mirai malware is, therefore, evaluated. Intrusion Detection System has also been proposed for IoT. Fu et al. [56] presented an automata-based intrusion detection method for Internet of Things. By using an extension of Labelled Transition Systems, three types of IoT attacks can be detected: jam-attack, false-attack, and reply-attack. Similarly, Gajewski et al. [57] proposed a two-tier Intrusion Detection System in the smart-home environment, which can identify network attacks by using neural networks based on monitoring records.

Besides, based on the analysis of the behavior of attackers, Martin et al. [58] combined three practical techniques: honeypot, deep packet inspection (DPI), and a realization of moving target defense (MTD) in port forwarding to detect anomaly. Summerville et al. [59] proposed a deep packet anomaly detection approach with the ability to distinguish between normal and abnormal payloads. The bit-pattern matching technique was used to perform feature selection. However, this method is limited due to the resource constrained of IoT devices, while our works focus on consumer IoT devices. Moreover, several popular IoT devices such as Amazon Echo or Tp-link Bulb rely on the TLS protocol for communication encryption. Thus, the DPI approach is more difficult in the context of smart IoT Devices and use more resources. Besides, using the DPI method without the user's permission can also be considered a violation of the information privacy of the user [60].

2.6 Smart home testbed

Many previous research works deployed IoT testbeds that aim to research the IoT network. There are large-scale testbeds facilities such as FIT IoT-Lab [50] or WISEBED [51] which used a large number of sensors from different vendors. However, for reproducing real a smart home environment, most testbeds are smaller scale testbed equipped with commercial IoT devices as shown in the Table 2.3. Indeed, for a consumer user, the number of devices in a smart home is mostly limited (e.g., a few for each room, etc.) However, the way to build a testbed of these research works is also different. For example, [8], [9] and [28] use only one manufacturer per device while [10] uses three different manufactures for camera or [47] uses three different manufactures for plug. Besides, some testbeds [9] [10] have additional non-IoT devices or testbed from [47] includes Raspberry Pi sensors. These studies all use smart home testbed and Machine Learning to identify the class of IoT devices (e.g., hub, camera, plug, etc.). In my research, I also deploy a small-scale testbed as most previous papers are used.

Table 2.3: References for IoT testbeds

Testbeds	# of		Manufactures	Type of IoT devices
	IoT devices	non-IoT devices		
Adjih et al. [50]	2845	0	WSN430, M3, A8, Turtlebot, Wifibot	Sensor, Mobile robot
Chatzigiannakis et al. [51]	750+	0	Pacemate, iSense, TelosB, MicaZ, SunSPOT, Tmote Sky, MSB-A2	Sensor
Feng et al. [47]	11	0	Samsung, Insteon, YI, Belkin, Wemo, Z-Wave, Raspberry	Hub, Camera, Plug, Sensor
Ammar et al. [10]	7	5	D-Link, Panasonic, TRENDnet, Philips, Chromecast	Camera, Light, Speaker, TV
Anthi et al. [9]	7	3	Amazon, Belkin, TP-Link, Hive, Apple, HP	Hub, Camera, Plug, Sensor, TV, Printer
Apthorpe et al. [8]	4	0	Amazon, Nest, Belkin, Sense	Hub, Camera, Plug, Monitor
Shahid et al. [28]	4	0	TP-Link, Nest, D-Link	Light, Camera, Plug, Sensor
Our testbed	5	0	Amazon, TP-Link, Lefun, Teckin	Hub, Camera, Light, Plug

Chapter 3

Generating IoT traffic: A Case Study on Anomaly Detection

3.1 Introduction

The Internet of Things (IoT) has been rapidly extending these last years and has already an impact on our daily life. There are more and more sophisticated sensor devices remotely accessible through the Internet, performing complex tasks such as collecting data or monitoring the environment for providing new services. For instance, Cisco has predicted that approximately 28.5 billion devices will be connected to the Internet by 2022 [13]. This trend leads to new kind of applications for various environments such as smart home, smart healthcare, smart industry, smart cities, etc. As an example, it is now common to have several sensors at home to control the heating system, video monitoring, or lighting system.

As it is expected that the Internet of Things will count for a major part of the Internet traffic, there is still only a few studies for characterizing IoT traffic. Besides, IoT is also facing new challenges regarding cyber-security and privacy; the rise of IoT has also unveiled new vulnerabilities for devices. Indeed, even though the monitoring environment can provide new services to users, the collected data also conveys critical information about users and their privacy. For instance, Mirai DDoS Botnet has seriously slowed down the Internet in 2016 [40]. Some exploits have also reported the virtual Carjacking of a vehicle [12]; it has also been shown that the data from Heart monitoring systems for babies were unencrypted, and alert in case of an emergency could have been modified and have a strong impact on the medical process [14].

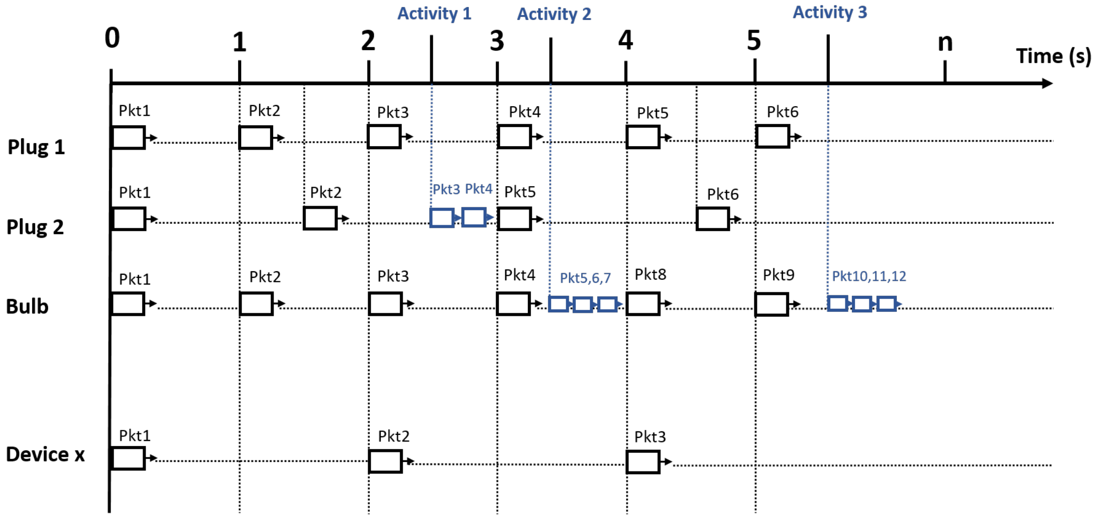
Traffic generators are essential tools for evaluating the performance of the network and characterize traffic [15]. There have been already several traffic generators [16] such as Iperf, PackETH, D-ITG, and Ostinato but they focus on the Internet traffic while IoT traffic has different characteristics, such as heterogeneity of source, multiple sources, new traffic pattern and different supported services [17]. As IoT traffic show different properties, it is not clear which data should be collected, which rate and from which source. For instance, there are plenty of different kinds of IoT devices, from the smart camera to smart light, also designed by various manufacturers, presenting different functionalities and whose network traffic pattern is significantly different [18].

In this context, knowing the characteristics of IoT traffic could help to prevent security threats and mitigating vulnerabilities. For instance, a network administrator can detect abnormal changes in IoT traffic and early detect attacks and provide counter-measures.

In this paper, I design IoTTGen, a novel IoT traffic generator. IoTTGen is a packet-level traffic generator tool and it is used to study the properties of IoT traffic. IoTTGen has been used to model different IoT use case environments such as smart home or bio-medical environments. To the best of my knowledge, IoTTGen is the first tool for generating IoT traffic and study its characteristics.

I then use IoTTGen to generate IoT Traffic and also anomalous IoT traffic from the real dataset. The entropy of traffic parameters is computed and I can visually compare the traffic on Behavior Shape graphs. My traffic generator shows its ability to capture the different properties of IoT traffic. By visually comparing the traffic shape, it is possible to detect traffic anomalies and react accordingly to security threats in the network.

The remainder of this paper is organized as follows. Section II introduces IoTTGen, my novel IoT traffic generator. Section III presents the experiments I performed with my generator to study the IoT traffic properties. Section IV analyzes the results of my experiments. Section V surveys the related work while concluding remarks are in Section VI as well as future perspectives.



Timestamp	Name of Device	IP Source	IP Destination	Port Source	Port Destination	Packet Size
T_1	Plug 1	192.168.100.10	192.168.100.1	21	80	10

Figure 3.1: IoTTGen - Packet Generation process

3.2 IoT Traffic Generator

3.2.1 Overview

IoT traffic is the aggregation of packets generated by several devices that could come from different environments such as smart home or smart cities. These environments involve several sensors that are dedicated to specific tasks such as monitoring system or collecting cyber-physical values (temperature, humidity, etc.) Thus, compared with Internet traffic where traffic has some human-generated aspects (flash-crowd, popularity, etc.), IoT traffic can be more easily predicted as sensor devices are deployed to perform continuously the same tasks and generate periodically the same amount of data. Some alerts can also occur, adding less predictable traffic behavior, but they are still part of the way of working of the sensor devices. Thus, I design IoTTGen, a packet-level IoT traffic generator tool. My packet-level generator is able to finely tune all the feature parameters of the traffic such as packet size or time interval between packets. I believe this tool can be essential for modeling IoT traffic, to study its characteristics, to model unpredicted traffic behavior, and to understand IoT traffic anomalies.

3.2.2 Architecture

With IoTTGen, different kinds of IoT environments can be modeled, in which each sensor device produces its own traffic trace according to its functionality and characteristics. For instance: one may expect that a video recording camera will generate continuous flows of data with large packet-size, while smart plug generates small-size packets at a slow pace. IoTTGen is designed to easily configure the parameters of each device. All the packets generated for each device are stored into a single trace file, and different formats are supported such as pcap, csv, or txt.

IoTTGen architecture is composed of the following components:

- Device configuration module,
- Packet creator module,
- Main controller.

The device configuration module defines the configuration of IoT devices such as packet size, port number, payload, and period of time between packets. In order to add a new device (smart light bulb or camera), the user needs to define a new device template.

The packet creator module is forging the packets based on the device configuration from the previous module. I rely on Scapy [19] to forge packets and generate real packet traffic. Depending on its needs, it is also possible to generate only packet traces in text format. For the long-duration experiment, it reduces drastically the duration of the packet generation process.

The main controller is responsible to control the execution of the IoTTGen. The controller will extract the parameters provided by the device configuration module and instantiate the packet creator to forge packets or provide packet trace. The main controller will be also responsible to merge each device trace into a single trace. It is of course still possible to generate one trace per device.

3.2.2.1 Device generated traffic

IoT devices are dedicated entities performing continuously their task such as sensing the environment, transferring the data among objects or users automatically without human intervention. Thus each device is generating continuously traffic even though there is no user requesting explicit information. Fig. 3.1 illustrates the packets generation process of IoTTGen. The period for generating packets varies for each kind of device and can be configured. For example, in Fig. 3.1, devices have a period of 1s 1.5s, 1s, and 2s. As for generating traffic, all the devices are synchronized and have the same time origin, but it is also possible to configure the starting time and add some delays among packets and periods to reduce the synchronicity of devices. Then, each device generates a different amount of packets with a different time period for the entire duration of the scenario.

3.2.2.2 Human generated traffic

Besides, IoTTGen can also generate traffic triggered by human activities. Each device has different event patterns, such as the smart plug has turn-on event and turn-off event, so IoTTGen can also model the traffic generated by human activities. The user also can create different scenarios for using IoT devices as regular daily activities. For example, when coming home, a user can turn on the bulb and plug, launch Spotify music service on the hub, and access the record activities of the camera. In Figure 3.1, Plug 1 is switched on at time Activity 1, and, as an example, two packets are generated for Plug 1. Then, when Bulb is switched on at time Activity 2, IoTTGen generates packets accordingly. Thus, human activities traffic can be generated easily using IoTTGen.

3.2.3 Use Case

My generator has been designed to emulate various kinds of scenario configurations with a large number of devices. As a larger-scale use case, let me consider a scenario in which a small company equipped its office with 50 smart devices as follows: 4 hubs for each main open space, 8 Cameras, 20 Light Bulbs, and 20 electric Plugs for each Desk. As for the experimental setup, I generated a two hour-long traffic trace in which there is no human activity, i.e., there is only the signaling traffic of the smart devices. No device is actuated by users (e.g., turning on/off light, or plug, etc.). IoTTGen generated the traffic for all these devices, whose traffic parameters are derived from previous measurements (packet size, number of packets, etc.).

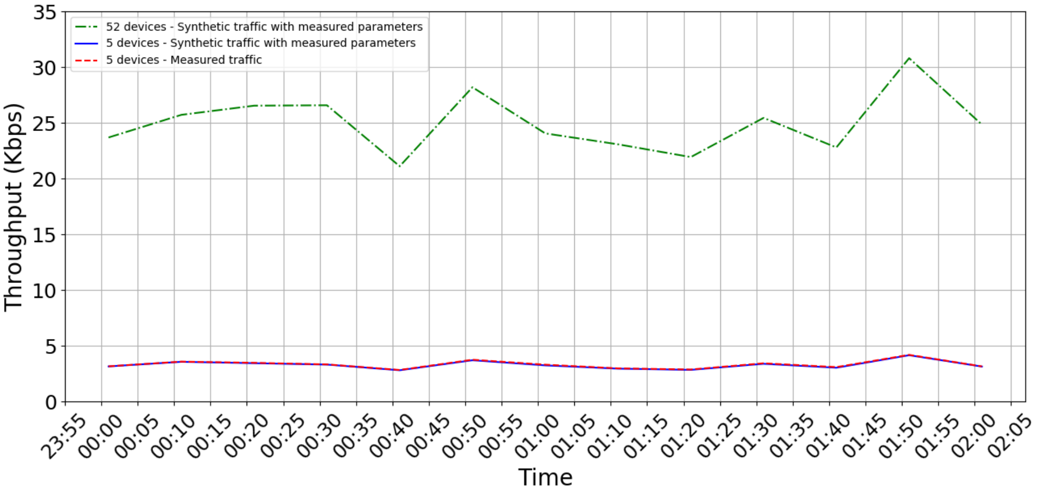


Figure 3.2: Measured traffic, Synthetic traffic with measured parameters of 5 devices and 52 devices

The overall traffic of this experiment is shown in Figure 3.2. In the figure, I also show the overall traffic of the smart home experiments: (i) synthetic traffic generated by IoTTGen and (ii) measured traffic from my smart-home testbed. As it was expected, the synthetic and measured traffic for the smart-home testbed reach the same bandwidth (4Kbps) and they overlap: IoTTGen is able to capture the characteristics of the traffic and to generate traffic accurately. For the larger-scale experiments with 50 smart devices, there is obviously more generated traffic (25Kbps). This use case is to show the ability of my generator to emulate any scenario and generate the traffic accordingly.

In the following, I will rely on the smart-home scenarios as it is the more popular set up configuration with commercial smart devices. IoTTGen is however able to generate the traffic of any kind of smart device (e.g., weather, motion sensors) by providing the traffic profile of these devices.

In the following section, I will show how I apply IoTTGen into two different IoT environments and generate a different kind of traffic.

3.3 Experiments

In this section, IoTTGen is used for implementing experiments and generating synthetic IoT traffic for two different IoT use cases: smart home and bio-medical environments. I also generate malicious traffic by extracting anomalous traffic from the real dataset [20]. Thus, I aim to investigate the properties of IoT traffic and to study the impact of malicious traffic. All the experiments have been conducted on a PC with Intel Core i7-7700 3.6GHz processor and 8GB of memory. The operating system is 64-bit Windows 10 Professional.

3.3.1 Smart Home Environment

A smart home is a house equipped with (multiple) cyber-physical sensors allowing inhabitants to obtain information on their environment (e.g., temperature) and to control and monitor it remotely (e.g., turn on types of equipment, etc.). One can mention typical sensors for temperature, humidity, light control, smart hub, etc. In my smart home environment, the devices are connected to the Internet with Wi-Fi through a home gateway, which can control the flow of information among smart appliances to the remote network. Thus, remote users can also access data and control home sensors with dedicated devices such as smartphones, tablets, computer, etc.

3.3.1.1 Smart Home Scenario

In order to generate IoT traffic for a smart home, I set up an experiment with a four-room house equipped with 13 smart devices as follows: one smart hub (e.g., Amazon Echo) which is in charge of controlling other devices; then, each of the four-room is equipped with a smart camera (e.g., Belkin NetCam), a smart light (e.g., Lifx Bulb), and a smart plug (e.g., TP-Link Smart Plug). The experiment is described in Fig. 3.3.

I consider two distinct sets of parameters for the smart home environments (a) a case with custom parameters; (b) another case with parameters extracted from the dataset [21]. The rationale is to show that the different parameters will have an impact on generated traffic. Table 3.2 summarizes the custom parameters that I model based on the functionality of each device. For instance, a smart plug generates periodically short-size packets (100 Bps) as they are low bandwidth sensor devices; Smart hub may have larger packet size (200 Bps) for a management purpose but with the same period. Differently, I consider that a smart camera is a high bandwidth device and it is continuously generating large-size video

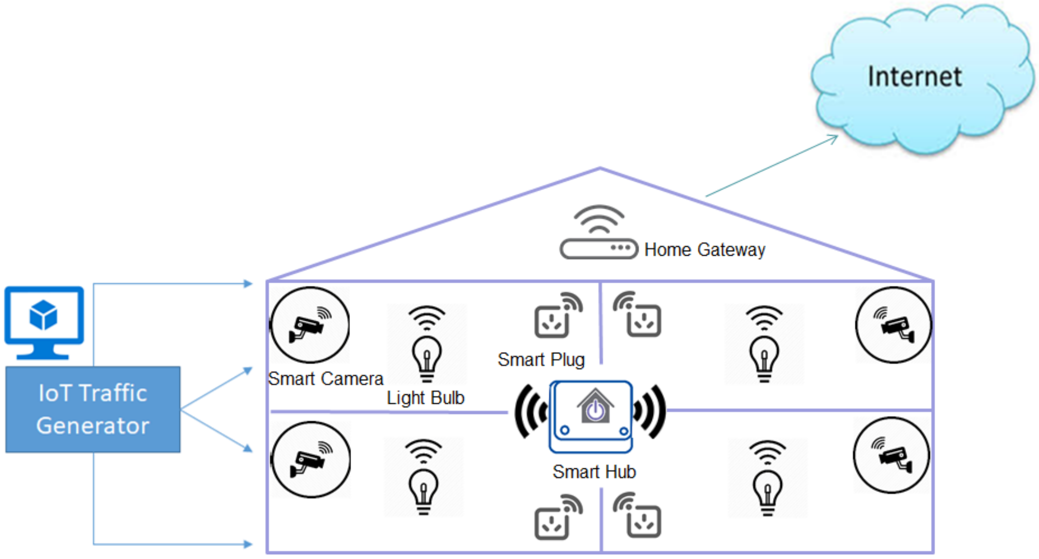


Figure 3.3: Smart Home Scenario

Table 3.1: IoTTGen - Extracted Parameters for Smart Home

	Period (s)	Packet Size (B)
Smart Hub	2.77	144
Smart Light	3.2	94
Smart Camera	2	100
Smart Plug	10	120

Table 3.2: IoTTGen - Custom Parameters for Smart Home

	Period (s)	Packet Size (B)
Smart Hub	1	200
Smart Light	1	100
Smart Camera	0.05	1,000
Smart Plug	1	100

packets (1,000 Bytes) at a shorter period (50ms) for a video bitrate at 160 Kbps. Table 3.1 shows the parameters as they have been extracted from the previous study [21]. With the same network configuration, smart hub, smart light, smart plug, and smart camera generated short-size packets (144, 94, 120, and 100 Bytes) at distinct periods (2.77, 3.2, 10, and 2s).

The different parameter values (custom vs. extracted parameters) will have an impact on the properties of the traffic and it will be described next in Section IV (Fig. 3.12).

For this smart home environment scenario, the overall smart-home traffic has been generated by IoTTGen for three durations: 8 hours, 24 hours and 7 days.

3.3.2 Bio-medical Environment

IoT technology is also used for bio-medical systems, such as real-time monitoring, personal healthcare, remote medical assistance, and alert devices. Sensors are embedded in medical devices and collect health data [22] such as body temperature, blood pressure, oxygen, heart rate, etc. With bio-medical systems, there is no need for frequent visits at the hospital: Doctors are now able to keep track of patients' health thanks to wearable devices worn by patients or monitoring devices in the room [23]. Thus, IoT could help to enhance the health condition of patients and future diagnostic.

3.3.2.1 Bio-medical scenario

For the bio-medical scenario, I conduct another experiment in which patients are diagnosed with various diseases and allocated to specific treatment rooms under doctors' supervision. Thus, I consider a use case medical center with four distinct rooms according to patients' treatment and monitoring process. Each room can host up to four patients and sensors are whether in the room or embedded into the body to measure physiological parameters. This scenario is illustrated in Fig. 3.4: a control unit (CU) for each patient is collecting information from sensors and detect whether significant physiological events occur for patients. Upon event detection, the CU sends an alert embedding the condition status of the patient. The frequencies of alarms/alerts have been extracted from real values [24], as well as the duration of alert ranging from 10 seconds to 5 minutes [25]. A total of 16 CUs are used in this scenario and the parameters for each device is presented on Table 3.3.

3.3.3 IoT Traffic

The IoT traffic traces statistics are presented in Table 3.4. As it was expected that the number of packets and the total amount (volume) are proportional to the total duration of the experiments, one can observe that the bio-medical environment

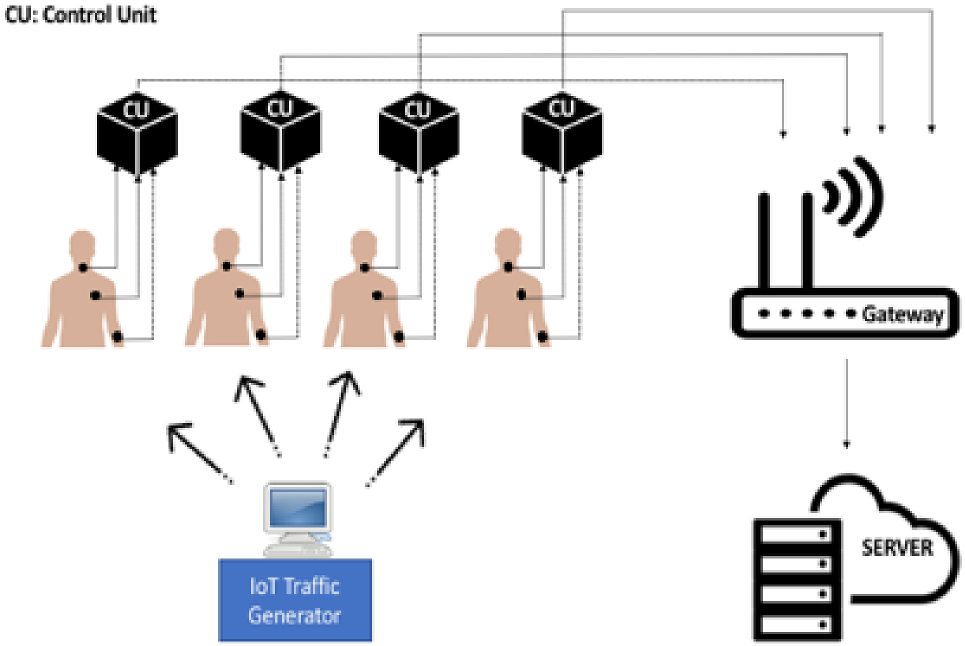


Figure 3.4: Bio-Medical Scenario

generated much more traffic than smart home. For the bio-medical scenario, there are much more packets generated as the period for each packet is shorter; the packet size is also larger and so does the total amount. It is noteworthy to mention that even though the two scenarios use a similar number of smart devices (13 for smart home and 16 for bio-medical), my IoTTGen succeeds in capturing the characteristics of the different traffic.

3.3.4 Anomalous Traffic

Besides IoT traffic for smart home and bio-medical environments, I also wanted to model IoT anomalous traffic. I, therefore, extracted from a public dataset [20] the traffic of several cybersecurity threats such as a) Port Scanning, b) Denial of Service (DoS) and c) Distributed Denial of Service (DDoS). The anomalous traffic statistics are presented on Table 3.5 and the total duration of the traces is 38 minutes. DDoS has been generated by 5 bots, while DoS and Port Scanning have been generated by a single bot.

Then, each of the malicious traffic traces is injected into my generated IoT traffic, and I obtain five different traces: one synthetic trace for both IoT environment (refer to IoT traffic hereafter) and four malicious traffic traces for each anomaly and a mixture of them (refer to DDoS, DoS, Port Scanning and Mix).

Table 3.3: Parameters for Bio-Medical Environment

Control Unit	Sensors	Period (s)	Packet Size (B)
CU1	Body Temperature	0.5	50
	Blood Pressure	0.5	150
CU2	Body Temperature	1	50
	Heart Rate	1	75
	Respiratory Rate	1	50
CU3	Electromyography (EMG)	0.5	500
CU4	Cardiography (ECG)	0.5	150
Alert		0.1	50

Table 3.4: IoT Traffic Traces Statistics

	Smart Home		Bio-Medical	
Duration	# Packets	Volume	#Packets	Volume
8 hours	115,517	11.47 MB	1,606,740	244.16 MB
24 hours	346,551	34.4 MB	4,781,820	730.64 MB
7 days	2,425,859	240.8 MB	33,055,490	4.97 GB

Table 3.5: IoT Anomaly Traffic Statistics (38 minutes) [20]

	# of Packets	Packet Size	Volume
DDoS	29,375,746	60 B	1.64 GB
DoS	27,634,013	60 B	1.54 GB
Port Scanning	896,335	100 B – 1 MB	841.7 MB

Fig. 3.5 shows the generated traffic used for the smart home scenario. The total duration of the experiment is 24 hours and cybersecurity threats occur at 06:00 during 38 minutes. The generated anomalous traffic is the mixture of the 3 different attacks and the IoT traffic.

3.4 Results

By designing an IoT traffic generator, my main objective is to characterize IoT traffic and be able to detect IoT anomaly. In order to study the different prop-

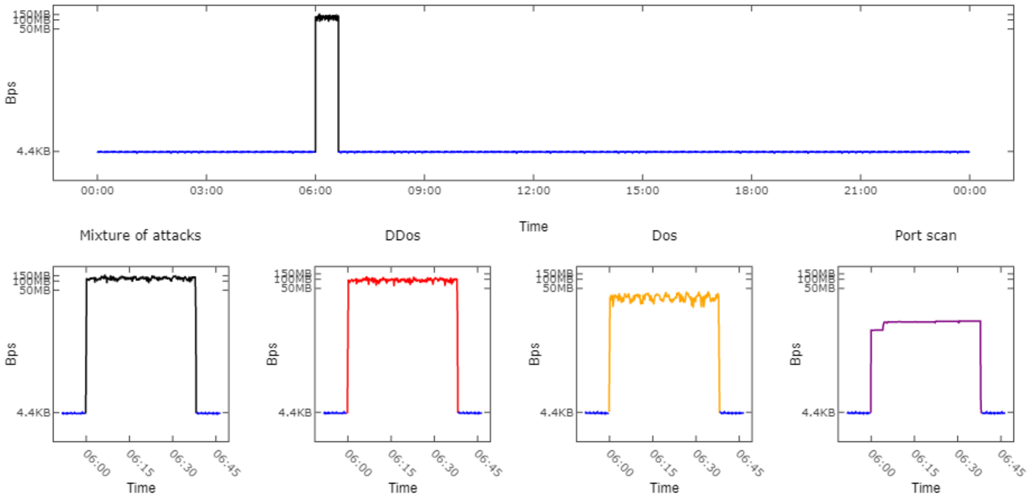


Figure 3.5: Generated IoT traffic and extracted Anomaly traffic

erties of the traffic, I compute the entropy of 6 traffic parameters: IP source, IP destination, port source, port destination, packet size, and bytes count. I then plot the Behavior Shape (BS) graphs [26] of the entropy and can visually compare the traffic properties.

3.4.1 Validation

Table 3.6: Smart-home Testbed

	Manufactures	Model	# of devices
Smart Hub	Amazon	Echo Dot	1
Smart Light	TP-Link	Kasa Wi-Fi Smart Bulb	1
Smart Camera	Lefun	Indoor Security Camera	1
Smart Plug	TP-Link	Wi-Fi Smart Plug	1
	Teckin	Wi-Fi Smart Plug	1

Prior to studying IoT traffic in smart-home and bio-medical environments, I aimed at validating the effectiveness of my generator. To this end, I deployed a lab-scale testbed with 5 devices as for a room in smart-home (Hub, Light, Camera, Plug) and measured traffic for 24 hours as shown in Table 3.6. I also used IoTTGen to generate traffic based on parameters extracted from Table 3.7. These parameters are similar to parameters of measured traffic from my testbed. Figures 3.6 and 3.7 show the BS of generated traffic from IoTTGen (based on parameters in Table 3.7). We can observe that the BS of the synthetic with measured parameters and measured traffic overlap for both figures as they exhibit the

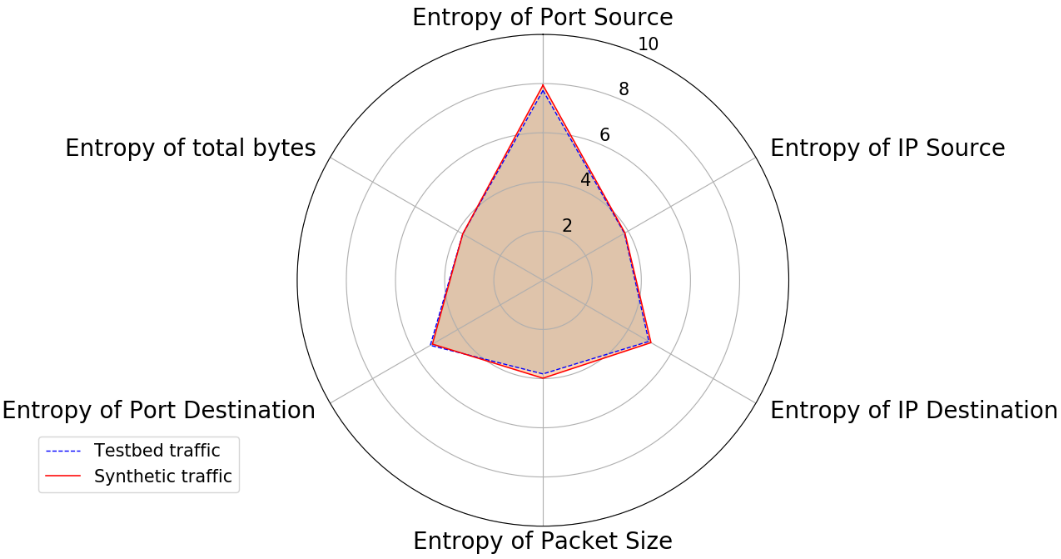


Figure 3.6: Behavior Shape: Synthetic traffic with measured parameters and Measured traffic

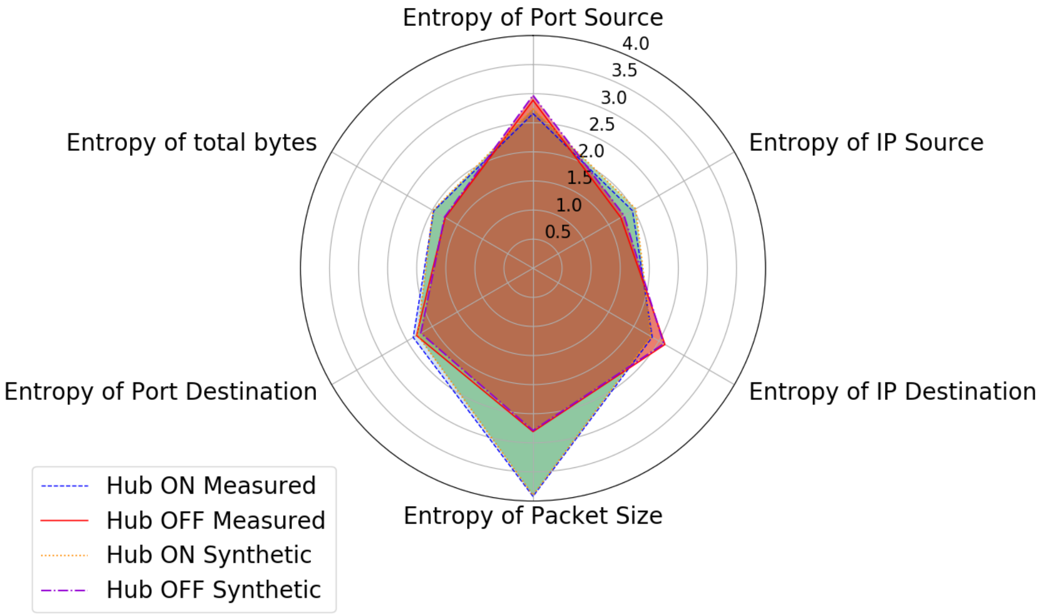


Figure 3.7: Behavior Shape of Hub traffic: Synthetic with measured parameters and Measured

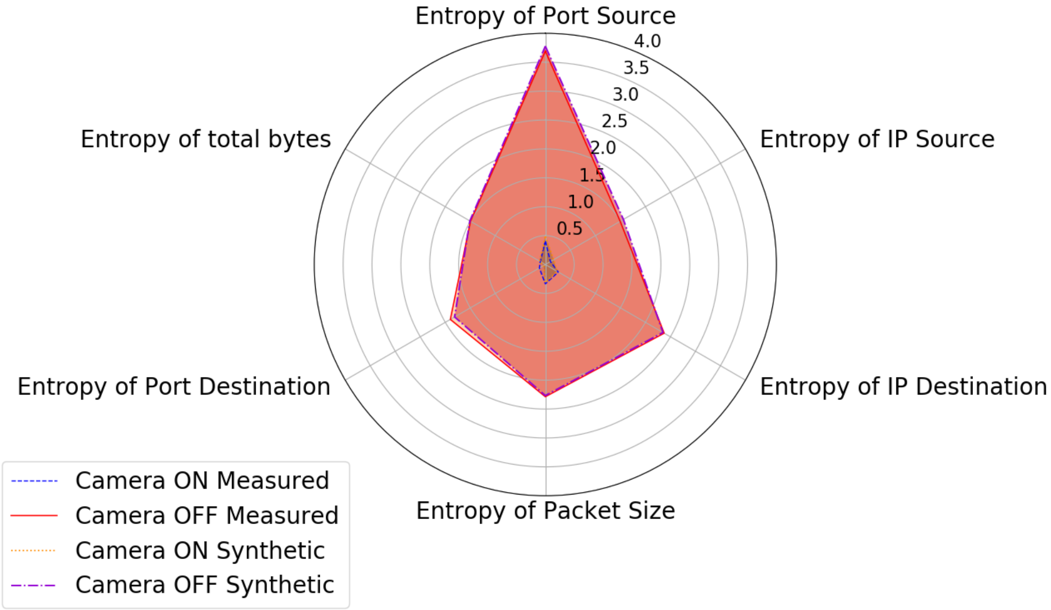


Figure 3.8: Behavior Shape of Camera traffic: Synthetic with measured parameters and Measured

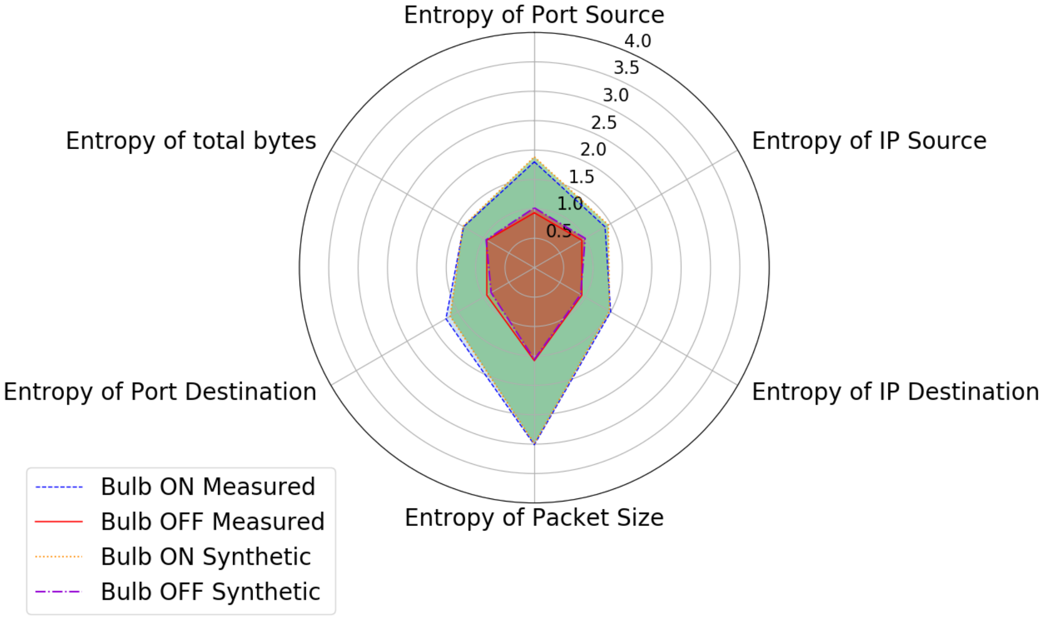


Figure 3.9: Behavior Shape of Bulb traffic: Synthetic with measured parameters and Measured

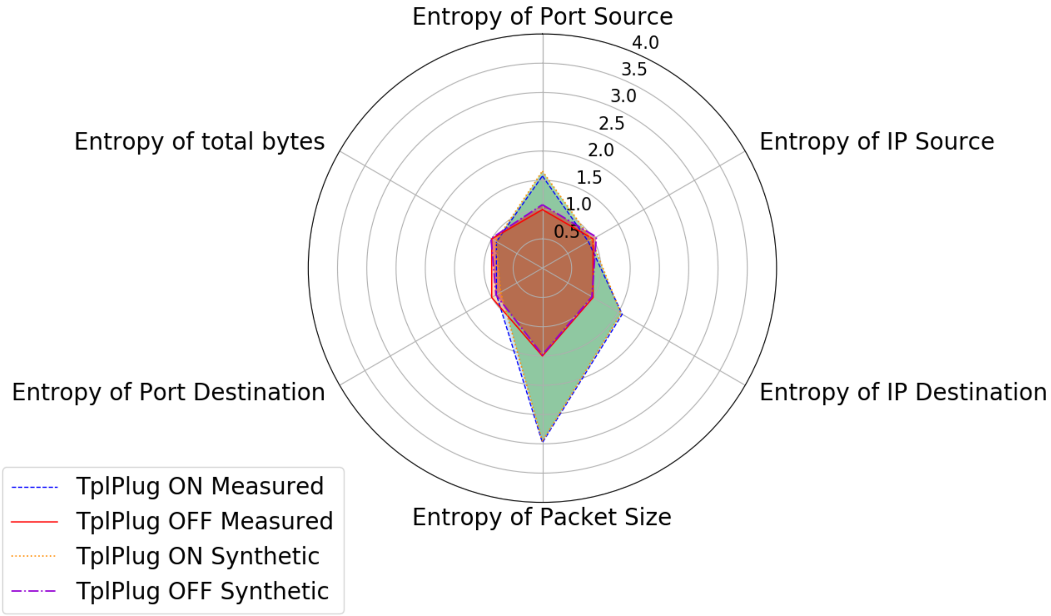


Figure 3.10: Behavior Shape of TplPlug traffic: Synthetic with measured parameters and Measured

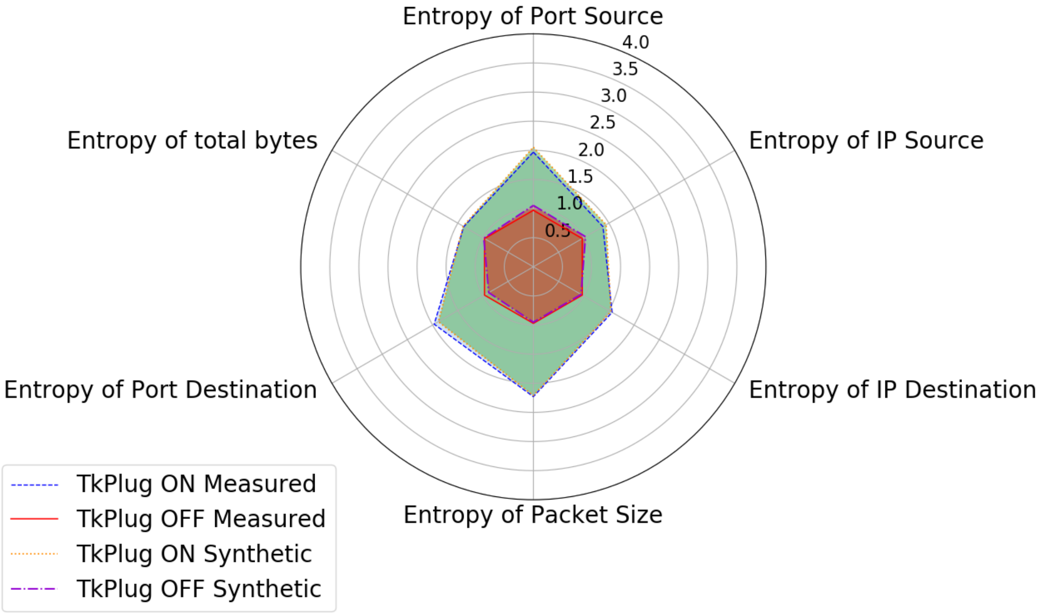


Figure 3.11: Behavior Shape of TkPlug traffic: Synthetic with measured parameters and Measured

same properties. In Figure 3.6, entropy values are nearly even in pairs of synthetic and measured traffic. In Figure 3.7, 3.8, 3.9, 3.10, and 3.11, it's similar when I focus only on each device with ON/OFF activity. It shows that my generator succeeds in modeling the IoT traffic and capture its main characteristics.

Table 3.7: Testbed - Measured Parameters

	Period (s)	Packet Size (B)
Hub	1.05	82.5
Camera	0.78	306
Bulb	12.88	80
TplPlug	46.73	85
TkPlug	12.01	85

3.4.2 Behavior Shape Traffic Analysis

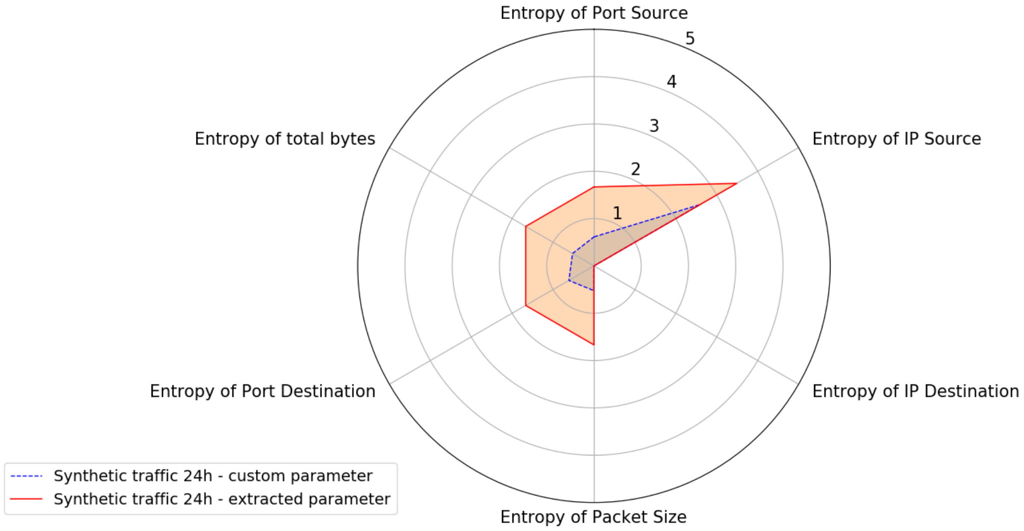


Figure 3.12: Behavior Shape: Custom and extracted parameter

Fig. 3.12, 3.13, 3.14 present the BS of the generated traffic for smart home and bio-medical environments from the experiments presented in Section III.

First of all, for all the figures, one can immediately observe that the entropy value for the IP destination parameter is equal to 0. This is due to the fact that for all the experiments, the traffic generated from IoT devices flows to the same destination, i.e., the gateway, and then there is no entropy as there is only a single destination.

In Fig. 3.12, IoT Traffic has been generated during 24 hours as a regular daily

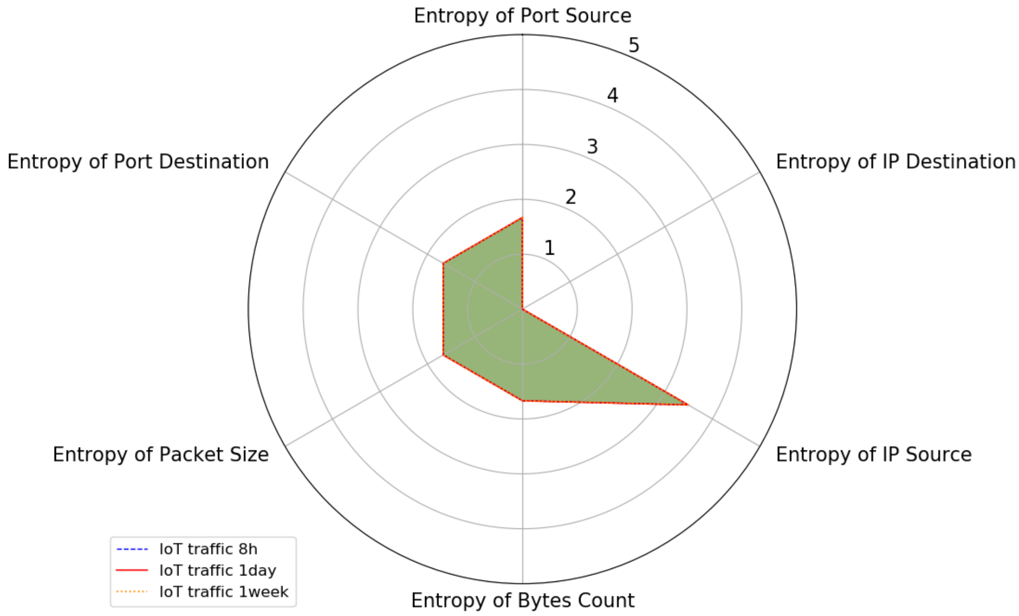


Figure 3.13: Behavior Shape: Smart Home Traffic

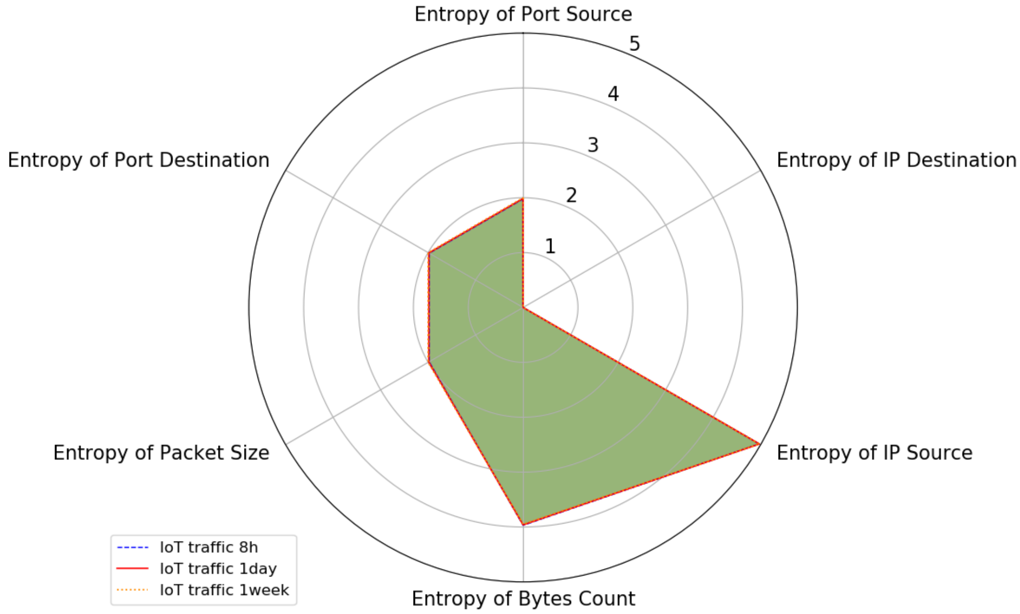


Figure 3.14: Behavior Shape: Bio-Medical Traffic

activity with the same network configuration (smart home environment). I can observe that parameter values have a direct impact on the shape of the traffic: extracted parameters exhibit a larger shape than custom parameters. Indeed, there is the same number of IoT devices but parameters are different (period, packet size) resulting in different BSs.

In Fig. 3.13 and 3.14, the experiments have been performed for 8 hours, 24 hours, and 7 days. For all these duration, the IoT traffic exhibits the same BS. Indeed, as synthetic traffic is generated, the traffic parameters stay unchanged during all the experiments and there is no evolution of the traffic (e.g., there is no new connected devices, or failure, etc.). Furthermore, the area for the BS in the bio-medical environment (Fig. 3.14) is larger than for the smart home environment (Fig. 3.13). Indeed, there are more IoT devices in the bio-medical environment and they are generating much more packets with a shorter period of time compared with smart home (Table III) and it will lead to higher entropy for all parameters.

3.4.3 Anomalous Traffic

Fig. 3.15 and 3.16 present the BS of anomalous traffic (see Section III.D) for smart home and bio-medical environments. Each figure shows the IoT traffic, each of the malicious traffic (DoS, DDoS, Port Scanning) and the aggregation of all traffic (IoT and anomalies). As I previously observed that the duration of experiments has no impact on the entropy values, for this experiment, I focus on the daily activity pattern and present the results of the experiment for 24 hours.

From Fig. 3.15 and 3.16, I can immediately observe that malicious traffic has an impact on the entropy values and the BS of the IoT traffic.

For all the malicious traffic, there is a higher entropy for the destination port in both environments. It is more pronounced for DoS and DDoS (14.5) than for Port Scanning (12.8 for smart home and 4.8 for bio-medical). Indeed, these anomalies come from security threats targeting a large number of destination ports. Thus, by computing the entropy value, it is possible to visually observe such anomaly in the network.

Regarding the entropy value of the IP source, IoT traffic has higher entropy than other malicious traffic. Remind that traffic is generated by 13 IoT devices

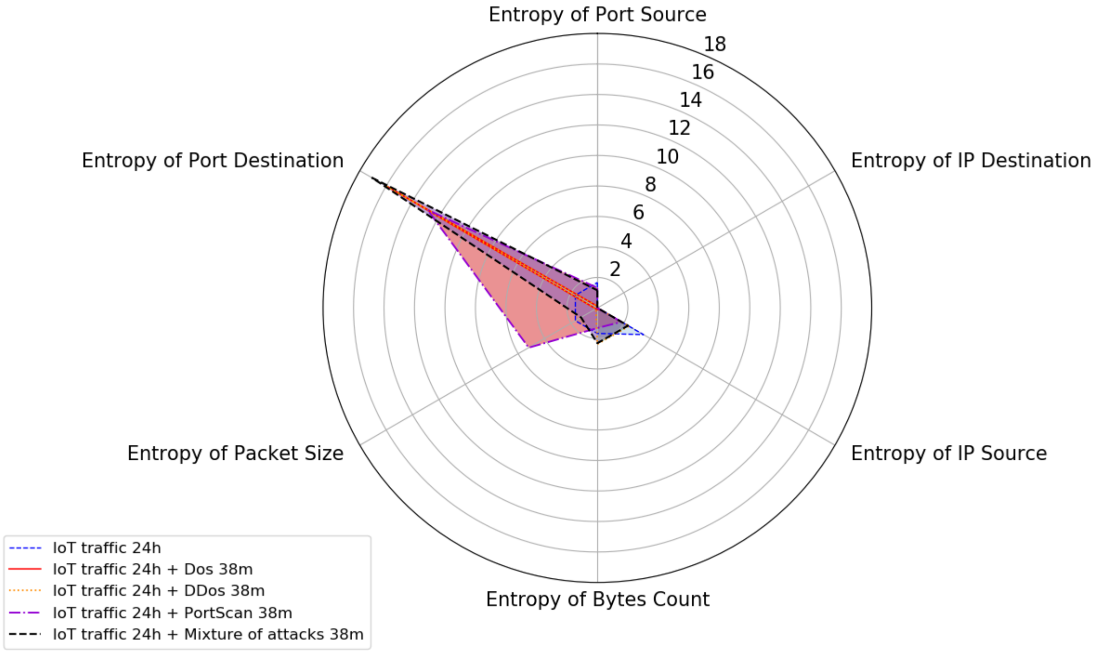


Figure 3.15: Behavior Shape: Smart Home and Anomalies Traffic

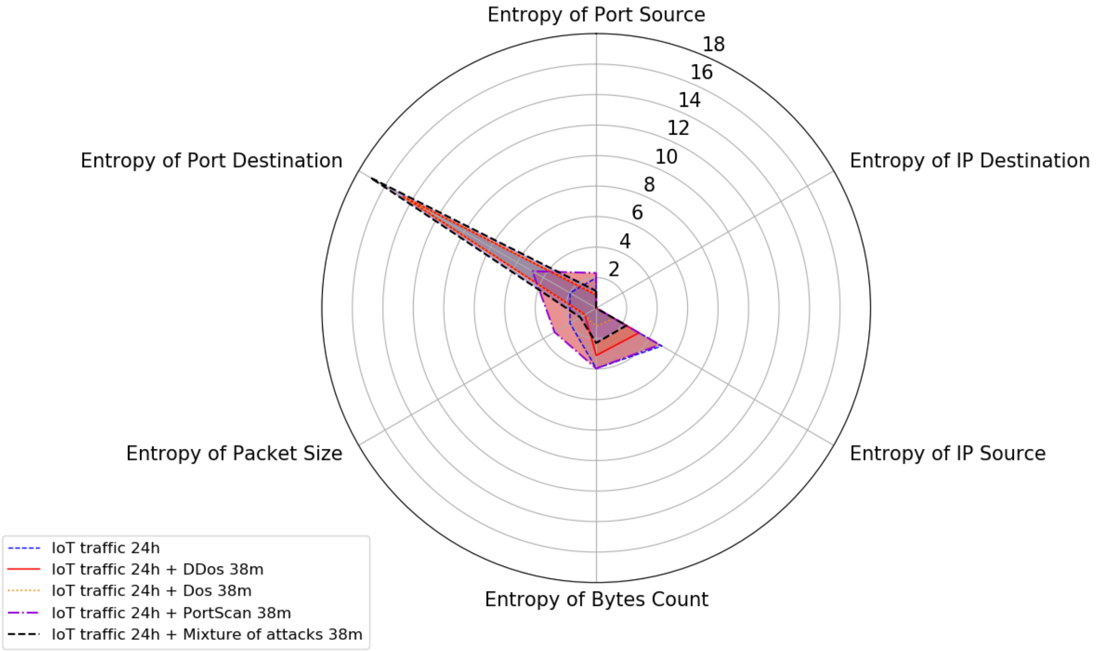


Figure 3.16: Behavior Shape: Bio-Medical and Anomalies Traffic

for the smart home environment, 16 for the bio-medical and that the DDoS is generated by 5 bots, while DoS and Port Scanning are generated by a single bot, then there are more distinct IP sources in the legitimate traffic leading to more variations and higher entropy. However, this parameter allows us to observe variation in the traffic and detect malicious traffic from a legitimate one. For source port, the observation is similar as for IP Source. DoS and DDoS rely on a single port to send traffic and then traffic entropy reaches the lowest value at 0.11 for smart-home and 0.9 for bio-medical. IoT scenarios involve more source ports and show higher entropy. One can also observe that the impact is more pronounced for smart home; indeed, there are fewer packets than with bio-medical environments and anomalous traffic such as DoS or DDoS count for a larger part of the total traffic.

Regarding the packet size parameter, Port Scanning reaches a higher value than other traffic and DoS and DDoS traffic show the lowest value. This is due to the fact that for DoS and DDoS traffic send only 60 Bytes packets while Port Scanning uses various sizes of packets (100 B to 1 MB, Section III.D Table 3.5) and IoT traffic uses various sizes of packets according to devices and environment as shown in Table 3.1 and Table 3.3. Thus, there are more variations for Port Scanning and higher entropy.

Similarly, the bytes count computes the total bytes by IP source and the entropy depends on the diversity of packet size and the number of IP sources. That is why the DoS traffic shows lower entropy.

Besides, in order to evaluate the effectiveness of my generator under the influence of anomalous traffic, I also use BS graphs. From Figure 3.17, we can observe that anomalous traffic directly impacts the shape of the traffic. This impact is equivalent to both synthetic and measured traffic. As I analyzed above, the malicious traffic shapes are still different from legitimate traffic. With different attack cases, the shape of anomalous synthetic traffic nearly coincides with the anomalous measured traffic. These observations are similar when we consider the traffic of each device with ON/OFF activity under various network conditions as shown in Figure 3.18, 3.19, 3.20, 3.21, 3.22, 3.23, 3.24, 3.25, 3.26, 3.27, 3.28, 3.29, 3.30, 3.31, 3.32, 3.33, 3.34, 3.35, 3.36, and 3.37.

It shows that with IoTTGen, we can still model IoT traffic successfully under various network conditions.

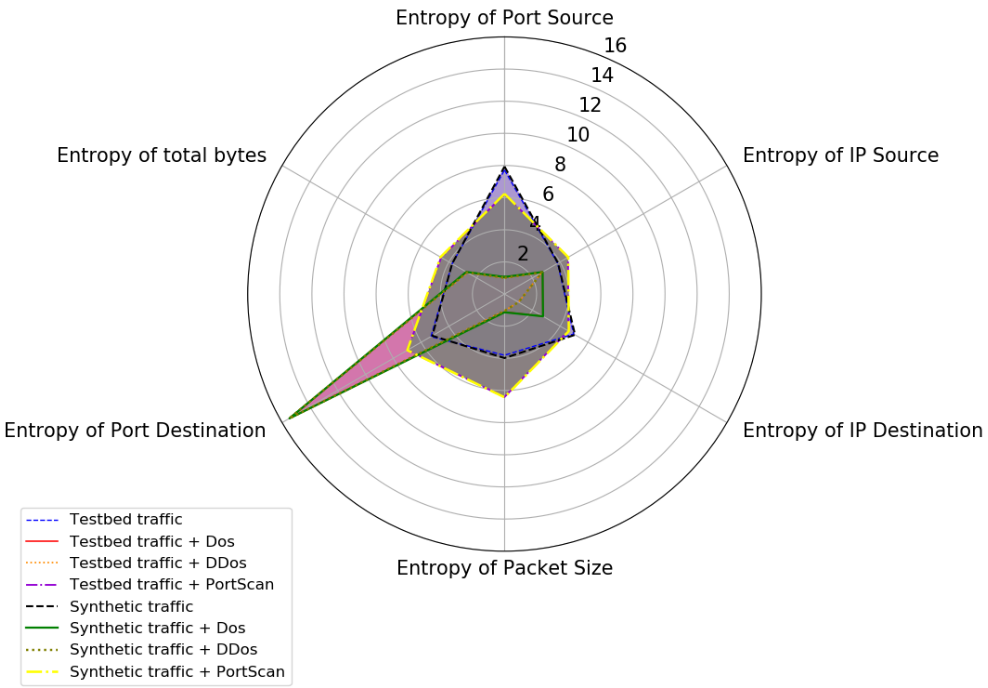


Figure 3.17: Behavior Shape: Synthetic Traffic with measured parameters, Measured Traffic and Anomalous Traffic

These experiments in two distinct IoT environments have been performed in order to compute the level of entropy with regards to traffic parameters and to visually observe the traffic on BS graphs. I observed clearly that different traffic such as legitimate or malicious traffic show different entropy values and have different impacts on the network. By using my IoT Generator, I succeed in picturing the characteristics of different IoT traffic, and I show that it is possible to detect anomalies based on entropy and visual representation of the traffic such as Behavior Shape.

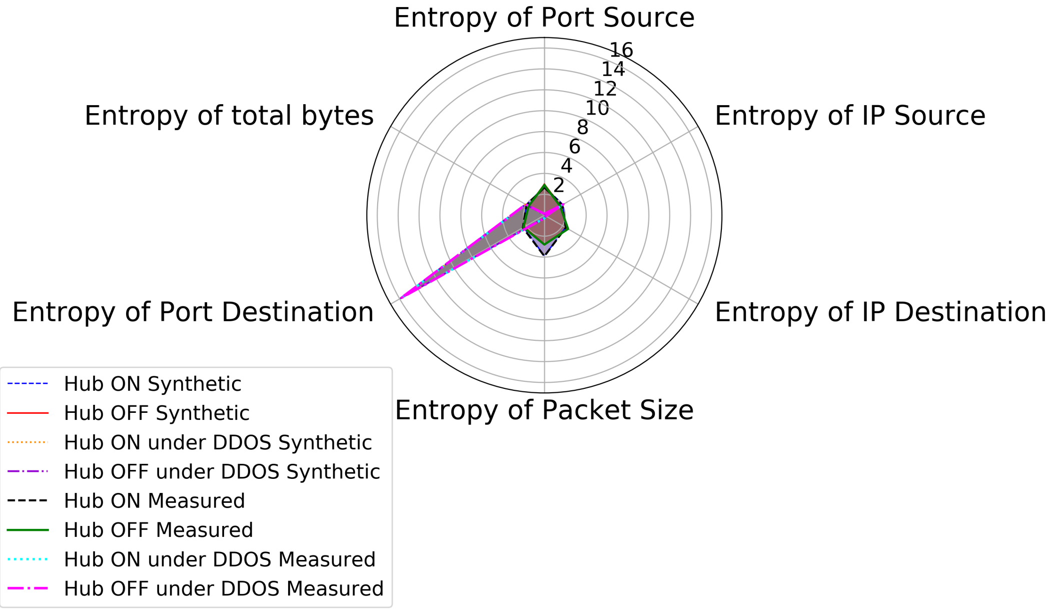


Figure 3.18: Behavior Shape of Hub traffic under DDOS: Synthetic traffic, Measured traffic and Anomalies traffic

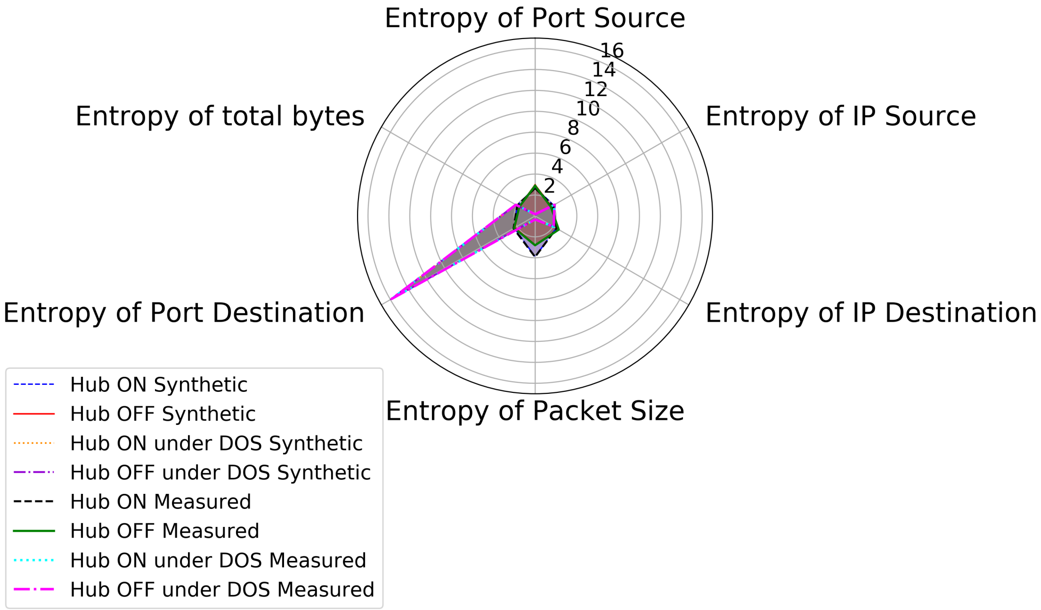


Figure 3.19: Behavior Shape of Hub traffic under DOS: Synthetic traffic, Measured traffic and Anomalies traffic

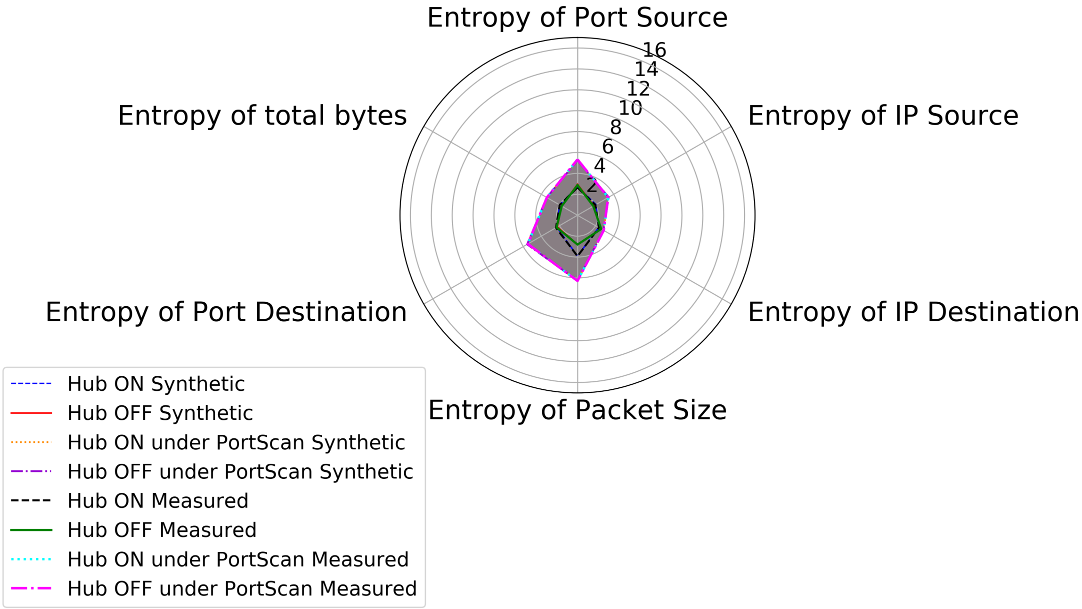


Figure 3.20: Behavior Shape of Hub traffic under PortScanning: Synthetic traffic, Measured traffic and Anomalies traffic

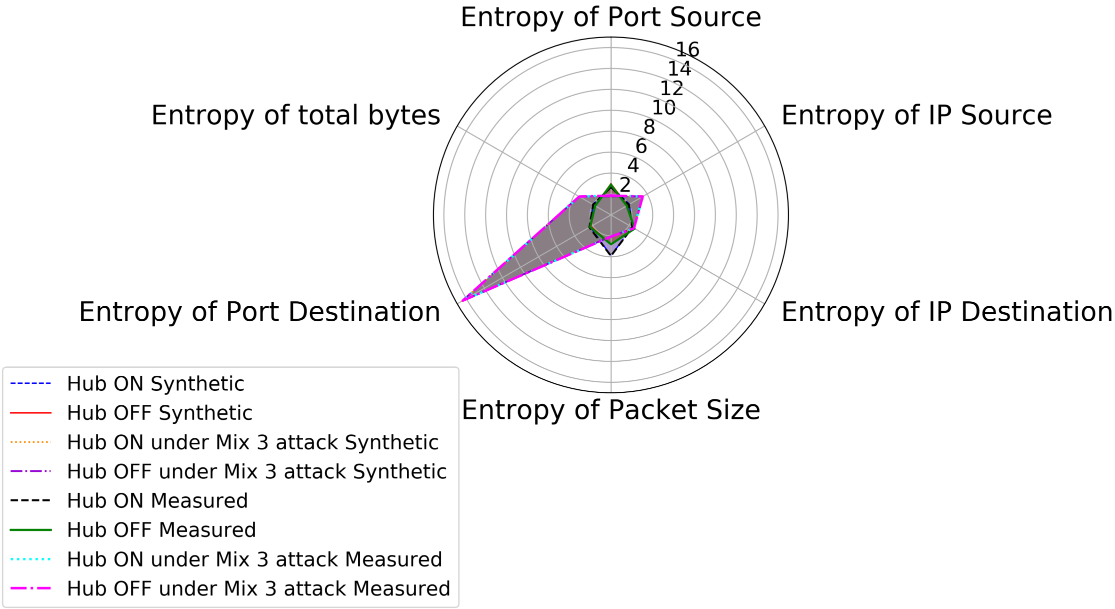


Figure 3.21: Behavior Shape of Hub traffic under 3-attack: Synthetic traffic, Measured traffic and Anomalies traffic

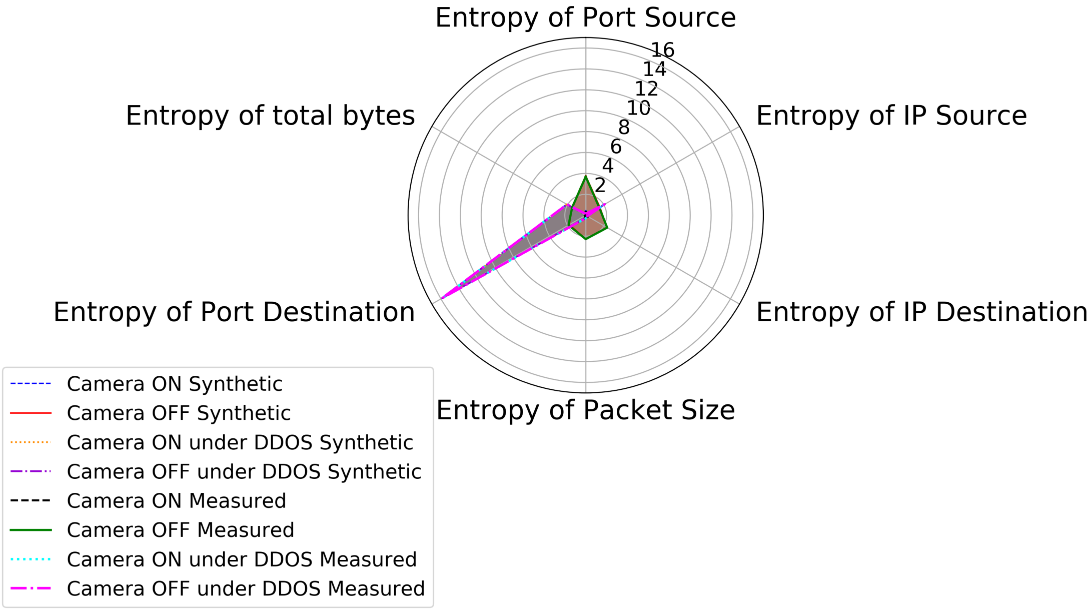


Figure 3.22: Behavior Shape of Camera traffic under DDOS: Synthetic traffic, Measured traffic and Anomalies traffic

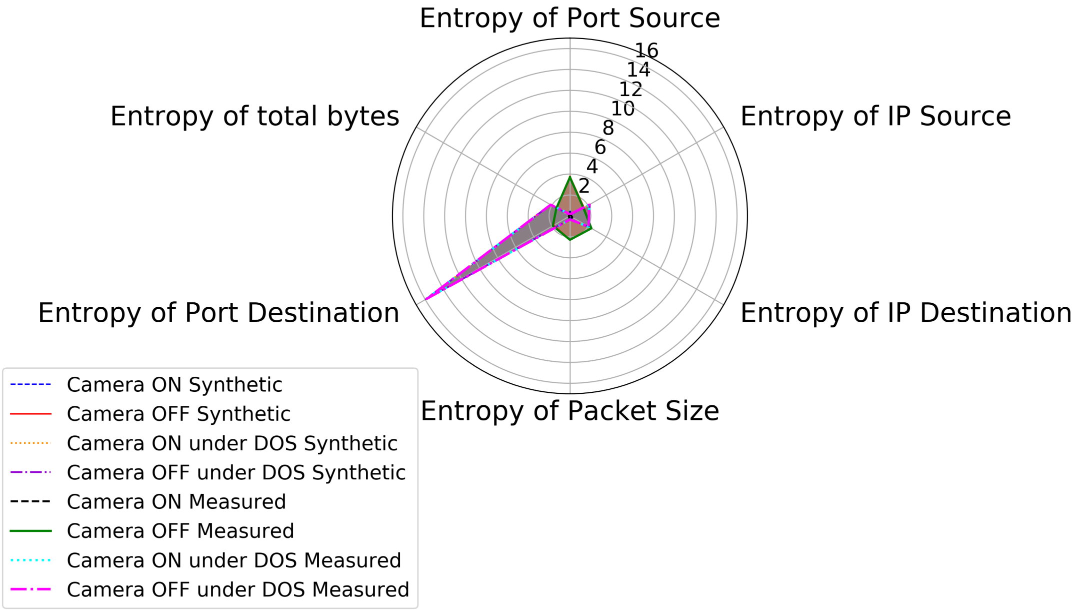


Figure 3.23: Behavior Shape of Camera traffic under DOS: Synthetic traffic, Measured traffic and Anomalies traffic

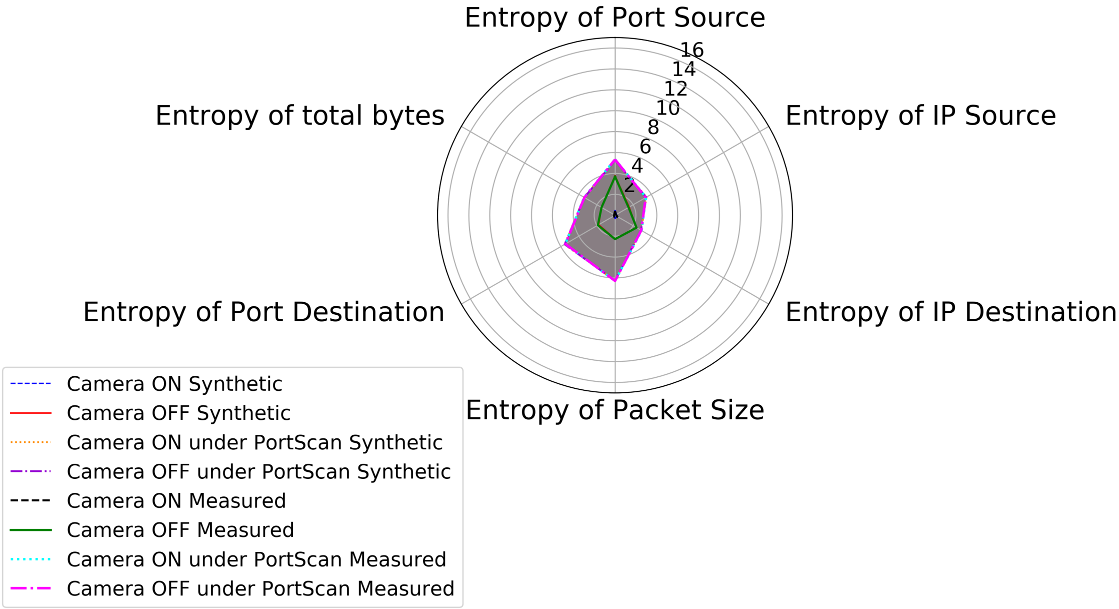


Figure 3.24: Behavior Shape of Camera traffic under PortScanning: Synthetic traffic, Measured traffic and Anomalies traffic

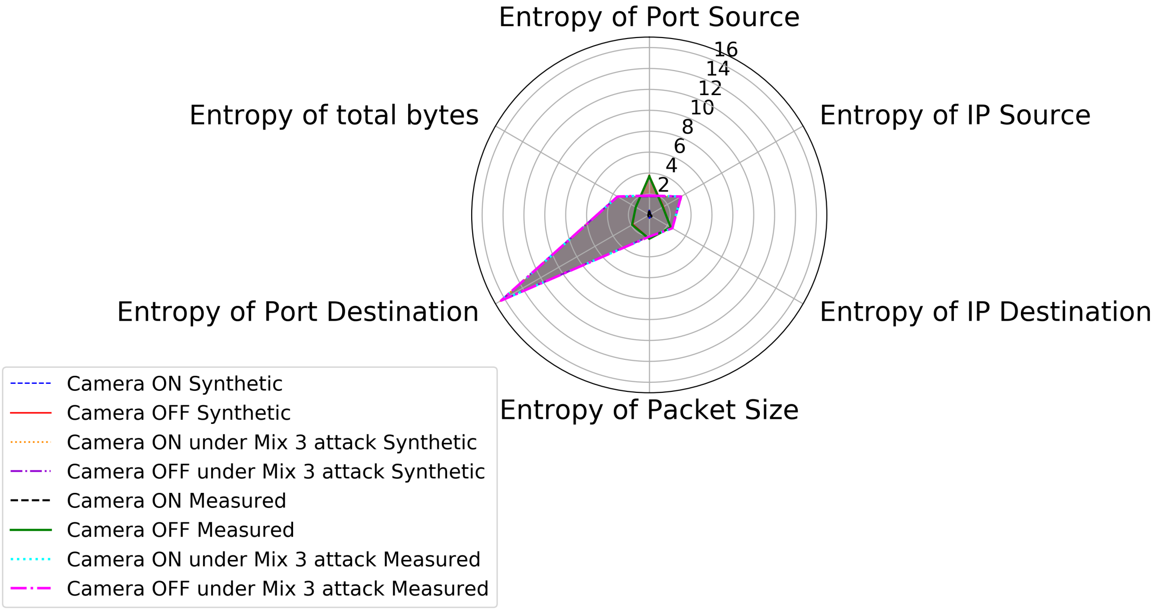


Figure 3.25: Behavior Shape of Camera traffic under 3-attack: Synthetic traffic, Measured traffic and Anomalies traffic

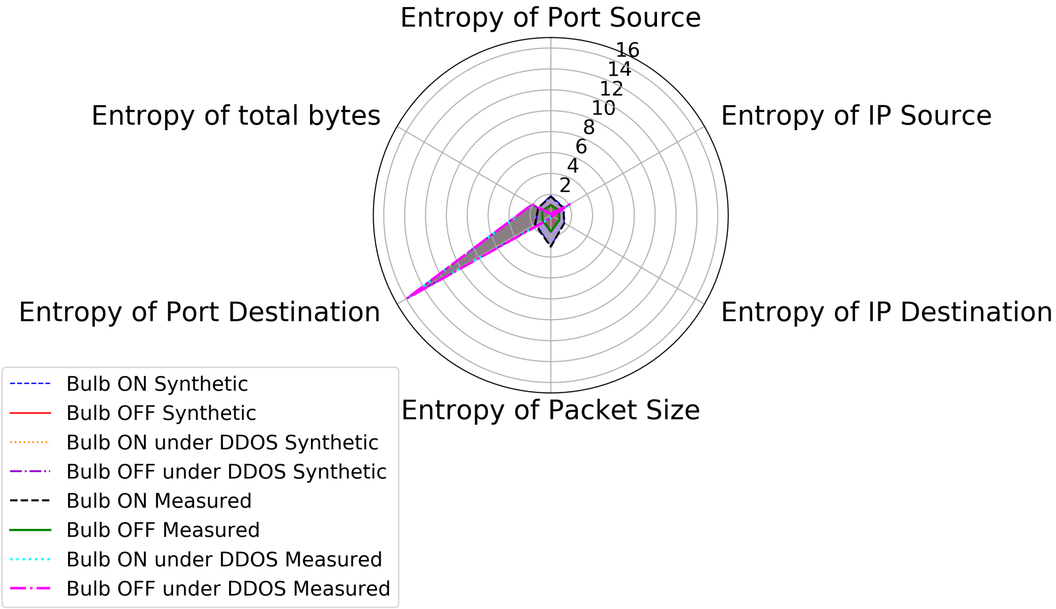


Figure 3.26: Behavior Shape of Bulb traffic under DDOS: Synthetic traffic, Measured traffic and Anomalies traffic

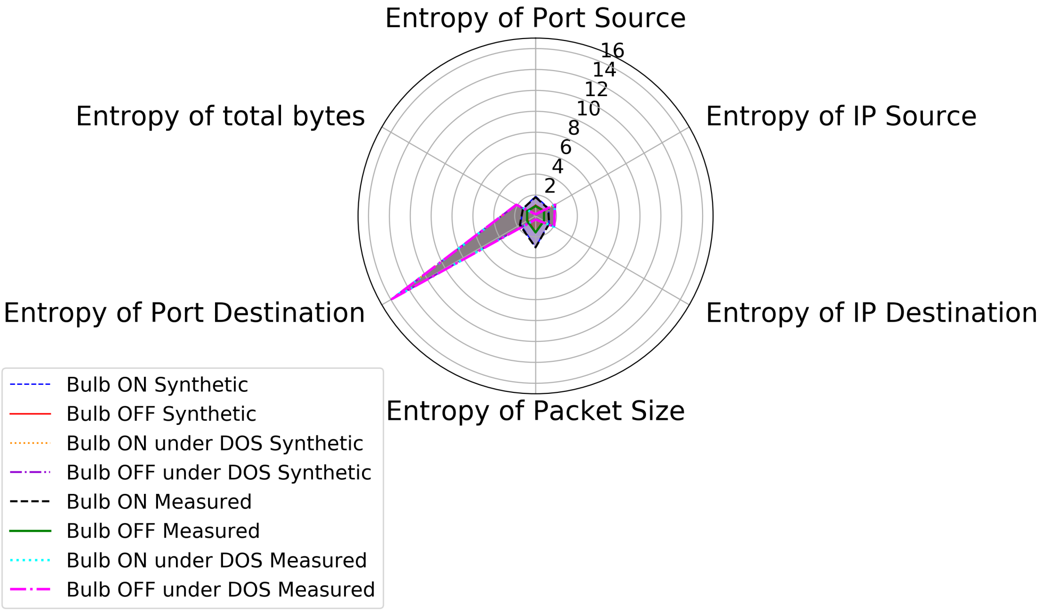


Figure 3.27: Behavior Shape of Bulb traffic under DOS: Synthetic traffic, Measured traffic and Anomalies traffic

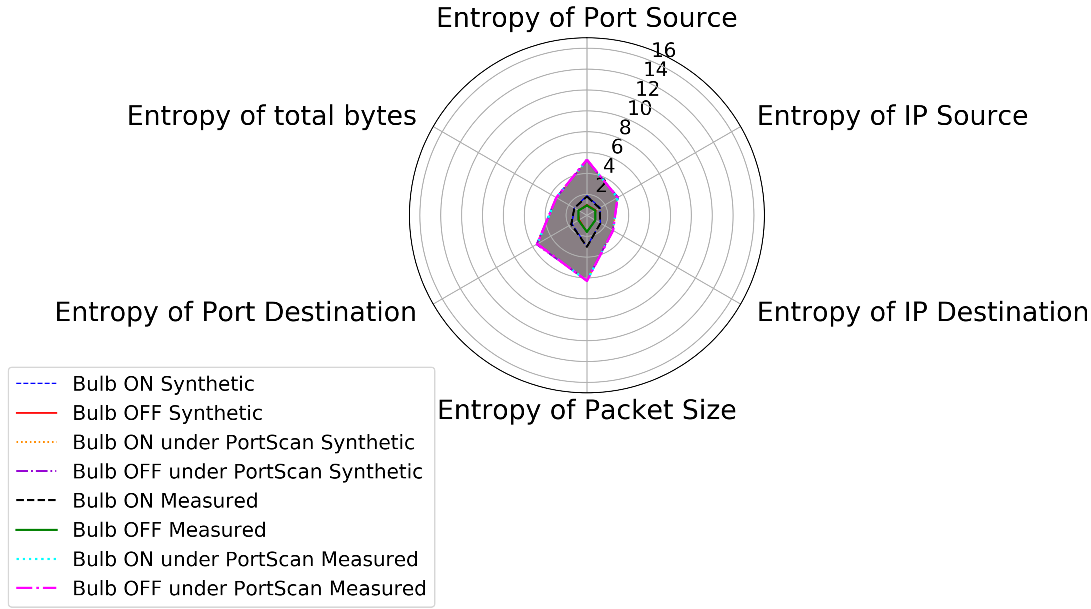


Figure 3.28: Behavior Shape of Bulb traffic under PortScanning: Synthetic traffic, Measured traffic and Anomalies traffic

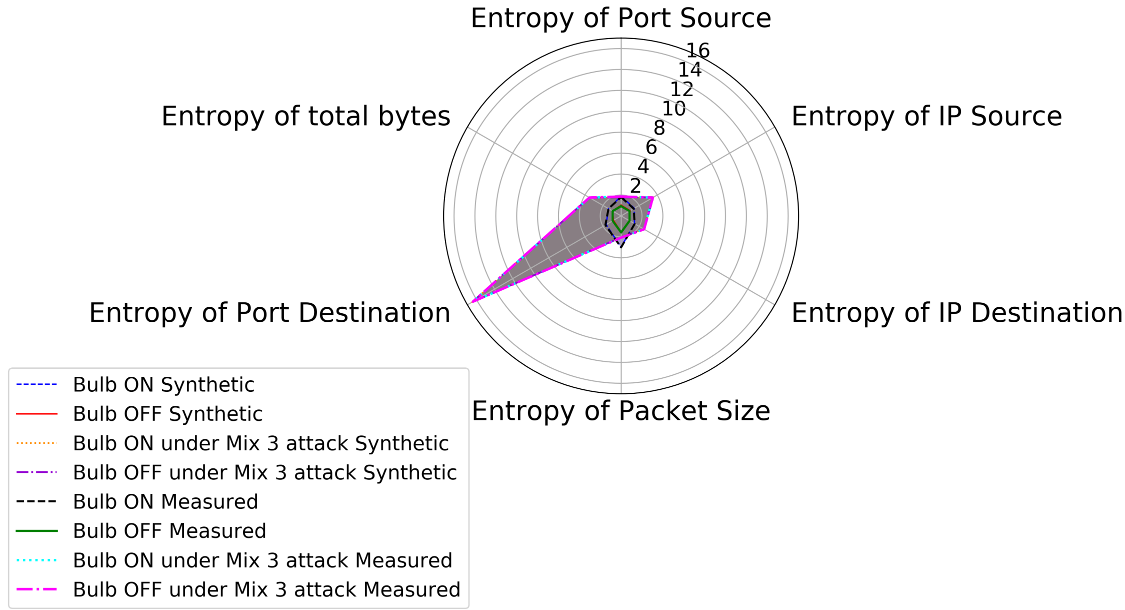


Figure 3.29: Behavior Shape of Bulb traffic under 3-attack: Synthetic traffic, Measured traffic and Anomalies traffic

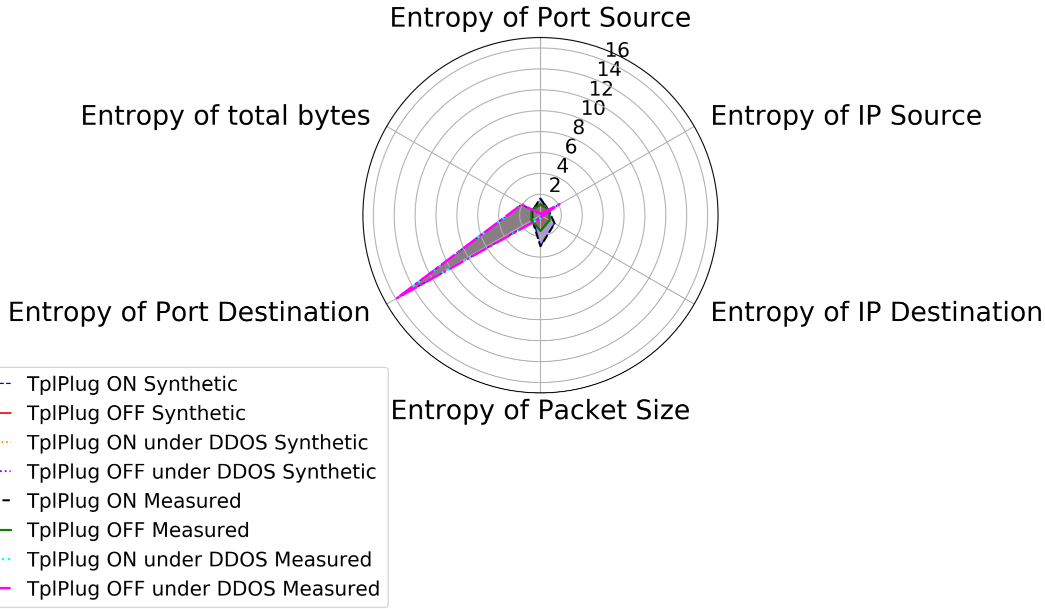


Figure 3.30: Behavior Shape of TplPlug traffic under DDOS: Synthetic traffic, Measured traffic and Anomalies traffic

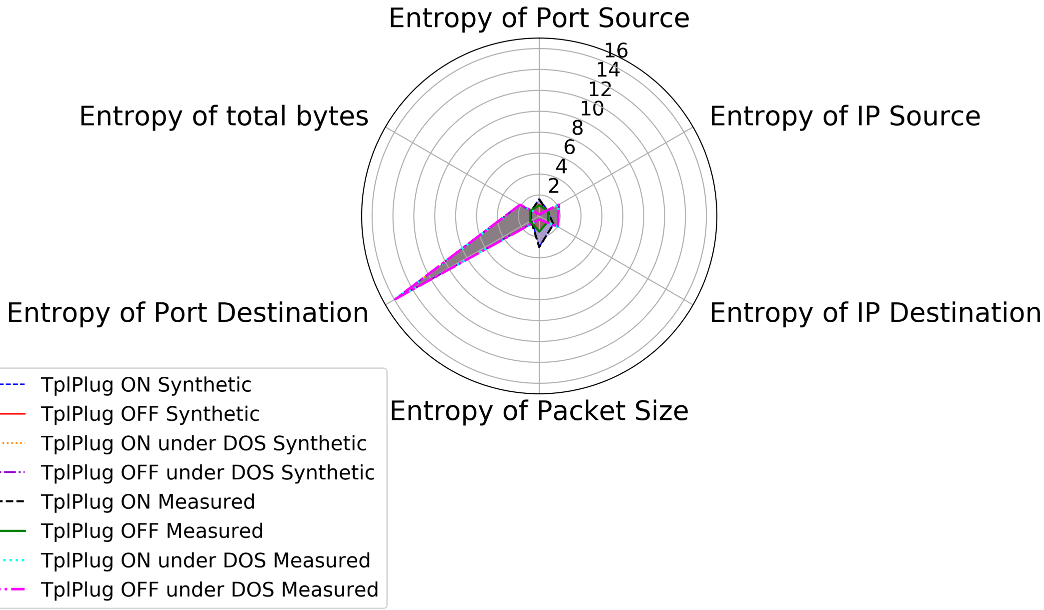


Figure 3.31: Behavior Shape of TplPlug traffic under DOS: Synthetic traffic, Measured traffic and Anomalies traffic

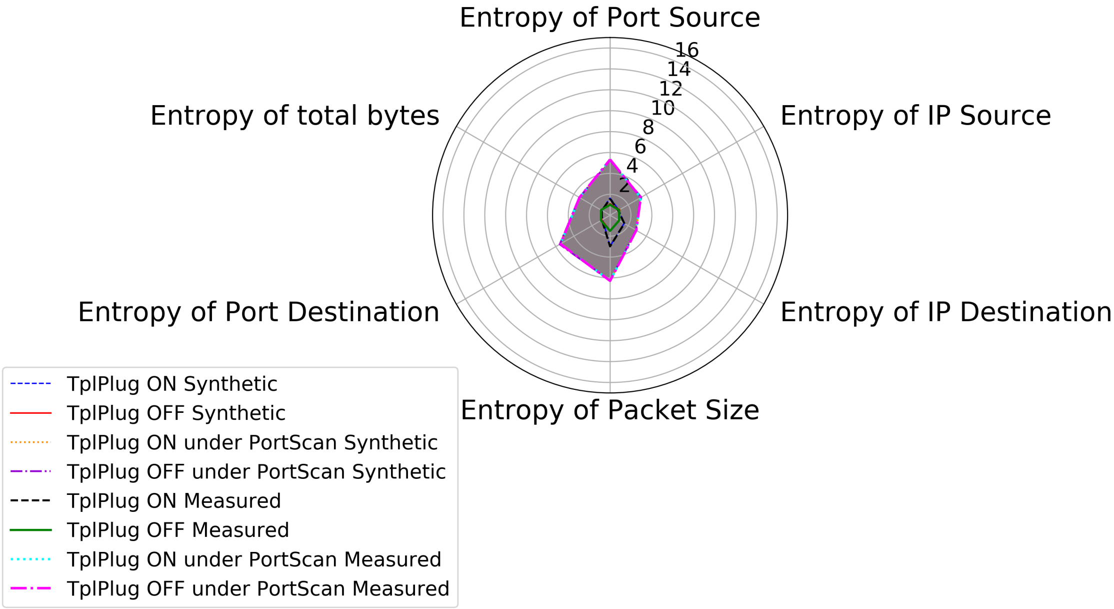


Figure 3.32: Behavior Shape of TplPlug traffic under PortScanning: Synthetic traffic, Measured traffic and Anomalies traffic

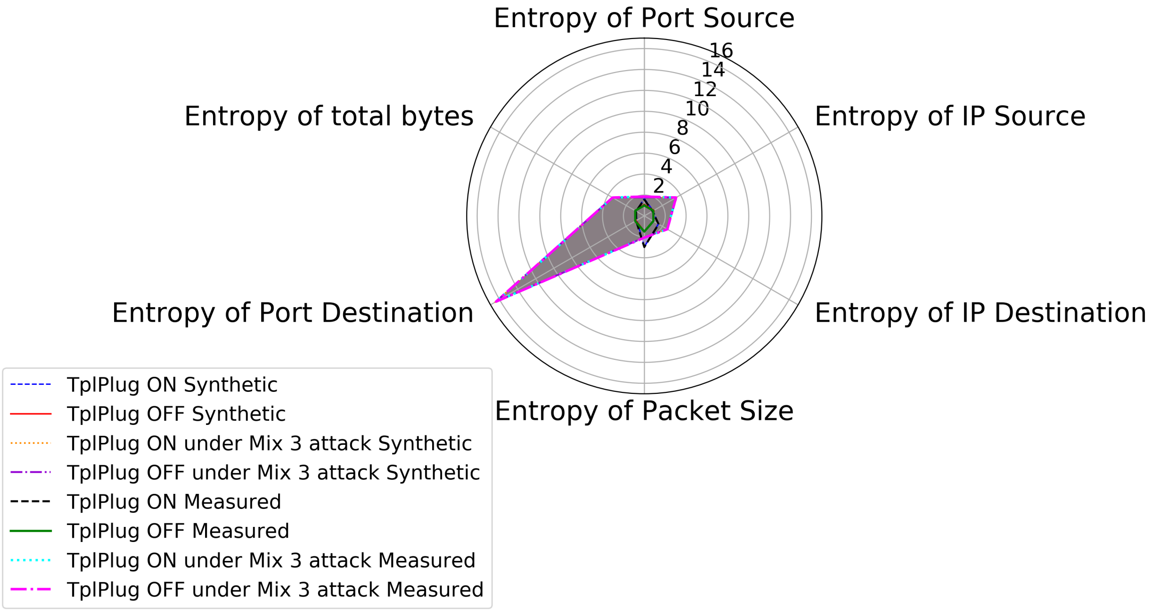


Figure 3.33: Behavior Shape of TplPlug traffic under 3-attack: Synthetic traffic, Measured traffic and Anomalies traffic

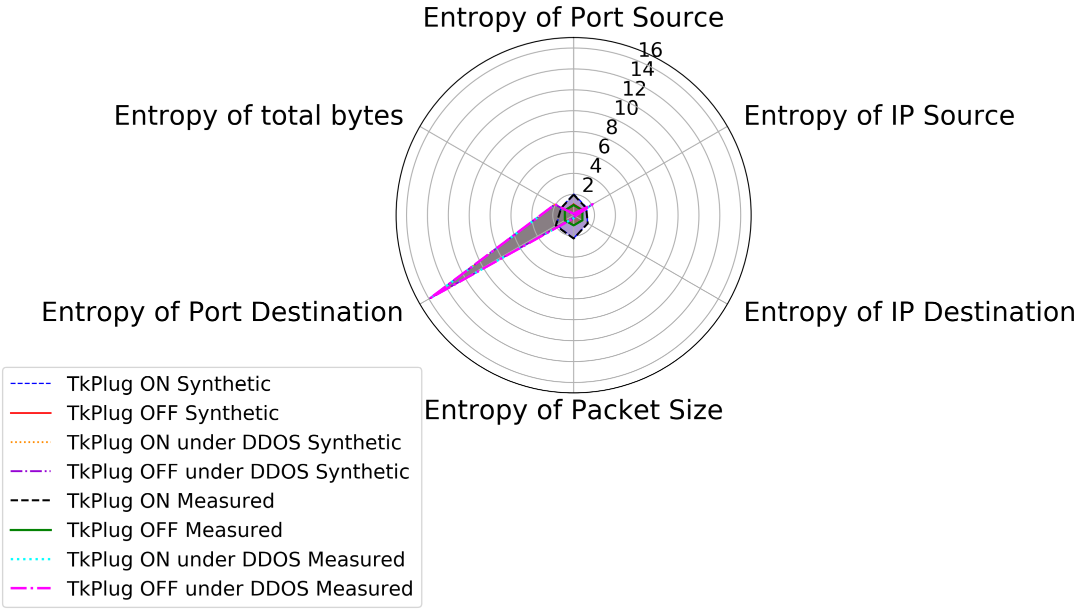


Figure 3.34: Behavior Shape of TkPlug traffic under DDOS: Synthetic traffic, Measured traffic and Anomalies traffic

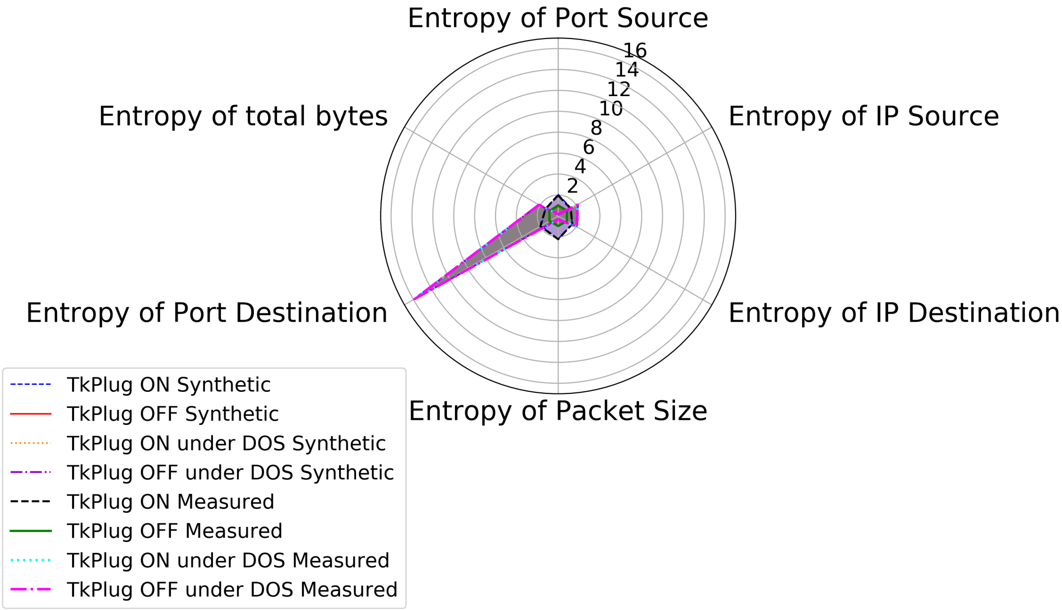


Figure 3.35: Behavior Shape of TkPlug traffic under DOS: Synthetic traffic, Measured traffic and Anomalies traffic

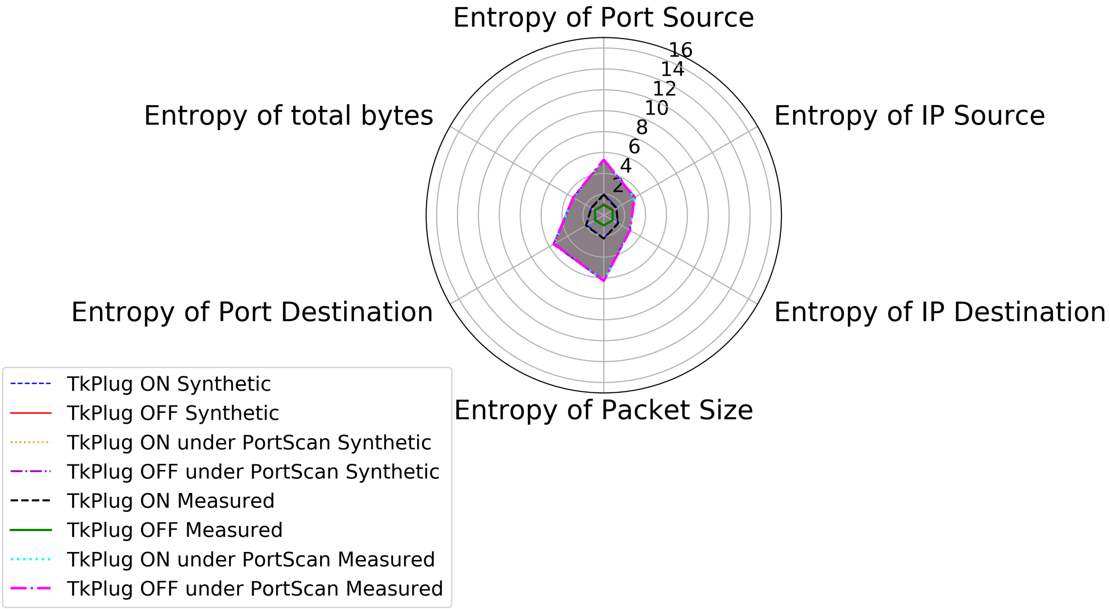


Figure 3.36: Behavior Shape of TkPlug traffic under PortScanning: Synthetic traffic, Measured traffic and Anomalies traffic

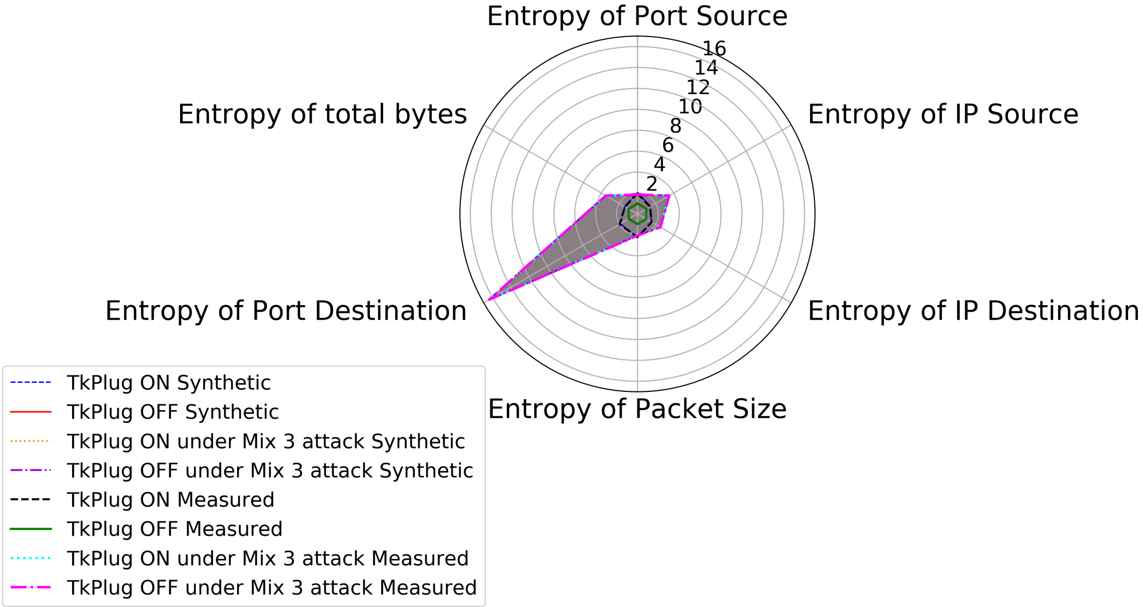


Figure 3.37: Behavior Shape of TkPlug traffic under 3-attack: Synthetic traffic, Measured traffic and Anomalies traffic

3.5 Conclusion

In this chapter, I presented IoTTGen, my new IoT traffic generator tool. I use IoTTGen to perform experiments and especially for two use cases: smart home and bio-medical environments. IoT traffic has been generated and IoT anomalies extracted from a public dataset have also been included. Traffic parameters entropy has been computed and observed through Behavior Shape graphs.

My results show that the shape of the traffic differs significantly for different IoT environments. Anomalous traffic also has an important impact on the traffic shape. My traffic generator succeeds in representing the characteristics of the traffic and my methodology shows that I can also compare traffic and highlight anomalies.

For future work, I are now using my generator to model different IoT environments and characterize different IoT traffic. For instance, I adapt my generator tool for new scenarios such as smart agriculture or smart factory. My generator is also tuned to generate IoT network anomalies and combine to measurement traffic. It is an essential tool for studying consumer IoT devices in a wide range of scenarios. Regarding my entropy-based identification method, I'm now adapting to classify IoT devices on the fly. This method could be used in the gateway to detect immediately legitimate or malicious devices in the network environment. I are also including my generator into a framework for detecting IoT network anomalies.

Chapter 4

Entropy-based IoT Devices Identification

4.1 Introduction

The Internet of Things (IoT) is the worldwide deployment of connected devices sensing the physical world and sharing the collected information through the Internet. There has been a wide range of novel IoT applications providing accurate information into cyberspace for industry, home, or healthcare. This technology is now part of everyday life and end-users can interact with IoT devices through their smartphones, tablets, or computers. Thus, Cisco has predicted that more than 500 billion devices will be connected to the Internet by 2030 [37], hence having an impact both on the economic growth, as well as a technological impact on the Internet.

As it is expected that the Internet of Things will count for a major part of the Internet traffic, IoT devices are still mostly studied regarding their hardware capabilities [38] or operating systems [39], and only a few studies focus on IoT devices as communication entities with the aim of characterizing the IoT traffic. IoT traffic should however exhibit different characteristics than current consumer Internet traffic because of the multiplicity of sources, the heterogeneity of hardware devices, and also novel services leading to new traffic patterns [17].

The IoT traffic could be seen as the aggregation of packets generated by several devices from different environments such as smart home or smart cities. These environments involve several sensors that are dedicated to specific tasks such as monitoring systems or collecting cyber-physical values (temperature, humidity, etc.). Thus, compared with Internet traffic where traffic has some human-centric features (flash-crowd, popularity, etc.), the IoT traffic and sensor devices continuously perform the same operations and generate the same amount of data periodically.

Besides, the rise of IoT has also unveiled new vulnerabilities as observed in 2016 with the Mirai DDoS Botnet, which had a severe impact on the Internet [40]. IoT is, therefore, facing new challenges regarding the cybersecurity of devices and the privacy of data and communication. Indeed, the collected data can convey critical information about users, their privacy, and the environment. In this context, it is essential to characterize the traffic of IoT devices in order to prevent security threats and mitigate vulnerabilities.

In this paper, I proposed a new method to identify IoT devices. My method is based on the traffic entropy computed for each device and Machine Learning (ML) algorithms to classify devices. My method succeeds in identifying devices under various network conditions with performance up to over 94% in all cases. My method is also robust to unpredictable network behavior with anomalies spreading into the network.

The remainder of this paper is organized as follows. Section 5.2 introduces my testbed, the data collection, and the methodology for identifying IoT traffic. Section 5.3 discusses on the dataset and the traffic entropy. Section 5.4 emphasizes the results of my IoT devices classification method. Section 5.5 summarizes the paper.

4.2 IoT identification method

I propose a new approach to identify and classify IoT devices based on machine learning and traffic entropy value. Before presenting my methodology to identify devices, I first describe my experiment testbed and the IoT device data traffic collection.

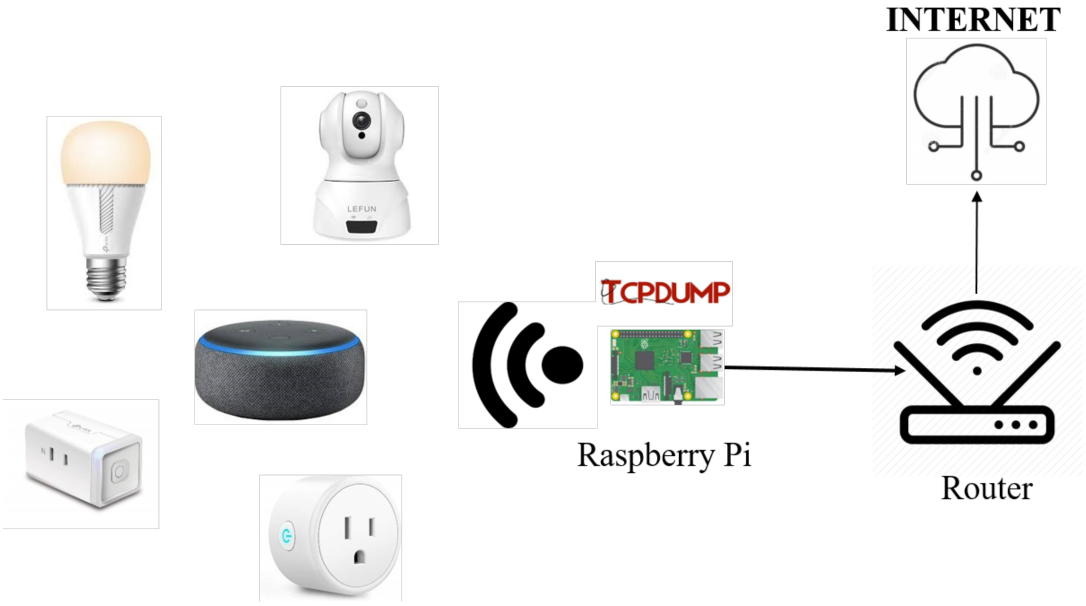


Figure 4.1: Smart-home Testbed

4.2.1 Testbed

In order to collect IoT traffic, I set up a Smart Home experiment testbed, which is presented in Fig. 4.1. My testbed is composed of five on-market popular IoT devices for Smart-Home: an Amazon Echo dot as a Smart Hub, a Lefun Indoor Security Camera, a TP-Link Kasa Wi-Fi Smart Bulb, a TP-Link Wi-Fi Smart Plug, and a Teckin Wi-Fi Smart Plug. These devices are connected to the Internet with Wi-Fi through a home gateway, which can control the flow information among smart appliances to the remote network.

A Raspberry Pi 3 Model B is configured as a wireless access point and serves as the gateway to the public Internet and for collecting the traffic. The Raspbian Jessie OS was used for the Raspberry Pi and additional software applications were also installed such as DNSMasq for DNS and DHCP services, Hostapd for the access point, and authentication server services, and Tcpdump for collecting the traffic. All the traffic from IoT devices was recorded and stored into a single trace with the pcap file format. I collected the IoT traffic for several days and I will present results for one-day traffic as other days show similar traffic properties. Fig. 4.2 shows the one-day traffic on my testbed.

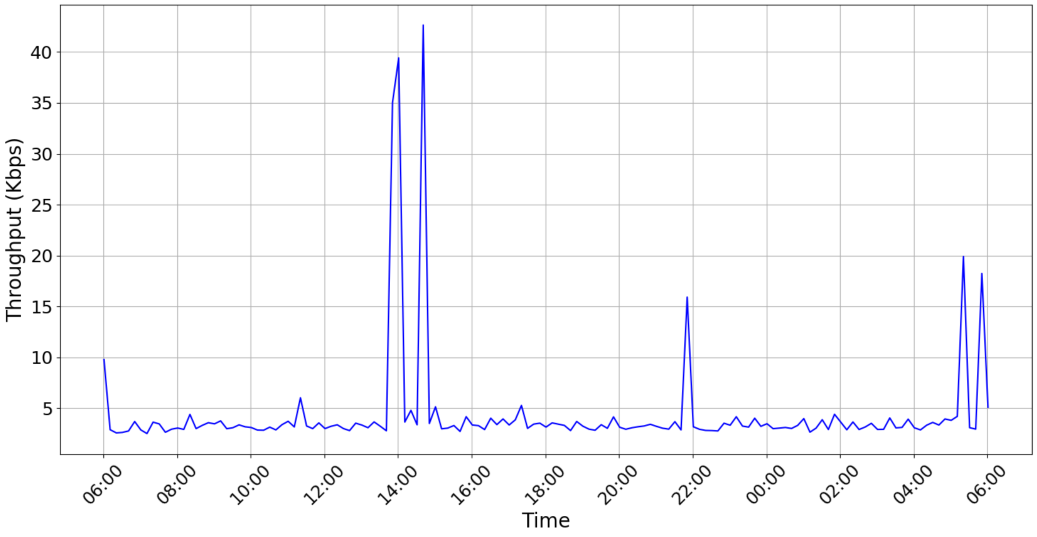


Figure 4.2: Testbed IoT traffic

4.2.2 Scenarios

4.2.2.1 ON/OFF Activity

The IoT devices are dedicated entities responsible for sensing or interacting with the physical world, e.g., activating a bulb light or switching off a plug. The vast majority of use cases in the IoT environment are the periodic transmission of messages containing sensor measurements, status, or simple commands. For instance, in the case of smart light, it sends periodically its on/off status. On the traffic collection day, I performed some activities using the IoT devices from 13:50 to 15:00 and I can observe in Fig. 4.2 that the traffic has significantly increased at this period of time. I also had some other activities at almost 22:00 and 05:30. I will refer to these periods of activity as ON periods, while the others are referred to as OFF periods. The total traffic reaches barely 3.2 Kbps without any activity (OFF) while it can reach 16 Kbps or up to 40 Kbps while there are some activities using devices (ON).

4.2.2.2 Anomalous Traffic

Besides the regular users' activity described previously as ON or OFF period, another scenario of interest is when the IoT devices are under attacks and there are some security threats in the network. Indeed, there are more and more IoT devices connected to the Internet and there have been a lot of cybersecurity

Table 4.1: IoT anomaly traffic traces (1 minute duration)

	# of Packets	Packet Size	Volume
DDOS	773,045	60B	44 MB
DOS	727,210	60B	41 MB
Port Scanning	23,587	100 B - 1 KB	22 MB

threats or anomalies as seen with Mirai botnet, etc. Thus, it is also essential to detect the devices accurately when under attack or with anomalous traffic. Then the operator will be able to react quickly in case of new threats.

To this end, I rely on a public IoT Traffic dataset [20] and I extracted the traffic of three cybersecurity threats: a) Port Scanning, b) Denial of Service (DoS), and c) Distributed Denial of Service (DDoS). The anomalous traffic traces are presented in Table 4.1. The total duration of each trace is one-minute-long. Then each trace is injected into my collected IoT traffic trace. I then obtain five different traces: the original traffic depicted in Fig. 4.2, three traces with including a single anomaly (DDoS, DoS, and Port Scanning) and one trace with all the anomalies.

4.2.3 Entropy

As I aim to identify IoT devices based on network traffic (legitimate or anomalous), the frequencies of traffic parameters such as IP addresses or ports can help identify the devices and the network conditions. Hence, I compute the entropy values of the following traffic parameters [41]: IP Source, IP Destination, Port Source, Port Destination, Packet Size, and Bytes.

Information Entropy is a quantity in information theory used to measure the uncertainty [42] (4.1).

$$H(X) = \sum_{i=1}^{n} p(x_i) log(p(x_i)) \tag{4.1}$$

The value of entropy can vary from 0 to log(n): a 0-value means that the observations (i.e., packets) are similar, whereas higher entropy value shows that observations are different. In the rest of the paper, I will see that these parameters convey sufficient information to identify the devices under different network conditions precisely.

As the network activity can vary during day time, intense activity period (ON) or no activity (OFF), as well as under several cybersecurity threats (DDoS, DoS, Port Scanning), I split the one-day traffic traces into five minutes-long duration traffic traces (288 five-minute traces for a 24-hours day trace). In order to classify the five-minute traces into the ON or OFF period or anomalous, I rely on the k-means clustering method combined with the mean silhouette value to optimize the number of clusters [43]. Thus, each five-minute traces can be classified as active, inactive, or under cybersecurity attacks.

4.3 IoT traffic observation

4.3.1 Traffic Traces

Table 4.2 summarizes the one day trace for my collected traffic. The Camera which is a high bandwidth demand device generated the major part of packets and Bytes, while Plugs count for a small amount of the traffic. The Hub also generated a lot of packets compared with the Camera but the overall Hub traffic count for a lower amount of data.

Table 4.2: IoT traffic traces (1 day)

	# of Packets	Packet Size	Volume
Hub	82,251	82.5 B	10.9 MB
Camera	110,776	306 B	33.5 MB
Bulb	6,707	80 B	563 KB
TplPlug	1,849	85 B	150 KB
TkPlug	7,198	85 B	583 KB

4.3.2 Cloud Servers

There is a large number of IoT manufacturers on the market today. Each manufacturer has its own cloud server to manage its devices and each DNS queryresponse pair is mapped into a particular domain owned by manufacturers, as shown in Table 4.3. For instance, the TP-Link plug device will be directed to devs.tplinkcloud.com. For Internet users, they may access many online servers (i.e., Web, OSN, e-Business) during their activities. Differently, the IoT devices are dedicated to a single task and communicate only to a pre-established servers. By inspecting the remote servers that IoT devices are connecting to, this can indicate whether the device has been corrupted or whether it may send information to non-legitimate servers.

Table 4.3: Cloud servers and DNS queries for IoT devices

Devices	Cloud servers	DNS queries
Hub	Amazon CloudFront	d3p8zr0ffa9t17.cloudfront. net
Camera	Mipc	s0.mipcgw.com
Bulb	TP-Link cloud	devs.tplinkcloud.com
Tpl Plug	TP-Link cloud	devs.tplinkcloud.com
Tk Plug	Tuya cloud	a3.tuyaus.com

4.3.3 Traffic Entropy Behavior Shape

By deploying an IoT testbed, my main objective is to identify IoT devices based on traffic entropy. For traffic visualization, I then plot the Behavior Shape (BS) graphs [26] of the entropy value of the traffic features for each IoT device. As I split the traffic into five minutes-long time slots, each trace from the same state (i.e., ON, OFF, or anomalous) shows similar properties and I show the BS of a single five-minute trace for each device.

4.3.3.1 IoT Devices Traffic

Fig. 4.3 shows the BS of one-day traffic for five distinct days. As the traffic shapes are consistent over a day, a single day traffic is representative of other days.

Fig. 4.4, 4.5, 4.6, 4.7, and 4.8 present BS for different smart devices during one day. I can observe that the area of the OFF period is smaller than of ON period, except for the Camera. It can be explained that in the ON period, the Camera uses almost only one type of packet while the remaining devices use more diverse types. This leads to a decrease in the entropy value expressed by a smaller BS area for the traffic of Camera ON.

Moreover, the BS for each device is very different and this BS can help to identify the IoT devices. This can be explained by the fact that each device has different characteristics and functionality and they are sending or receiving different kinds of the packet. For example, the Camera uses many different source

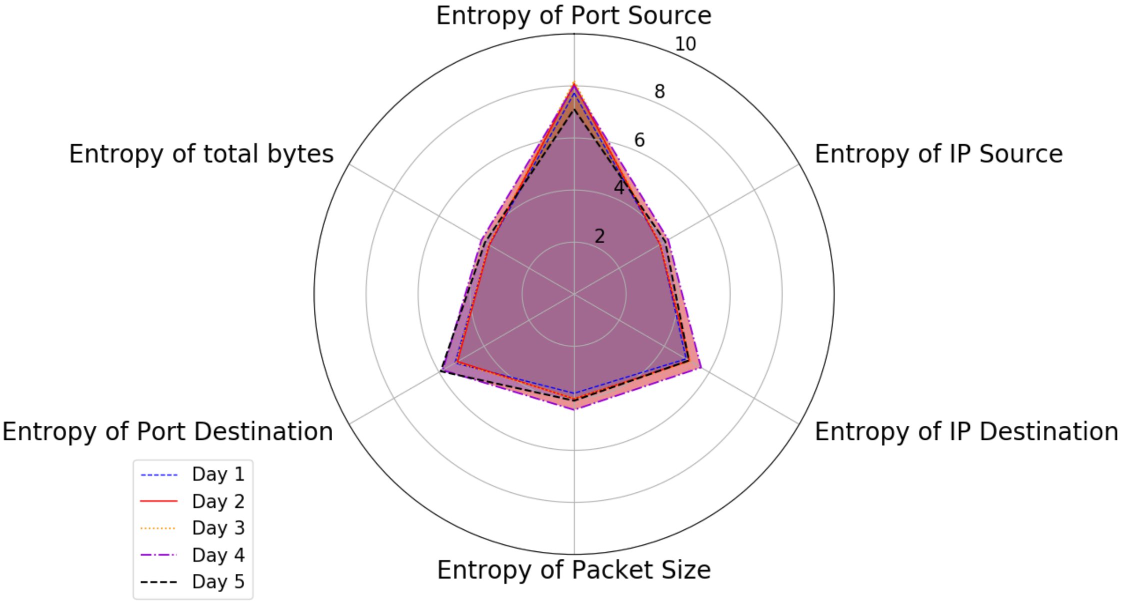


Figure 4.3: Behavior Shape of IoT traffic in five consecutive days

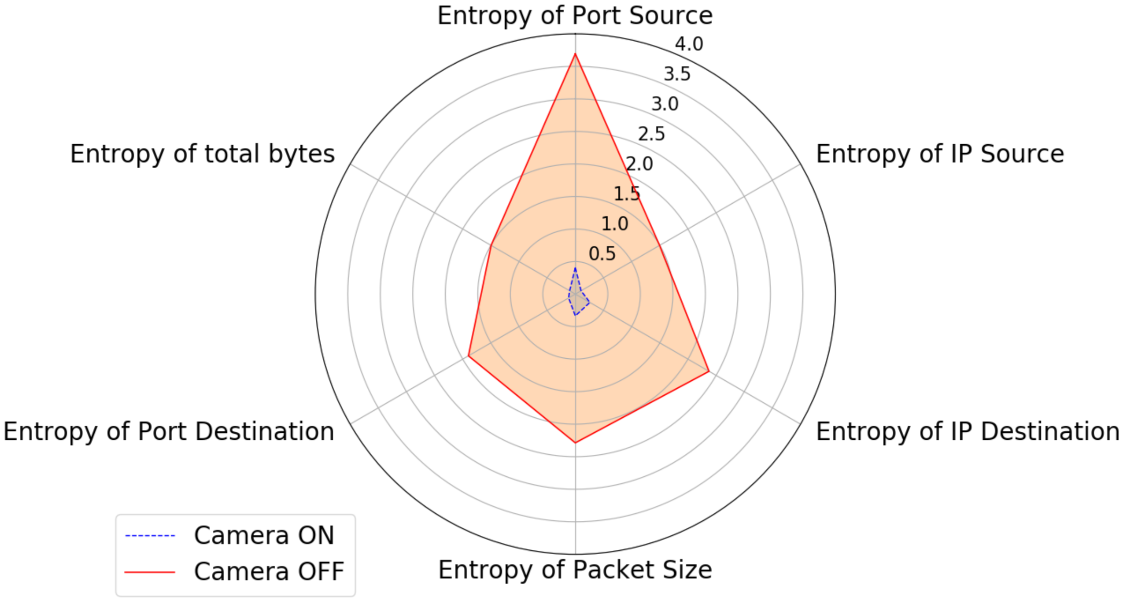


Figure 4.4: Behavior Shape of Camera Traffic

ports when sending packets, and shows a greater entropy value for Port Source (3.7) than for Hub (2.9), or other devices (i.e., TplPlug 1.55, TkPlug 2.0 and Bulb 1.9).

In addition, regarding the OFF period, the BS for Camera and Hub are larger than other devices. Those devices have many functionalities and generate many

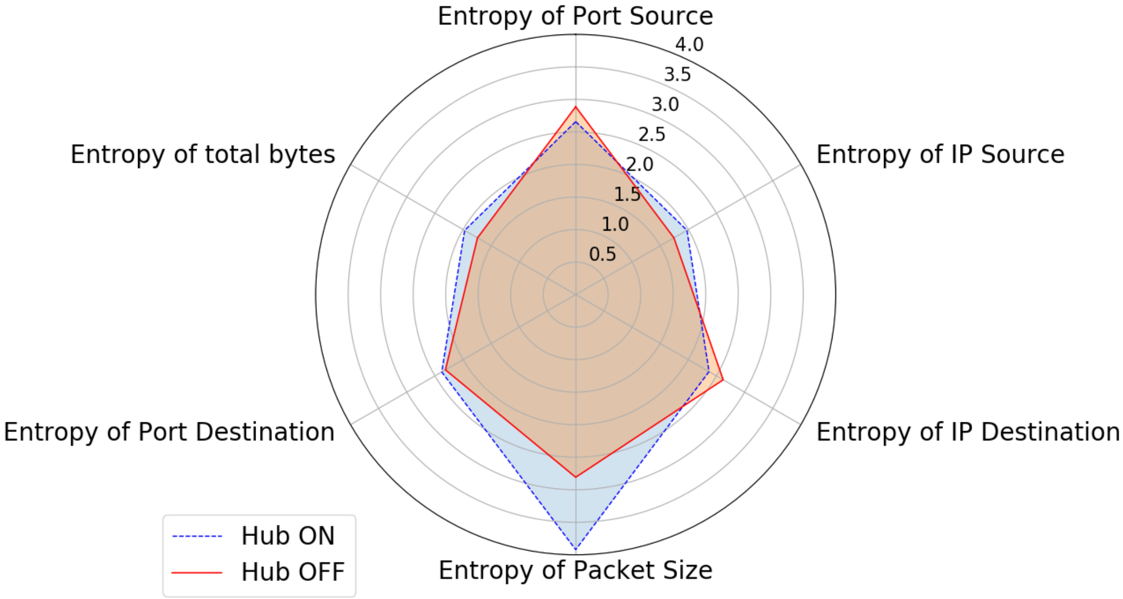


Figure 4.5: Behavior Shape of Hub traffic

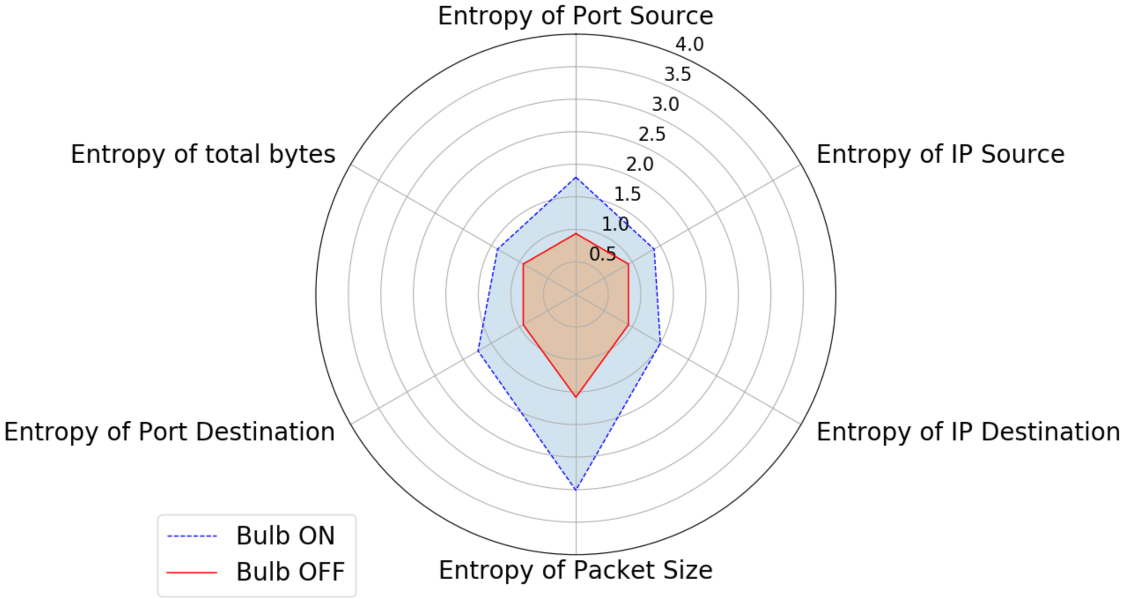


Figure 4.6: Behavior Shape of Bulb traffic

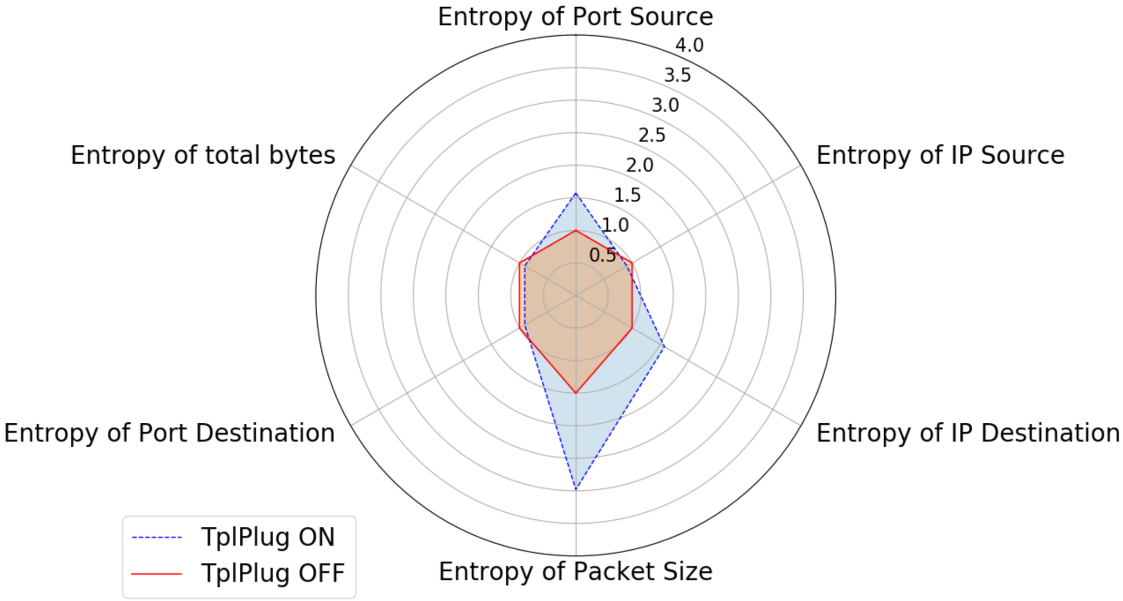


Figure 4.7: Behavior Shape of TplPlug traffic

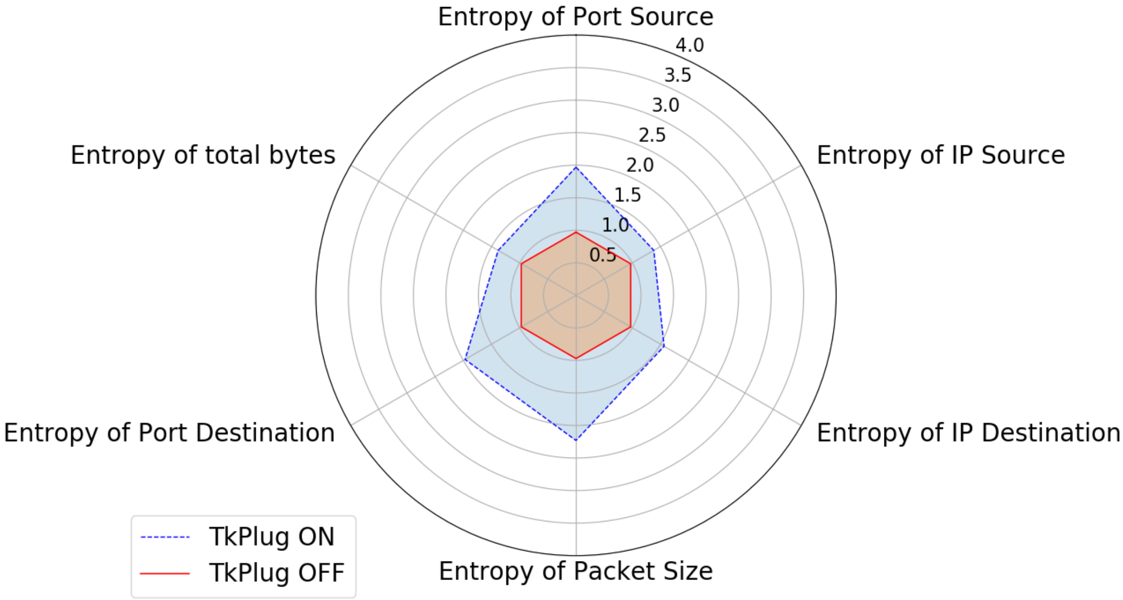


Figure 4.8: Behavior Shape of TkPlug traffic

more packets even in the OFF period compared to other devices (Bulb or Plugs). Indeed, the entropy values for such devices reach a higher level.

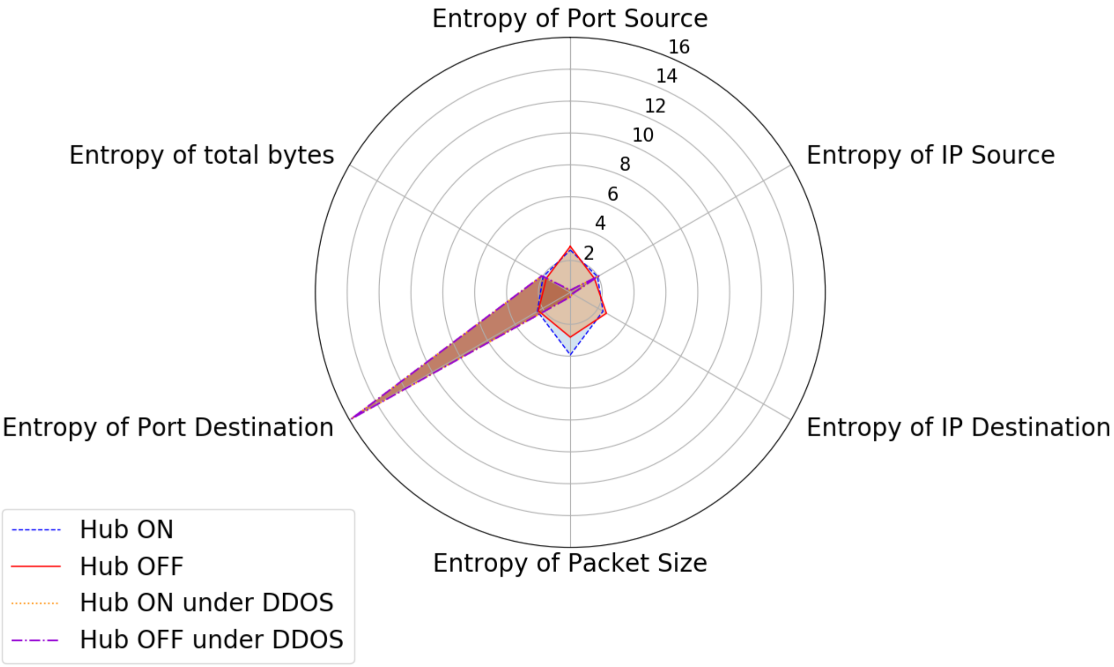


Figure 4.9: Behavior Shape of Hub traffic with DDOS

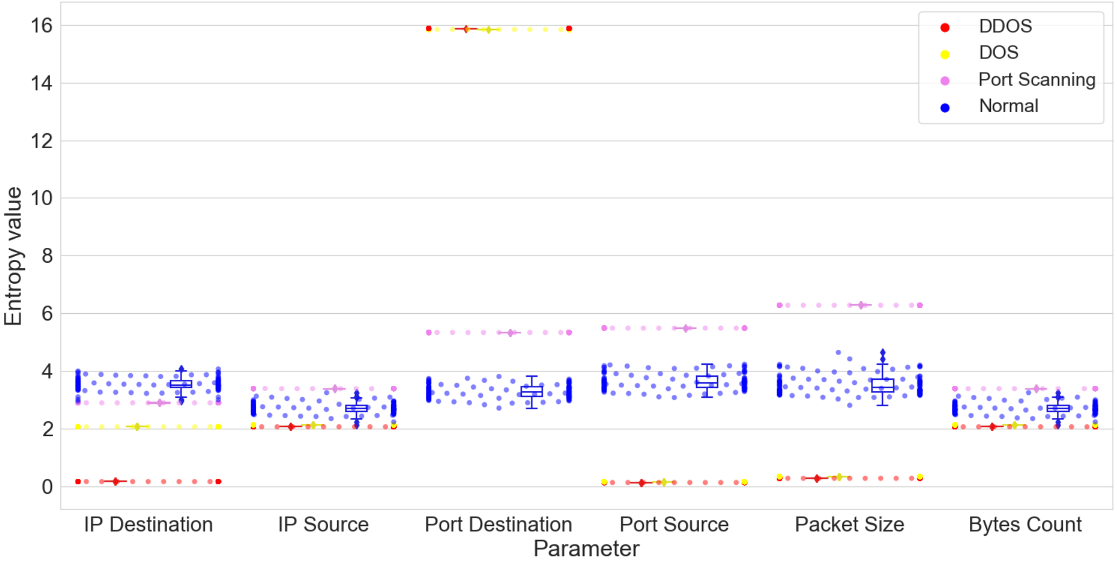


Figure 4.10: Entropy for normal and anomalous traffic

4.3.3.2 Anomalous Traffic

When the network is under attack, the number of packets increased drastically. Therefore, anomalous traffic has a strong impact on the shape of the IoT traffic. From Fig. 4.9, I can observe that the shape of Hub traffic with DDOS is totally different from the Hub traffic. DDOS exhibits much higher entropy values for

Port Destination (16) because it targets a large number of destination ports. For Packet Size and Port Source, the Hub traffic has higher entropy than DDOS traffic. Indeed, DDOS relies on a single port to send the packets with the same size, so its traffic entropy for this feature is close to zero. Overall, the Behavior Shape is able to represent the nature of the traffic.

Besides, I also compare the entropy values of all device traffic with each anomalous traffic. Fig. 4.10 shows that the difference between the range of entropy values of anomalous traffic and normal traffic. The entropy values of normal traffic always fluctuate within a certain range while these values of all anomalous traffic are almost kept stable. Additionally, the minimum or maximum value of normal traffic is always greater or less than of anomalous traffic. From the observation of the entropy value, it is possible to detect anomalies for IoT devices.

4.4 Classification and Evaluation

4.4.1 Classification Algorithms

I observed before that the entropy values of traffic features show different characteristics and can help to identify devices.

I now aim at classifying the devices by relying on Machine Learning algorithms. Remind that my collected one day trace was split into several five minutes-long traces. I evaluate the effectiveness of my classification by using a 10-fold cross-validation method [44] and then apply it to an independent validation dataset. Dataset is randomly divided into two datasets: training dataset (80% of total instances, i.e., five minutes-long traces) and validation dataset (20% of total instances).

I first rely on six classification algorithms : (i) Decision Tree (DT), (ii) Random Forest (RF), (iii) K-Nearest Neighbors (KNN), (iv) Gaussian Naive Bayes (NB), (v) Neural Network (NN) and (vi) Support Vector Machines (SVM) through the Weka software [45]. For evaluating the performance of the algorithms, I consider the following metrics such as True Positive Rate (TPR), False Positive Rate (FPR), Precision, Recall, and F-Measure. Table 4.4 shows the classification results of these algorithms on the validation dataset for all devices and network conditions.

From my experiments, SVM and NN algorithms perform poorly with the TPR metric to only 0.2996 and 0.3137, respectively. NB and DT show better performances but NB only reaches an average level of performances (0.5712), while DT reaches a higher level and can classify properly about 72% of the IoT devices. RF and KNN algorithms outperform other algorithms and show a high level of performances: KNN succeeds in classifying more than 92% of the devices, and RF exhibits even higher performances up to 94.74%. As RF and KNN succeed at classifying IoT devices based on the entropy value, RF shows better performances among all, and I will rely on the Random Forest algorithm for computing the Classification Matrix in the following.

Table 4.4: Performances of ML algorithms (validation dataset)

	RF	KNN	DT	NB	NN	SVM
TPR	0.9474	0.9277	0.7179	0.5712	0.3137	0.2996
FPR	0.0013	0.0022	0.0110	0.0159	0.0283	0.0291
Precision	0.9512	0.9356	0.7271	0.5767	0.3188	0.3008
Recall	0.9474	0.9277	0.7179	0.5712	0.3137	0.2996
F-Measure	0.9465	0.9267	0.7171	0.5583	0.3813	0.2835

4.4.2 Confusion Matrix for IoT Identification

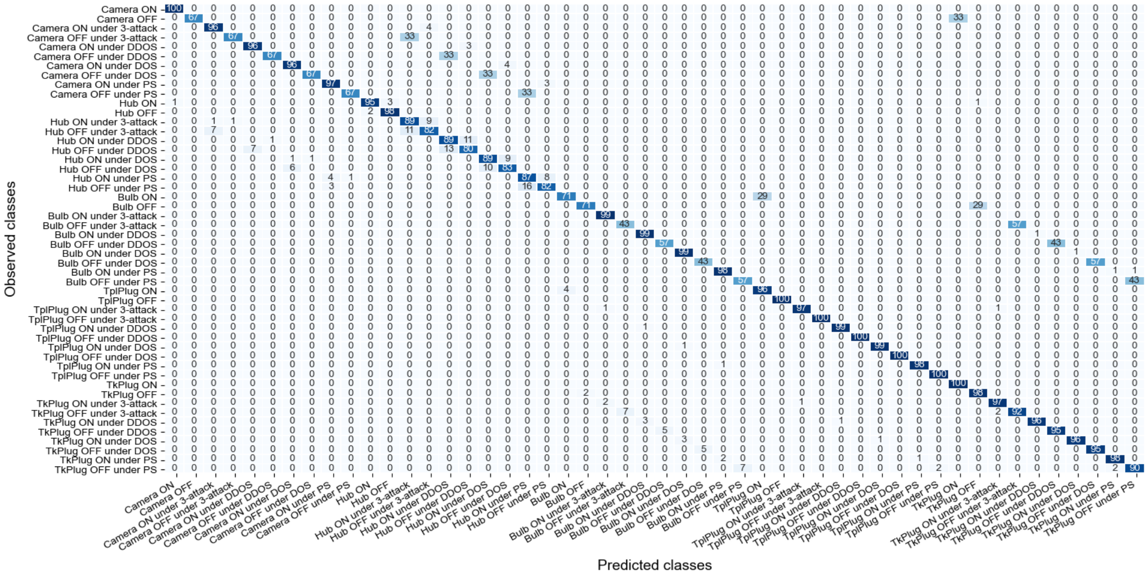


Figure 4.11: Confusion Matrix RF classification algorithm

Through all my data, I can define fifty classes based on the states of each device and the type of network traffic (normal or anomalous). More precisely, the five devices can be into an ON or OFF states and under five different network conditions: regular traffic, DDoS, DoS, Port Scanning, and the three attacks jointly.

After processing the data with RF, I obtain a probability vector for each class that will be shown on the Confusion Matrix (CM) in Fig. 4.11. The accuracy of the classification depends on the ratio of accurate predictions. The CM provides further information into not only the accuracy of different classifiers but also which classes are correctly or incorrectly predicted and the type of misclassification.

For all devices, my classification method reaches a very high level of accuracy for detecting the devices under different network conditions. For the Plugs, 96% of the traffic for this device is accurately classified. The precision for Hub under the regular network condition is also very high (95%) and still over 80% under anomalous network conditions. The prediction for Camera OFF also reaches a high level (96%), but the prediction accuracy decreases under network anomalies. Similarly, for the light bulb, the prediction is very high for the ON state (98-99%) but drops drastically with the OFF period (43-57%). Basically, the predictions are very accurate with regular traffic and are dropping while the network is compromised by cyberattacks. The intense activity period (ON) also

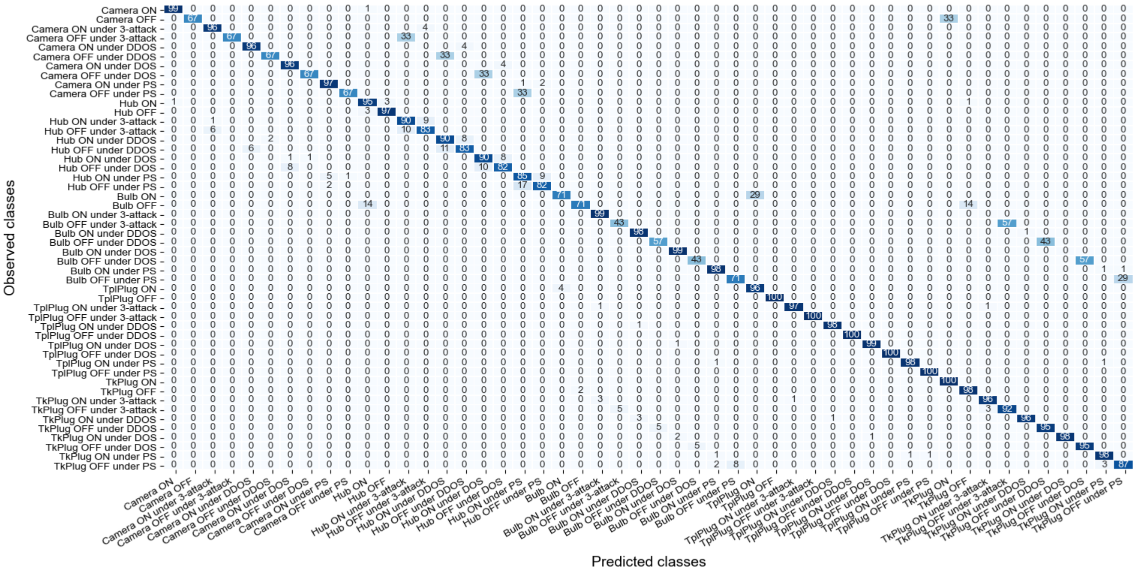


Figure 4.12: Confusion Matrix RF classification algorithm - Synthetic traffic

shows higher accuracy than OFF with no user activity. Besides, I also classify IoT traffic based on synthetic traffic from IoTTGen and measured parameters from Table 3.7. Fig. 4.12 shows the confusion matrix of synthetic traffic. I observe that these accuracy are almost similar to of classification of measured traffic.

Finally, the classification of IoT Devices based on entropy succeeds in identifying IoT devices and also under various network conditions.

4.5 Conclusion

In this paper, I present my new method to classify IoT devices. My method is based on the computation of the entropy values of several traffic features. Machine learning algorithms such as Random Forest are then used to classify the devices based on the entropy value of the IoT traffic.

My results show that I can reach a high level of performance for classification, especially in the case of intense activity (94% accuracy). I also train my method under different scenarios and network attacks and it is still able to classify the devices accurately.

Chapter 5

Conclusion and Future Work

This chapter concludes the dissertation and figured out directions for future work.

5.1 Conclusion

In this dissertation, I presented IoTTGen, my new IoT traffic generator tool. I use IoTTGen to perform experiments and especially for an use cases: smart home environments. IoT traffic has been generated and IoT anomalies extracted from a public dataset have also been included. Traffic parameters entropy has been computed and observed through Behavior Shape graphs. My results show that the shape of the traffic differs significantly for different scenarios and each type of IoT devices. Anomalous traffic also has an important impact on the traffic shape. My traffic generator succeeds in representing the characteristics of the traffic and my methodology shows that I can also compare traffic and highlight anomalies.

Besides, I also present my new method to classify IoT devices. My method is based on the computation of the entropy values of several traffic features. Machine learning algorithms such as Random Forest are then used to classify the devices based on the entropy value of the IoT traffic.

My results show that I can reach a high level of performance for classification, especially in the case of intense activity (94% accuracy). I also train my method under different scenarios and network attacks and it is still able to classify the devices accurately.

5.2 Future work

For future work, I'm now using my generator to model different IoT environments and characterize different IoT traffic. I'm also including my generator into a framework for detecting IoT network anomalies. Regarding to IoT devices identificatiion, I'm adapting my method in order to collect and classify the IoT devices on the fly.

Bibliography

[1] S. Avallone, S. Guadagno, D. Emma, A. Pescape, and G. Ventre, ”D-ITG distributed Internet traffic generator”, First International Conference on the Quantitative Evaluation of Systems, 2004.

[2] M. Jemec, “PackETH, Open Source Ethernet Packet Generator,” Available: http://packeth.sourceforge.net/

[3] R. Olsson, “pktgen the linux packet generator,” in Proceedings of the Linux Symposium, Ottawa, Canada, 2005.

[4] A. Tirumala, J. Ferguson, “Iperf 1.2 – The TCP/UDP Bandwidth Measurement Tool,” http://dast.nlanr.net/Projects/Iperf/, May 2001.

[5] B. R. Patil, M. Moharir, P. K. Mohanty, G. Shobha and S. Sajeev, ”Ostinato - A Powerful Traffic Generator,” 2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS), Bangalore, 2017, pp. 1-5

[6] ZTI Telecom, ”IP Traffic - test & measure,” http://www.zti-telecom.com.

[7] H. Nguyen-An, T. Silverston, T. Yamazaki, T. Miyoshi, ”Generating IoT traffic: A Case Study on Anomaly Detection,” IEEE International Symposium on Local and Metropolitan Area Networks, pp. 1-6, 2020.

[8] N. Apthorpe, D. Reisman, and N. Feamster, “A smart home is no castle: Privacy vulnerabilities of encrypted IoT traffic,” in DAT, 2016.

[9] E. Anthi, L. Williams and P. Burnap, ”Pulse: An adaptive intrusion detection for the Internet of Things,” Living in the Internet of Things: Cybersecurity of the IoT, London, 2018.

[10] N. Ammar, L. Noirie and S. Tixeuil, ”Network-Protocol-Based IoT Device Identification,” 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC), Rome, Italy, 2019, pp. 204-209.

[11] M. Miettinen, S. Marchal, I. Hafeez, N. Asokan, A. Sadeghi and S. Tarkoma, ”IoT SENTINEL: Automated Device-Type Identification for Security Enforcement in IoT,” 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, 2017, pp. 2177-2184.

[12] S. Woo, H. J. Jo and D. H. Lee, ”A Practical Wireless Attack on the Connected Car and Security Protocol for In-Vehicle CAN,” in IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 2, pp. 993-1006, April 2015.

[13] “Cisco Visual Networking Index: Forecast and Trends,” 2017-2022.

[14] “5 Infamous IoT Hacks and Vulnerabilities,” IoT World Congress, 2018.

[15] S. Molnar, P. Megyesi, and G. Szabo, “How to Validate Traffic Generators?”, IEEE ICC Workshops 2013.

[16] S. Mishra, S. Sonavane, and A. Gupta, “Study of Traffic Generation Tools”, IJARCCE, pp. 159-162, 2015.

[17] O. Bello and S. Zeadally, “Communication Issues in the Internet of Things (IoT)”, Springer London, pp. 189–219, 2013.

[18] L. Bai, L. Yao, S. S. Kanhere, X. Wang, and Z. Yang, “Automatic Device Classification from Network Traffic Streams of Internet of Things”, IEEE LCN 2018.

[19] “Scapy Project”, Biondi, 2014.

[20] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, “Towards the Development of Realistic Botnet Dataset in the Internet of Things for Network Forensic Analytics: Bot-IoT Dataset”, FGCS, 2019.

[21] A. Sivanathan, D. Sherratt, H. Gharakheili, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman, “Characterizing and Classifying IoT Traffic in Smart Cities and Campuses” IEEE Infocom WS 2017.

[22] A. M. Rahmani, N. K. Thanigaivelan, T. N. Gia, J. Granados, B. Negash, P. Liljeberg, and H. Tenhunen, “Smart E-health Gateway: Bringing Intelligence to Internet-of-Things based Ubiquitous Healthcare Systems”, IEEE CCNC 2015.

[23] M. Kang, E. Park, B. H. Cho, and K. S. Lee, “Recent Patient Health Monitoring Platforms Incorporating Internet of Things-Enabled Smart Devices,” IN Journal, vol. 22, pp. S76–82, 2018.

[24] D. C. McFarlane, A. K. Doig, J. A. Agutter, J. L. Mercurio, R. Mittu, L. M. Brewer, and N. D. Syroid, “Defeating Information Overload in Health Surveillance using a Metacognitive Aid Innovation from Military Combat Systems”, J. of DM & S., vol. 14, no. 4, pp. 371–388, 2017.

[25] D. C. McFarlane, A. K. Doig, J. A. Agutter, L. M. Brewer, N. D. Syroid, and R. Mittu, “Faster Clinical Response to the Onset of Adverse Events: A Wearable Metacognitive Attention Aid for Nurse Triage of Clinical Alarms”, PLOS ONE, vol. 13, no. 5, p. e0197157, May 2018.

[26] R. Ferrando and P. Stacey, “Classification of Device Behaviour in Internet of Things Infrastructures: Towards Distinguishing the Abnormal from Security Threats,” IML 2017.

[27] C. Majumdar, M. Löpez-Benítez, and S. Merchant, “Experimental Evaluation of the Poissoness of Real Sensor Data Traffic in the Internet of Things,” in IEEE CCNC 2019.

[28] M. R. Shahid, G. Blanc, Z. Zhang, and H. Debar, “IoT devices recognition through network traffic analysis,” IEEE Big Data, 2018.

[29] L. Arnaboldi and C. Morisset, “Generating Synthetic Data for Real World Detection of DoS Attacks in the IoT,” in Lecture Notes in Computer Science, pp. 130–145, 2018.

[30] L. Arnaboldi and C. Morisset, “Generating Synthetic Data for Real World Detection of DoS Attacks in the IoT”, LNCS, pp.130–145, 2018.

[31] F. Erlacher and F. Dressler, “How to test an ids?: Genesids: An Automated System for Generating Attack Traffic”, SIGCOMM, 2018.

[32] K. Huang, Q. Zhang, C. Zhou, N. Xiong, and Y. Qin, “An Efficient Intrusion Detection Approach for Visual Sensor Networks Based on Traffic Pattern Learning”, IEEE SMC, pp. 2704–2713, Oct. 2017.

[33] S. Papadopoulos, A. Drosou, N. Dimitriou, O. H. Abdelrahman, G. Gorbil, and D. Tzovaras, “A BRPCA Based Approach for Anomaly Detection in Mobile Networks,” in Springer Information Sciences and Systems, pp. 115–125, 2015.

[34] O. Salem, Y. Liu, and A. Mehaoua, “Anomaly Detection in Medical Wireless Sensor Networks”, J. of CS & Eng., vol. 7, no. 4, p. 13, 2013.

[35] P. Casas, P. Fiadino, and A. D’Alconzo, “Machine-Learning Based Approaches for Anomaly Detection and Classification in Cellular Networks”, IFIP TMA 2016.

[36] S. Papadopoulos, A. Drosou, N. Dimitriou, O. H. Abdelrahman, G. Gorbil, and D. Tzovaras, “A BRPCA Based Approach for Anomaly Detection in Mobile Networks,” in Information Sciences and Systems 2015, O. H. Abdelrahman, E. Gelenbe, G. Gorbil, and R. Lent, Eds. Springer International Publishing, 2016, pp. 115–125.

[37] ”2017 Corporate Social Responsibility Report”, 2017.

[38] J. W. Chuah, “The Internet of Things: An Overview and New Perspectives in Systems Design,” International Symposium on Integrated Circuits (ISIC), 2014.

[39] T. Macaulay, “RIoT Control: Understanding and Managing Risks and the Internet of Things,” Elsevier, 2016.

[40] C.Kolias, G. Kambourakis, A. Stavrou, and J. Voas, ”DDoS in the IoT: Mirai and other Botnets,” Computer, pp. 80-84, 2017.

[41] P. Bereziński, B. Jasiul, and M. Szpyrka, ”An entropy-based network anomaly detection method,” Entropy 17.4, pp. 2367-2408, 2015.

[42] P. Bereziński, M. Szpyrka, B. Jasiul, and M. Mazur, ”Network anomaly detection using parameterized entropy,” IFIP Int. Conf. on Comput. Inf. Syst. and Ind. Manage., pp. 465-478, 2015.

[43] R. Ooka, T. Miyoshi, and T. Yamazaki, "Unit Traffic Classification and Analysis on P2P Video Delivery Using Machine Learning," IEICE Commun. Express (ComEX), Vol. 8, No. 12, pp. 640-645, Dec. 2019.

[44] P. Refaeilzadeh, L. Tang, H. Liu, "Cross-Validation," Encyclopedia of Database Systems, Springer, Boston, MA, 2009.

[45] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," ACM SIGKDD explorations newsletter, vol. 11, no. 1, pp. 10–18, 2009.

[46] M. R. Shahid, G. Blanc, Z. Zhang, and H. Debar, "IoT devices Recognition Through Network Traffic Analysis," IEEE Big Data, 2018.

[47] Y. Feng, L. Deng, D. Chen, "IoT Devices Discovery and Identification using Network Traffic Data: poster," Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, ACM, 2019.

[48] B. Bezawada, M. Bachani, J. Peterson, H. Shirazi, I. Ray, and I. Ray, "Iotsense: Behavioral Fingerprinting of IoT Devices," CoRR, 2018.

[49] J. Ortiz, C. Crawford, and F. Le, "Devicemien: Network Device Behavior Modeling for Identifying Unknown IoT Device," in Proc. of the Int. Conf. on IoT Des. and Implementation, ACM, pp. 106–117, 2019.

[50] C. Adjih, E. Baccelli, E. Fleury, G. Harter, N. Mitton, T. Noel, R. Pisard Gibollet, F. Saint Marcel, G. Schreiner, J. Vandaele, T. Watteyne, "FIT IoT-LAB: A large scale open experimental IoT testbed" IEEE 2nd World Forum on Internet of Things, pp. 459-464, 2015.

[51] I. Chatzigiannakis, S. Fischer, C. Koninis, G. Mylonas, Pfisterer, "D. WISEBED: an open large-scale wireless sensor network testbed," International Conference on Sensor Applications, Experimentation and Logistics, pp. 68-87, 2009.

[52] G. Nychis, V. Sekar, D. G. Andersen, H. Kim, and H. Zhang, "An empirical evaluation of entropy-based traffic anomaly detection," ACM SIGCOMM IMC, pp. 151–156, 2008.

[53] A. S. Shukla and R. Maurya, “Entropy-Based Anomaly Detection in a Network,” Wireless Personal Communications, vol. 99, no. 4, pp. 1487-1501, Apr. 2018.

[54] C. Callegari, S. Giordano, and M. Pagano, “Entropy-based network anomaly detection,” in 2017 International Conference on Computing, Networking and Communications (ICNC), pp. 334–340, Jan. 2017.

[55] M. Ozcelik, N. Chalabianloo, and G. Gur, ”Software-defined Edge Defense Against IoT-based DDoS,” IEEE international conference on computer and information technology (IEEE CIT), pp. 308–313, 2017.

[56] Fu, Z. Yan, J. Cao, O. Kon, and X. Cao, “An Automata Based Intrusion Detection Method for Internet of Things,” Mobile Information Systems, May. 2017.

[57] M. Gajewski, J. M. Batalla, A. Levi, C. Togay, C. X. Mavromoustakis, and G. Mastorakis, ”Two-tier anomaly detection based on traffic profiling of the home automation system,” Computer Networks, vol. 158, pp. 46-60, 2019.

[58] V. Martin, Q. Cao, and T. Benson, ”Fending off IoT-hunting attacks at home networks” Proceedings of the 2nd Workshop on Cloud-Assisted Networking, pp. 67-72, 2017.

[59] D.H. Summerville, K.M. Zach, and Y. Chen, ”Ultra-lightweight deep packet anomaly detection for Internet of Things devices,” IEEE 34th international performance computing and communications conference, pp. 1-8, 2015.

[60] J. M. R. Danda, and C. Hota, ”Attack identification framework for IoT devices,” Information Systems Design and Intelligent Applications, pp. 505-513, 2016.

[61] “New IDC Forecast,” 2019.

[62] M. Alaa, A. A. Zaidan, B. B. Zaidan, M. Talal, and M. L. M. Kiah, “A review of smart home applications based on Internet of Things,” J. Netw. & Comput. Appl., vol. 97, pp. 48–65, Nov. 2017.

[63] S. Srivastava, S. Anmulwar, A. M. Sapkal, T. Batra, A. K. Gupta, and V. Kumar, ”Comparative study of various traffic generator tools,” in Proc. 2014

Recent Advances in Eng. & Comput. Sci. (RAECS'14), Chandigarh, India, pp. 1-6, 2014.