INSTITUTO POLITÉCNICO DE BRAGANÇA Escola Superior de Tecnologia e Gestão

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

Analyzing IDS Botnets Detection

Kahe Henrique Binda

Thesis presented to the School of Technology and Management of Polytechnic Institute of Bragança to the Fulfillment of the Requirements for the Master of Science Degree in System Information, in the scope of Double Degree with Federal University of Technology - Paraná

Supervised by:
Prof. Tiago Pedrosa
Prof. Nuno Rodrigues
Prof. Neylor Michel

Bragança
2018

Analyzing IDS Botnets Detection

Kahe Henrique Binda

Thesis presented to the School of Technology and Management of Polytechnic Institute of Bragança to the Fulfillment of the Requirements for the Master of Science Degree in System Information, in the scope of Double Degree with Federal University of Technology - Paraná

Supervised by:
Prof. Tiago Pedrosa
Prof. Nuno Rodrigues
Prof. Neylor Michel

Bragança
2018

Acknowledgments

I acknowledge and express thanks to both Federal University of Technology - Paraná (UTFPR), Campus Medianeira and the Polytechnic Institute of Bragança (IPB) for the opportunity provided and for all the effort made by these two institutions and the people who represent it, that allowed me this study and research opportunity.

I appreciate all the help and patience that my mentors Professor Tiago Pedrosa, Professor Nuno Rodrigues and Professor Neylor Michel, had with me during our working in this project.

I also like to thank my mother Ana, my father David and my girlfriend Barbara for the support during all those months, without them this project would not be possible. All the friends that I made in Bragança for supporting me in both good and bad moments.

Abstract

In a world increasingly connected with equipment permanently attached, the risk of cybersecurity had rise. Among the various vulnerabilities and forms of exploitation, the Botnets are those being addressed in this work. The number of botnets related infections has grown critically and, due to botnets' increased capacity and potential use for future infections, a continued development of solutions is needed to strengthen the protection of networks and systems. Intrusion Detection Systems (IDS) are one of the solutions that try to follow this evolution. The continuous evolution of tools and attack forms in order to evade detection, using mechanisms such as encryption (IPSec, SSL) and diverse architecture and different ways of implementing Botnets create great challenges to those who try to detect them. In order to better understand these challenges, this work proposes an architecture to map the behavior of botnets. For this, a topology was created with several components, such as Network Intrusion Detection System (NIDS) and Host Intrusion Detection System (HIDS), aided with information from honeypots for the detection and analysis of attacks. This approach enabled real data to be obtained from attempts, some successfully, from Malware infections, with the aim of transforming systems into Bots and integrating them into Botnets. An exploratory analysis of the data is performed to verify the detection capabilities and the cases where the components do not provide correct information. Some methods based on machine learning were also used to process and analyze the collected data.

Keywords: Botnet, IDS, Malware, Machine learning.

Resumo

Num mundo cada vez mais conectado com cada vez mais equipamentos ligados em permanência o risco de cibersegurança tem aumentado. De entre as diversas vulnerabilidades e formas de exploração continuada as Botnets são as visadas neste trabalho. Os números de infeções relacionadas com as Botnets têm crescido de forma critica e devido dotar de maiores capacidades os atacantes e seu grande poder de infeção futura é necessário um desenvolvimento continuo de soluções para reforçar a proteção das redes e sistemas. Os Sistemas de Deteccao de Intrusao (IDS) são uma das soluções que tentam acompanhar esta evolução deste tipo de ameaça. A evolução continua das ferramentas e formas de ataque por forma a fugir à detecção, utilizando mecanismos como tráfego cifrado (IPSec, SSL) e arquitectura diversa e formas diferentes da implementação das Botnets levantam grandes desafios a quem as tenta detectar. Por forma a compreender melhor estes desafios, este trabalho propõe uma arquitetura para mapear o comportamento das Botnets. Para isso criou-se uma topologia com diversos componentes, como Network Intrusion Detection System (NIDS) e Host Intrusion Detection System (HIDS), auxiliados com informação de honeypots para a deteção e análise de ataques. Esta abordagem permitiu obter dados reais de tentativas, algumas com sucesso, de infeções de Malware, com o intuito de transformar os sistemas em Bots e os integrar em Botnets. É efetuada uma análise exploratória dos dados para verificar a capacidade de deteção e os casos em que os sistemas não fornecem informação correta. Foram também utilizados alguns métodos baseados em machine learning para tratamento e análise dos dados coletados.

Palavras-chave: Botnet, IDS, Malware, Machine learning.

Contents

List of Tables

List of Figures

Acronyms

.pcap Packet Capture.

AI Artificial Intelligence.

C&C Command and Control.

CINS Collective Intelligence Network Security.

DDoS Distributed Denial of Service.

DHCP Dynamic Host Configuration Protocol.

DMZ Demilitarized Zone.

DNS Domain Name System.

DoS Denial of Service.

HIDS Host Intrusion Detection System.

HTTP Lypertext Transfer Protocol.

IDS Intrusion Detection System.

IP Internet Protocols.

IPB Instituto Politécnico de Bragança.

IPS Intrusion Prevention System.

IRC Internet Relay Chat.

MYSQL MY Structured Query Language.

NIDS Network Intrusion Detection System.

OSI Open System Interconnection.

P2P Peer to Peer.

PDU Protocol Data Unit.

PPTP Point-to-Point Tunneling Protocol.

SIP Session Initiation Protocol.

SSQL Semi-Structured Query Language.

Chapter 1

Introduction

Among the biggest threats that can be found on the Internet, we can highlight Botnets as one of the biggest risks. They can infect multiple computers and mobile devicse worldwide, from common users to complete infection of networks of educational institutions, government departments and companies.

Mobile devices are increasingly being targeted for malware. A trend that is growing in the first half of 2018, is the malware pre-installed on the devices. The RottenSys botnet is responsible for infecting almost 5 million devices [1].

Hosts are infected by Malware and are controlled remotely becoming a Bot, Bots are aggregated in networks (botnets). The Botmaster is the actor that to control all the Bots in a Botnet.

In the year 2018, Spamhaus posted a note on its Spamhaus Block List (SBL), recording about 9500 Command and Control (C&C) with a 32% increase. About 68% of those C&C are found and hosted on cybercriminals managed servers. The Botnet Controller List (BCL) is a subset of the SBL containing only IPv4 addresses of bots and C&C detected, compared with 2016 an increase of 40% and more than 90% over the year 2014, in Figure 1.1 can be visualized the increase of botnets between the year 2014 and 2018 [2].

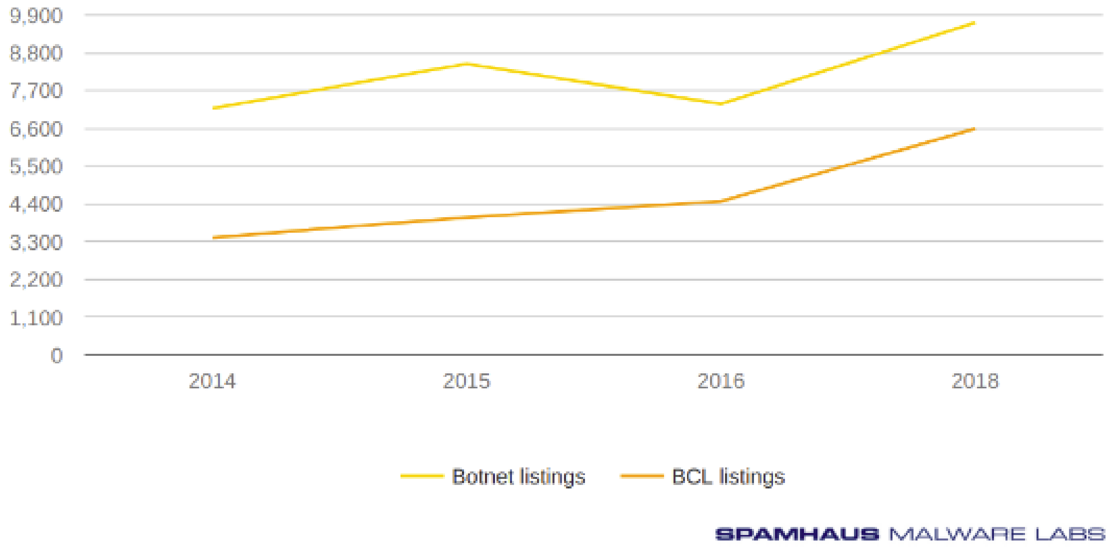


Figure 1.1: Botnet Listings VC BCL listings [2]

There are several types of Botnets with unique architectures and with different objectives. As they are scattered in the greatest amount possible in devices connected to the Internet they give huge resources to the Botmasters. Enabling them to perform Distributed Denial of Service (DDoS) attacks, to capture personal information in the hosts and use the victim's computational resources for any action that Botmaster wants to take. Due to this great diversity of Botnets, it is essential to understand their operation mode and ways of detecting them.

In a comparison between the first half of 2017 and 2018, depicted on Figure 1.2, it is registered a decrease on the number of attacks due to the seasonal slowdown generated at the beginning of the year, but the H1 indicators show a significant increase in attacks [3].

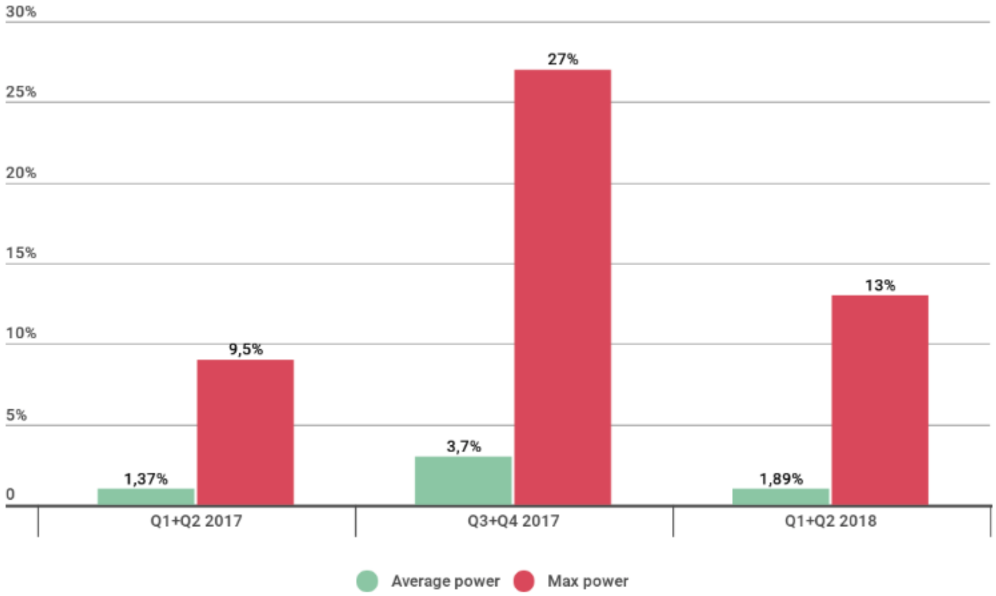


Figure 1.2: Change in DDoS attack power, 2017-2018 [3]

1.1 Objectives

This section shows the objectives to this work.

Considering the diversity of botnet infections is important to study in depth how they works, specially on the infection phase of the bots and their coordination from C&C hosts. The main objective of this work is to analyze how Intrusion Detection System (IDS) detects Botnets and foster detection improvement.

Considering the main objective, several phases were considered important in the planning:

- Build an architecture to analyze the traffic of botnets;
- Redirect all recorded logs to a central log server;
- Infecting hosts where Host Intrusion Detection System (HIDS) are configured;
- Analyze network traffic generated by Network Intrusion Detection System (NIDS);

- Analyze logs generated by Honeypots;
- Correlation between the data record in the diverse components;
- Implement machine learning algorithms to improve the detection of botnets.

1.2 Document Structure

This document is constituted by five chapters, the first chapter starts with the introduction, objectives and the structure of this work.

In the second chapter, an analysis of the state in art related, along with explanation of systems and devices used to develop the work presented.

The third chapter consists of describing and explaining the methodology used for the creation of a topology in order to capture events from the HIDS, NIDS and Honeypots, and for the infection and analysis phases.

The fourth chapter explains the procedures followed in the analysis phase using machine learning algorithms and presents the results.

The fifth chapter makes the conclusion of this work and considers future work research vectors.

Chapter 2

Literature Review

2.1 Cyber-Attack

At the beginning of computer networks, systems were very simple. Over time, the importance of connectivity grew and systems became more complex. Today there is no completly invulnerable computer system as it is possible to find many talented hackers, and consequently, all systems need several ways to ensure security [4]. The following section describes the Cyber-attacks, their reasons and their types, which can include: attacks, network scans, malware, Denial of Service and Distributed Denial of Service (DDoS).

2.1.1 Motivations

The big expansion of computers use and network usage in the last few years promoted what we, nowadays commonly know as cyberspace. On this space, diverse things are happening all the time, not all of them are benign, such as Cyber-attacks which are Internet Crimes. These cyber-attacks have become a serious problem in the 21st century, and there are several new techniques being developed, always complex to break the security systems and to obtain an advantage [4].

The Internet services, like E-commerce, Internet Banking, Social Network and others, create a huge amount of information, which generally is stored in servers and clouds (i.e. remote servers). In order to have greater availability and agility to compete in the market, those Internet services usually have weak security techniques, exposing weaknesses in the system that can be used to invade and hack them [5].

The biggest challenge to enssure security at information systems is how to keep large quantities of information, that can be simple data, such as photos, or very complex and important, such as financial transactions. These different applications operate on different computers architectures, which produce a large amount of data and require efficient processes to ensure the security of organizations [6].

2.1.2 Types of Cyber-Attacks

There are many types of cyber attacks and, especially, the attacker relies on common hacking techniques. Usually, these techniques are not highly effective, so every day new techniques are being invented. so it's to important study this and understand how this works and the different ways an attacker can execute [7]. Next, we identify the most important ones:

- Recognition and Collection: The invader executes a data gathering and data preprocessing of the system that he will attack. This recognition has three types: active, passive and sniffing [8].

- Backdoor: It is used to open a door inside the system, which means that is possible to execute remote commands behind the security system. The Backdoor make it possible to have a connection with the destination network avoiding any kind of detection. It is also possible to project a backdoor specifically to avoid any type of IDS [9].

- Network Scanner: A network scanner is performed to find possible targets and security failures. This process can be legal or illegal. The legal way proceeds through authorized people or by network security professionals. They aim to find the breaks, afterward correct them and then implement a defense process for new possible attacks. The illegal scan proceeds through malicious people who search for failures to invade the system [8].

- Malicious Codes: Known as malware, their goals are malicious activities. After the malware is installed, the attacker has access to the computer with administrator user, therefore, he can access all the information. The reason for an attacker to create a malware normally is to stole confidential information, scam practices, attacks and also the spam distribution. Two examples of programs are: [8].

  Virus: It is usually installed on the computer through the Internet. It can self-copy on the infected files, spreading quickly. The most common source for virus infections is the E-mail, where the host gets infected when the malicious email is opened by the user. Then, the malicious software can access all the user contacts and self-send to them. Multiple types of virus exist and every day new types are developed. Some of them can be highlighted: False Alarm, Backdoor, Trojan Horse, Macro [8].

  Spyware: Treated as a spy, it can monitor the infected computer activities and pass along all the collected information. Spyware programs are also classified as legal and illegal, the ones that are managed to monitor who uses the machine and those that are managed to steal sensible information, respectively [8].

- Denial of Service (DoS): When services are not available for being consumed, this attack can be: coordinated and distributed [8].

  DDoS Occurs when several hosts direct an attack to the server, achieving a number of solicitations bigger than it can support, so the system becomes unavailable [8].

  DoS cannot modify the content on computer data, networks, and systems. Regularly the victim has no idea of the attack, due to the fact that they are just to send requests to the server. The server attacked after suffering several attacks stop responding to the legal user and starts only to take care of the attacks [8].

- Social Engineering: a technique that applies persuasion. The attacker will persuade the victim to perform a certain action or to give some relevant information. These actions can cause damage in the entire computer or network, and the main vulnerability in those cases is the victim, that has no conscience about the danger on the Internet[4].

2.2 Security Controls and Mechanisms

The following section approaches some security controls and mechanisms considered important to the execution of the proposed goals. Starts by the explanation of the device that typically comes first and is used to protect a network against invasions, the Firewall, followed by the concepts and purpose analysis of the IDS and then of the Intrusion Prevention System (IPS).

2.2.1 Firewall

Firewall is a security device that can monitor the network traffic. Moreover, it can block a specific traffic flow based on a set of rules. The Firewall is the first implemented line of defense on computers against possible attackers [10]. An example the firewall can be seen in Figure 2.1

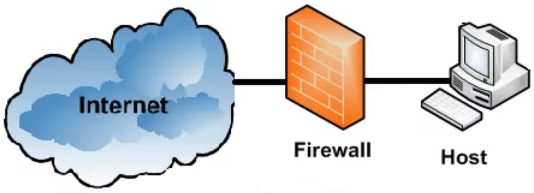


Figure 2.1: Firewall

A Firewall can inspect the system, and also allow or block traffic based on state, port, and protocol. These inspection rules are set by the administrator and can match the source address, destination address, and access parameters. The inspections use known information that comes from previous connections and packets [10].

But these Firewall rules are not enough to prevent all the attacks, due to some architectural limitations. It only prevent the intrusions from traffic as long as the allowed data matches to the applied set of rules, otherwise, the firewall will not notice the intruder. To minimize those limitations the IDS and the IPS are created [11].

The main difference between a Firewall and an IDS is in which layer it analyzes the package. The Firewall analyzes only two layers of the Open System Interconnection (OSI) model the network layer and the transport layer. Nevertheless, IDS analyzes the package body. The OSI model was developed to build network protocols . OSI consist of seven distinct layers with the fundamental ideas of networking [12].

2.2.2 Intrusion Detection System (IDS)

An IDS detects several Internet attacks on computer networks. It can monitor the attacks in other hosts and determinate if the attack is random, general or to a specific computer network. The IDS can analyze the Protocol Data Unit (PDU) all OSI layers and can report an alert to the administrator or add to a logs list. Moreover, the IDS also can test the vulnerability of the computer to attacks on others monitored host [13].

IDS represents the next step network security system evolution, where the software has the functionality to prevent known or unknown attacks. IDS is able to decrypts attacks in layers that a Firewall cannot[14]. Can be seen in Figure 2.2

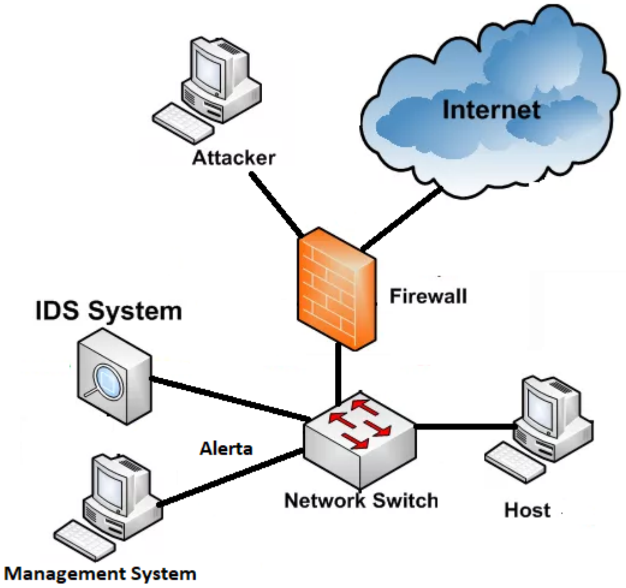


Figure 2.2: Intrusion Detection System Architecture

The IDS can be classified into two types:

- Based on signatures;
- Based on anomalies.

The ones that are based on signatures work with a table of known signatures of possibles attack or with access rule. It has a fast identification mechanism but requires a database that has to be frequently updated [15]. and regularly are created an large number new signatures, for this reason the method based on signatures can be limited

The anomalies based method, collects the current network traffic and after a period the system make an analysis, for all the assumed not regular and the can sends an if something is wrong. This is a robust method because all the unknown attacks can be prevented, and it biggest disadvantage are the false positives, because a not regular traffic is not necessarily an attack [15].

The hybrid system can combine both techniques, but they are more complex systems, implementing many restrictions or limitations to filter the maximum threats [16].

2.2.3 Intrusion Prevention System (IPS)

Considered as an extension of IDS, an IPS can be seen in the image 2.3. Can monitor the network traffic and search for malicious activities. The main difference is that IPS is installed in-line, which means that it stays in the communication path between the source and the destination, actively analyzing the traffic and reporting or blocking an intrusion that was detected [17]

The IPS is an active solution system, unlike the IDS which is passive. IPS analyzes the Logs generated by the IDS and takes active measures, like blocking Internet Protocols (IP) packets or alerting the Firewall to block inboard or outboard data [18]. IPS is able to operate invisibly on a network, and offers deep watch and monitor bad logons, inappropriate content, bad behavior among others [19].

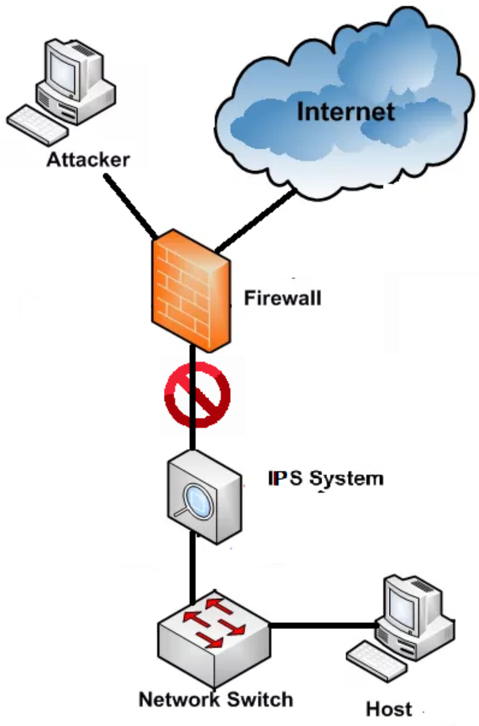


Figure 2.3: Intrusion Prevention System Architecture

2.3 Honeypot

Honeypot is a tool to collect information about the attacker through a trap, it can be just a simulation of a system or a host that can give false access to the attacker. The following section has the explanation about the Honeypot, its types and where they can be installed, can be seen in the Figure 2.4.

Honeypot services have basically high and low interaction. High Interaction happens when the system has all the services simulated. The intruder will hardly notice that the machine is a Honeypot, that can be dangerous knowing that the attacker will have access to a machine, likewise, it is possible to obtain extra information about the attacker [15].

The low interaction is the opposite, it is characterized by a system that clones real services where the attacker will not have real access. The critical issue with this approach is that a knowledgeable attacker will quickly realize that it is a Honeypot and abort the attack, however, it is possible to capture some data [15].

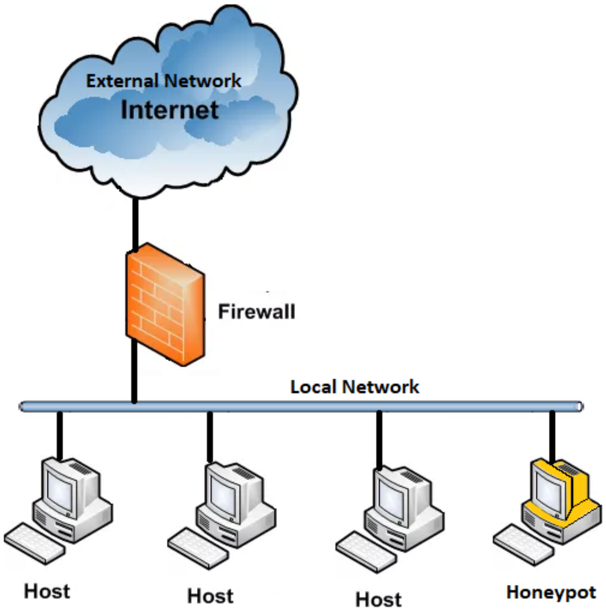


Figure 2.4: Honeypot Architecture

There are mainly two types of Honeypots: research and production.

The function of a research Honeypot is to be an attacking target and to collect information and intelligence about general threat organizations. It is often used for academic research, by enterprises and by all researchers/professionals that want to improve their skills. It enables to attract and study new methods and tools used by attackers.

Usually, the research Honeypot is from high interaction type, considering that its goal is precisely to study what the attacker is looking for in the system [20].

The production Honeypot detect and save data from some intrusion that might occur. With the collected data it is possible to improve the defenses against future threats. Production Honeypots an mostly used by companies, due to its immediate security provision, and easier deployment [20].

The Honeypot can be placed:

- Before the Firewall: Precisely to be the first target, and to capture extended information of the attacker.
- Inside the Demilitarized Zone (DMZ): The Honeypot stays together with other servers, so the attacker can find it, strikes it and falls into the trap.
- After the second Firewall and together to the internal network: This Honeypot aims to catch possible attackers on the internal network that can be performed by employees or people with access to the local network.

It is the responsibility of the administrator to analyze which is position to their network. The Honeypot's risk is the condition where an attacker is able to gain access and manage the network to arrange more attacks [15].

2.4 Botnet

This section will introduce the Botnets, its function and how it works, the basic components that integrate its architecture, its infrastructures, that can be centralized, decentralized and hybrid and the life-cycle of a botnet.

Botnets are one of the biggest threats to the Internet users. They are formed by several hosts that work to a determined person: the Botmaster. The hosts that are participating on the botnet are receiving and transmitting information from a C&C [21].

Botnets are like a computer army waiting for commands to act maliciously and one of the botnet's biggest advantage is its anonymity because the infected components do not belong to the attacker, therefore, identify the actual attacker is difficult. Notwithstanding each bot can be anywhere in the world acting distributed [22].

To expand the reach of botnets, they infect new vulnerable systems, as more systems become infected, it becomes a massive activity, and consequentely, offer a significant threat to the Internet and business companies [23].

2.4.1 Botnet's Basic Components

Botnets are networks of bots, which are remotely managed by the botmaster through the C&C. The botnets have some basic components, which are:

- Bot: Malware installed on the user host, usually used to malicious actions.

- User host: Physical or virtual machine infected by the bot.

- C&C: Command & Controler Server it is the way that the botmaster communicate, sends/receives information and commands to the bots.

- Botmaster: An individual who controls the bots, sending/receiving information and commands to a possible attack [24].

2.4.2 Infrastructure

A C&C server is the most relevant component for a Botnet infrastructure. Through it the bots receive the information and mandatory all bots must have an active connection to the server. This structure has two approaches, centralized and decentralized [22].

Centralized Architecture

Centralized architecture is compatible with the client-server model Figure 2.5, which bots are clients sending requests to the server and, consequently, the system gives his instructions, which brings stability to the network and fast response time [22].

Bots have an individual connection with the server and the Botmaster controls it. With to this direct connection, it is possible to send simultaneously commands to all bots, moreover to monitor the number of bots in the network [22].

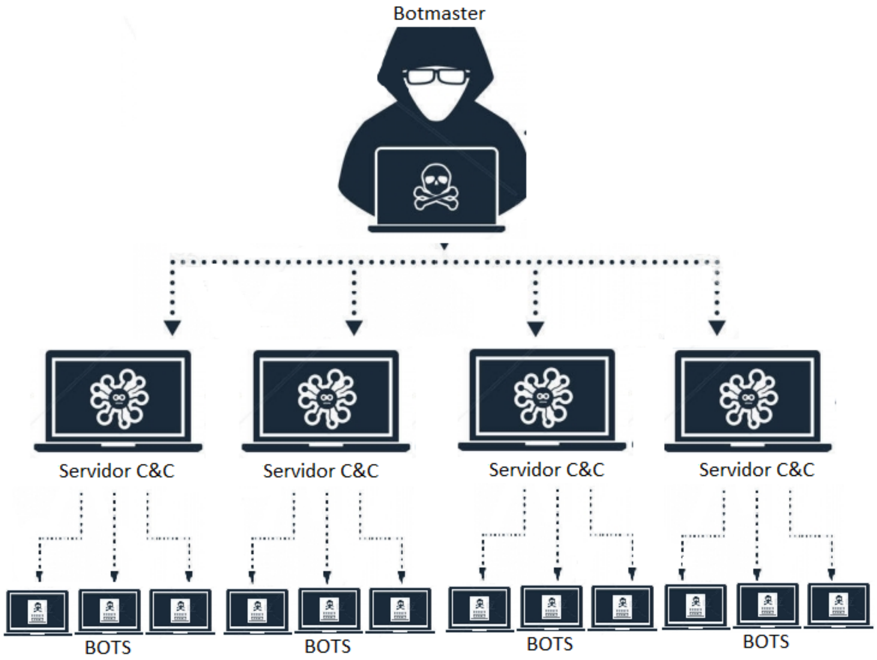


Figure 2.5: Centralized Architecture of a Botnet

The Internet Relay Chat Internet Relay Chat (IRC) protocol is still widely used to support the communication in centralized architectures. The main advantage of this protocol is that the number of bots is not limited, allowing thousands of it being added to the parallel network [25].

asdasdasdasdad IRC is basically text, so it is possible to create private conversations with hosts individually, originating a more specific manipulation, consequently needing to implement only one IRC instruction subset [25].

Another used protocol for Botnets with a centralized architecture is the Lypertext Transfer Protocol (HTTP). As it is the most used protocol on the Internet for data delivery, it has a great availability and its contents are usually rightful. This protocol uses weak filtration leaving a breach to the Botmaster control the network. These HTTP Botnets do not hold a connection to a C&C server, the bots uses regular intervals to contact the server, configured by the Botmaster [25].

Decentralized Architecture

With the advances in detection techniques, the centralized architecture could not provide a complete protection of the identity, therefore, a new decentralized architecture was created, shown in Figure 2.6 [22].

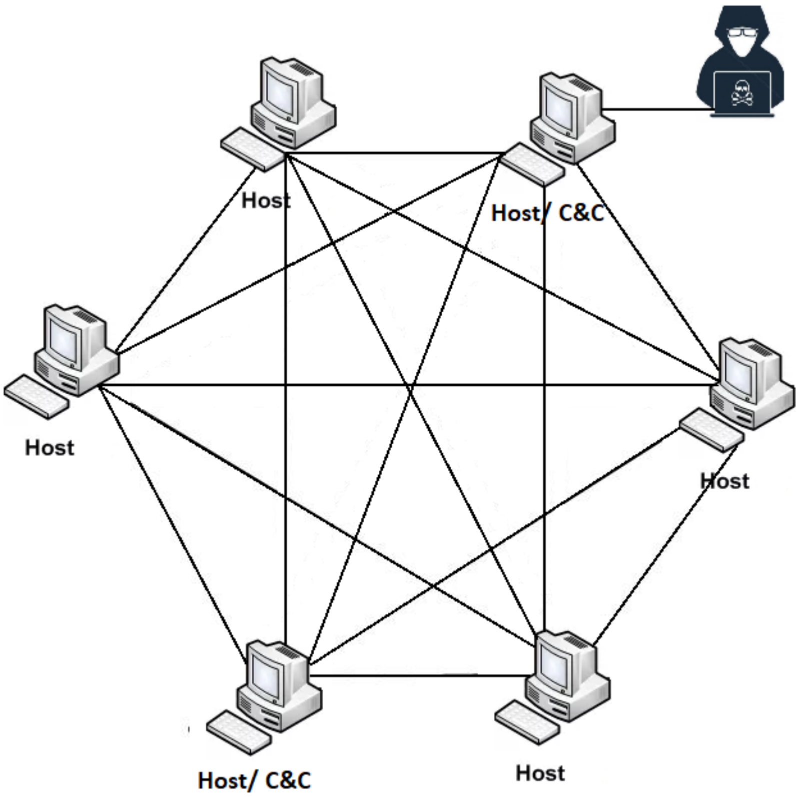


Figure 2.6: Decentralized Architecture of a Botnet

The decentralized architecture has no central C&C servers. Each bot is directly connected to another. The biggest benefit of this approach is the difficulty in locating the Botmaster, due to the big number of bots. On the other hand, the reaction time is longer because there is no central command to give the information [22].

P2P is the most common type of decentralized Botnet. In this architecture the bots work as a client as well as a server. The Botmaster uses a special key to send the commands to the bots, and even if the bots are off line, the Botnet remains working under the control of the Botmaster [26].

Hybrid Architecture

The Hybrid architecture uses components of both centralized and decentralized architectures 2.7. With this architecture, it is possible to obtain the advantages of both: the efficiency of the C&C server, from the centralized and the practicality of the anonymity, from the decentralized. This type uses HTTP, IRC, and P2P protocols to attend all the Botmaster needs [27].

The most progressive and demanding communication to protect a network is considered the Hybrid P2P, that possess the ability to exchange information and services to each other. Its three parts that are: Botmaster, Social Websites, Bot Group [28].

The Figure 2.7 shows a diagram explaining the process, that consists of the Botmaster implementing a malicious code into the website, then the servant bot or C&C obtains the malware information to send to the client bot. After receiving this information, the client bot attacks the target [28].

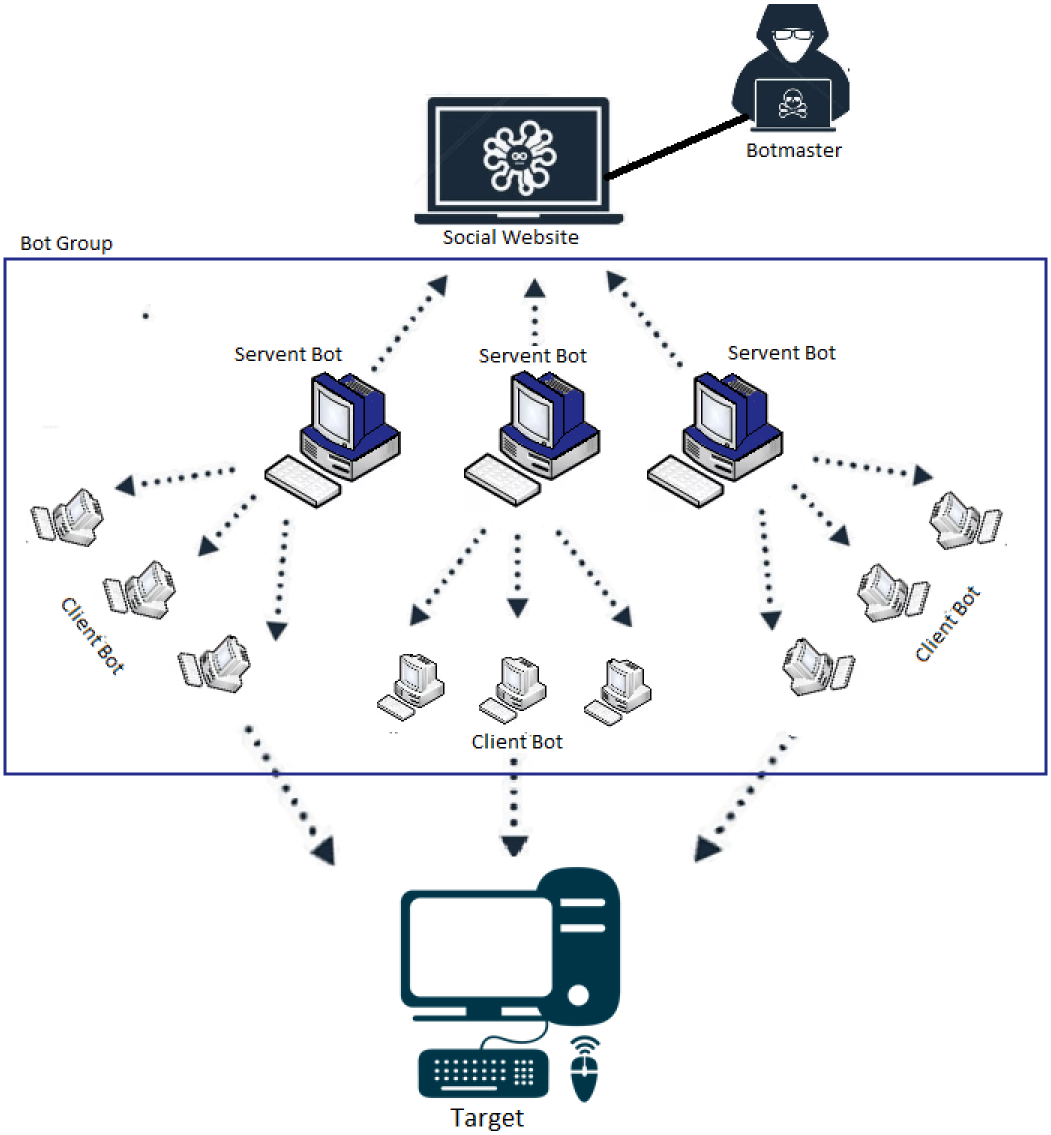


Figure 2.7: Hybrid P2P Architecture - Adapted from [29]

2.4.3 Botnet Life-Cycle

Usually, the botnets have regular steps or a similar behavior in the recruitment of vulnerable systems and to managed that, therefore, it has a life-cycle [20]

The typical botnet life-cycle has five phases Figure 2.8. In the initial infection phase, the Botnet search for vulnerabilities through a scanner of a possible target and then to infect with different methods. After the infection, starts the secondary injection phase, while the target becomes the bot. Through the chosen method to infect, it searches on a network the actual bot binary malware. The bot binary installs itself on the host, and after the installation, the host starts running the program making the host into an actual bot [27], [30].

On the connection phase, the bot establishes a C&C channel, while this process runs over and over, and this is a critical phase, due to its necessity. In the malicious command and control phase the target bot becomes a Botnet army, responding to the Botmaster commands. The last phase is the maintenance and update, the Botmaster needs to maintain the Botnet active and updated while to avoid new detection techniques [27], [30].

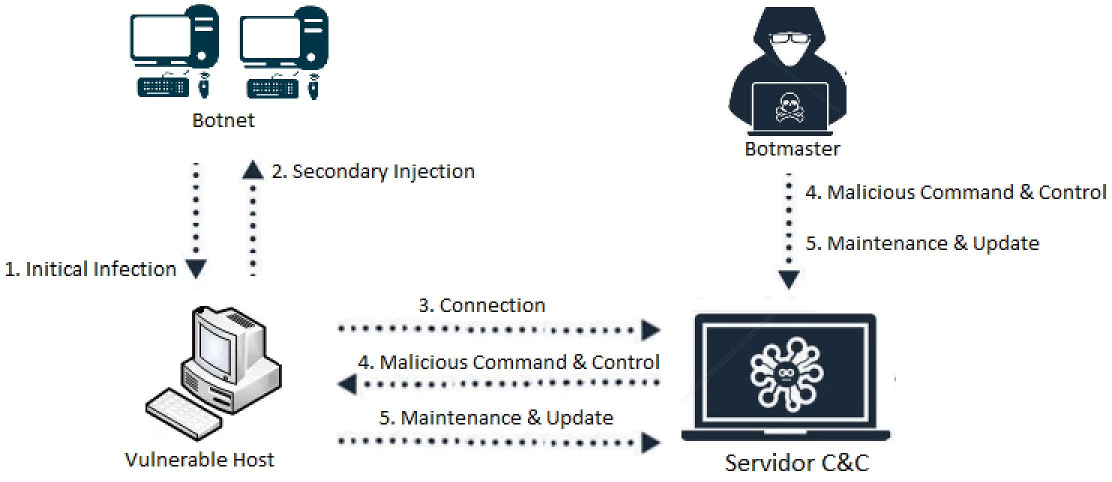


Figure 2.8: Botnet Life-Cycle - Adapted from [30]

2.5 Machine Learning Algorithms

The Artificial Intelligence (AI) enables machines to learn and think. The machine apprenticeship is a subfield of the AI study. There are many techniques in Machine

Learning used to classify data sets: Supervised, Unsupervised, Semi-Supervised, Reinforcement, Evolutionary Learning, and Deep Learning [31]. These techniques will be explained in this section.

- Supervised Learning: It is a very common technique that, can easily deduct a classification problem. The purpose is to make a computer learn a classification that the user has created. The supervised is able to map the inputs to desired outputs. With a pre-determined classification, supervised learning is very used for training neural networks and decision trees [32].

- Unsupervised Learning: The goal in this technique is to teach the computer how to solve a problem without knowing patterns. The unsupervised tries to find similarities in the data and classify them [32].

- Semi-supervised Learning: This technique is between supervised where and unsupervised, some provided information is supervised but not necessarily all. A very relevant prerequisite is whether the distribution of examples, decoded with help from unlabeled data, will be relevant to the classification problem [33].

- Reinforcement Learning: The process of this technique is essentially when the machine learns by trial and error, and by this, it can predict and acquire rewards. That is a complex overview, due to on the computer field action we have long-term effects on future rewards [34].

- Evolutionary Learning: Through the knowledge that this kind of technique obtain and exploit, it develop the ability to upgrade itself. The algorithm works better when is applied to populations instead an individual systems. On evolutionary learning, to improve its performance, the training on a human-designed environment is significant [35].

- Deep Learning: Deep Learning is a subset of AI whose function is to try to imitate the behavior and functioning of the human brain, such as in data processing, pattern making and decision making based on their knowledge.[36]

Chapter 3

Development

This chapter describes the methodology used on this dissertation. It consists in the creation of a network topology to detect Malware in a network of the communications laboratory at Instituto Politécnico de Bragança (IPB), which is a controlled environment where this implementation was realized. All files can be found in the repository "https://drive.google.com/open?id=1IVRIhCctw2EV0mVTlGS3X2yETH5dtPhp".

Topology is the definition of how the systems are connected, what is the arrangement among the devices of the computer network that is developed [37]. The following sections explain why this topology was used, why these devices where chosen, how it was developed, also each device function and their operation mode.

The sequence of activities developed were the following:

- Step 1: Creating the Topology;
- Step 2: Host infecting with Malwares;
- Step 3: Centralization of the logs for the central server;
- Step 4: Treatment of raw data for noise reduction and improve classification;
- Step 5: Correlation between the data obtained;
- Step 6: Implementation of Machine Learning Algorithms for Automation in Analyzes;

3.1 Topology

The first step consists of creating a topology, with the objective of obtaining a complete and diversified database through the logs and alerts generated by the devices connected to the network.

The devices have different functions to cover the maximum of Malware behavior, enabling to find patterns in their activities, to make behavior analysis and to classify the acquired data with machine learning algorithms. In Figure 3.1 is possible to understand the connection between the devices used on the proposed topology. The table 3.1 shows the chosen implementation for each topology components.

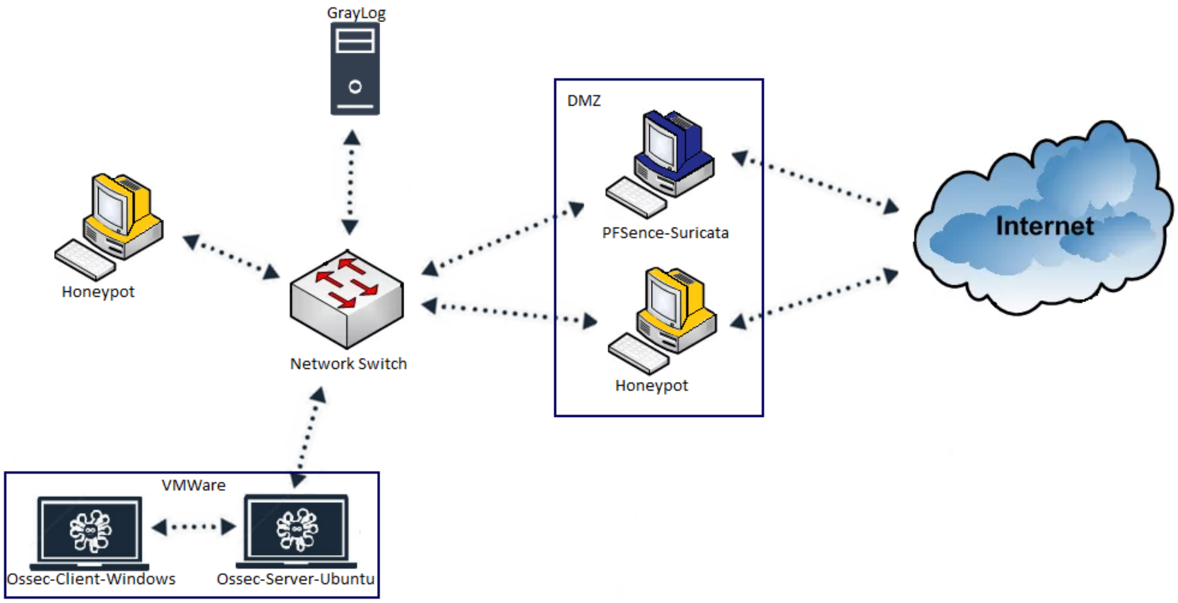


Figure 3.1: Proposed Network Topology

Table 3.1: Chosen implementation for each topology components.

Components	**Chosen Implementation**
NIDS	Suricata
HIDS	OSSEC-Server
HIDS	OSSEC-Client
Log Server	Graylog
Intern Honeypot	HONEYPI
Extern Honeypot	Dionaea

Each implementation is briefly characterized as follows:

- NIDS - Suricata

  The chosen NIDS was Suricata version 4.0.5, a new generation Open Source NIDS, which has a powerful detection motor with signature rules. If these rules are triggered, the network administrator is informed and the alert generated is sent to an e-mail or to a central server. Suricata is also compatible with many devices, having a unified platform, that helps interconnections.

  Suricata was installed on the gateway together to the Firewall Pf-Sense. This Firewall Pf-Sense was already configured on the network and Suricata is compatible with it. On this process, a copy of traffic was redirected to Suricata and then it performed analyzes. As the detection was made on a copy of the traffic no latency was added to the network.

- HIDS - OSSEC Client/Server

  The chosen HIDS was the OSSEC 2.9.9, based on a client-server system and it is Open Source. Its function is to for instance: actions inside a specific host, the files activities, monitor the integrity and the initialized processes. The OSSEC has a detection motor based on signatures, thus it can send the generated alerts by e-mail or Syslog, which is a tool for network devices to send alert messages to a logging server.

  - Ossec-Server was installed on a VMWare machine with distribution Ubuntu 16.04
  - Ossec-Client was installed on a VMWare machine with distribution Windows 10

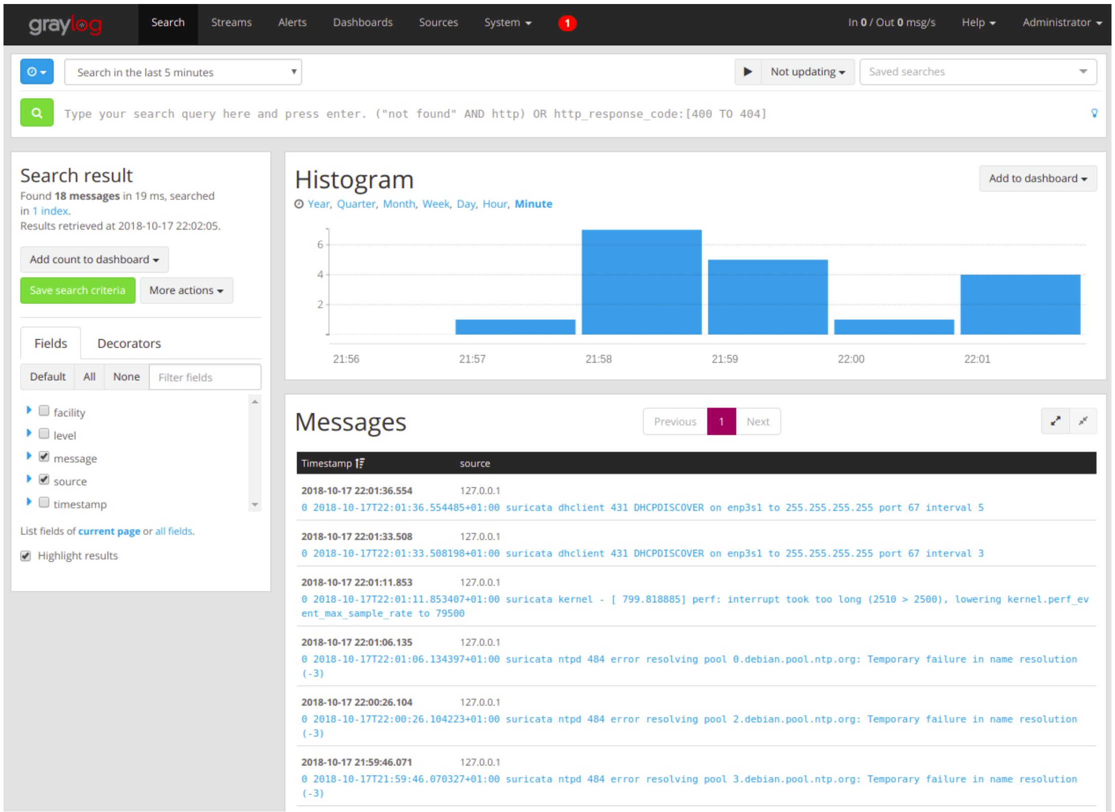
- Log Server - Graylog The log server used was Graylog 2, which was installed on a Debian Server. Graylog 2 is a highly interactive log server, which makes easier the visualization of registered logs and it is compatible with several devices and log syntaxes.

  On Graylog 2 it is possible to visualize data on graphics (Figure 3.2), is also possible to do a deep search with specific fields restrictions, as a specific IP, message name, among others. There is a graphic user interface, which allows users to interact with it.

- Inter Honeypot - HONEYPI

  The honeypot used on the intern network was the HoneyPi. this honeypot was chosen because the system was configured on a RaspberryPi, as in Figure 3.3, show that it is possible to obtain an efficient honeypot with a low-cost device. The HoneyPi is a low interaction honeypot, which registers the connection attempts as:

  - Port Scanning Activity;
  - Connection FTP attempt;
  - Connection Telnet attempt;
  - Connection VNC attempt;



Search result

Found 18 messages in 19 ms, searched in 1 index.
Results retrieved at 2018-10-17 22:02:05.

```
Timestamp                 source
2018-10-17 22:01:36.554   127.0.0.1
0 2018-10-17T22:01:36.554485+01:00 suricata dhclient 431 DHCPDISCOVER on enp3s1 to 255.255.255.255 port 67 interval 5

2018-10-17 22:01:33.508   127.0.0.1
0 2018-10-17T22:01:33.508198+01:00 suricata dhclient 431 DHCPDISCOVER on enp3s1 to 255.255.255.255 port 67 interval 3

2018-10-17 22:01:11.853   127.0.0.1
0 2018-10-17T22:01:11.853407+01:00 suricata kernel - [ 799.818885] perf: interrupt took too long (2510 > 2500), lowering kernel.perf_ev
ent_max_sample_rate to 79500

2018-10-17 22:01:06.135   127.0.0.1
0 2018-10-17T22:01:06.134397+01:00 suricata ntpd 484 error resolving pool 0.debian.pool.ntp.org: Temporary failure in name resolution
(-3)

2018-10-17 22:00:26.104   127.0.0.1
0 2018-10-17T22:00:26.104223+01:00 suricata ntpd 484 error resolving pool 2.debian.pool.ntp.org: Temporary failure in name resolution
(-3)

2018-10-17 21:59:46.071   127.0.0.1
0 2018-10-17T21:59:46.070327+01:00 suricata ntpd 484 error resolving pool 3.debian.pool.ntp.org: Temporary failure in name resolution
(-3)
```

Figure 3.2: Graylog 2

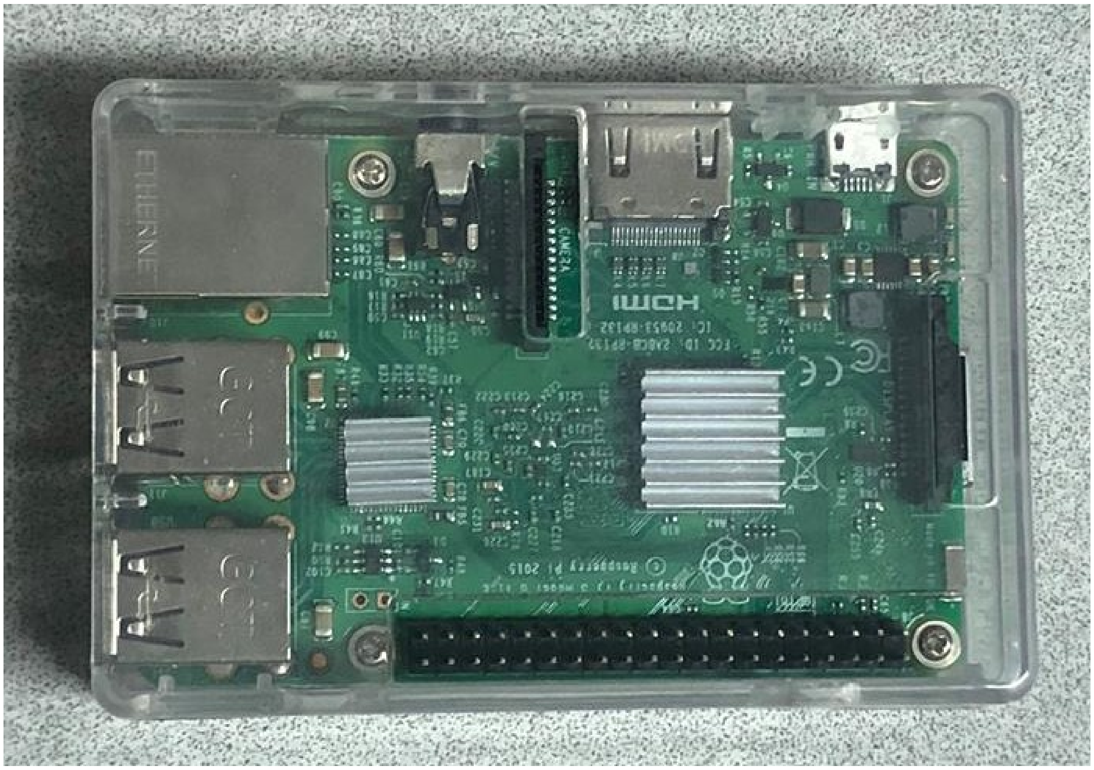


Figure 3.3: RaspberryPi

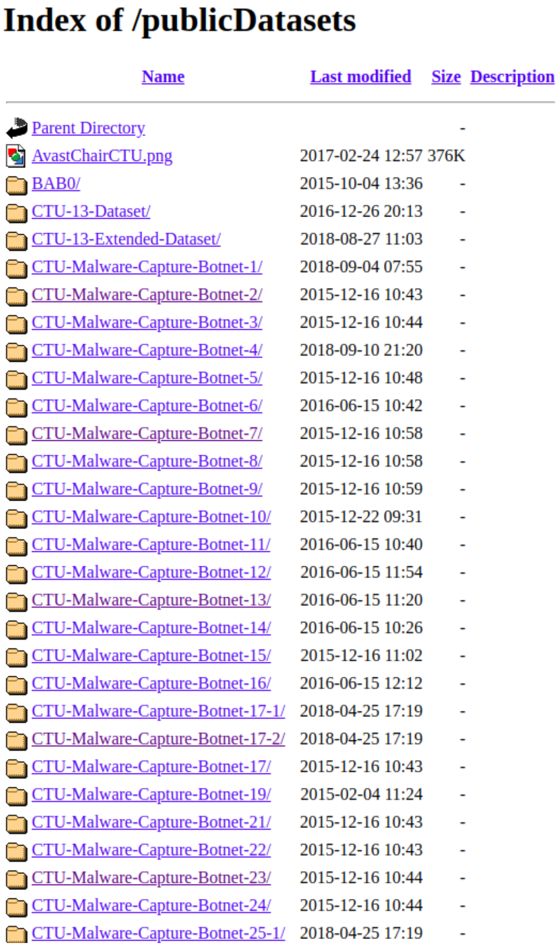
- Extern Honeypot - Dionaea

The extern honeypot was the Dionaea. It can emulate and provide services as Semi-Structured Query Language (SSQL), MY Structured Query Language (MYSQL), Point-to-Point Tunneling Protocol (PPTP), Session Initiation Protocol (SIP) among others. Dionaea also makes a copy of the binary of the malware, which has infected. It was installed on a WMWare machine with an Ubuntu 14.06 distribution.

3.2 Host infection

Step 2, consists on the infection of a Windows Virtual Machine, where the HIDS OSSEC-Client is installed. The malware was found on the [38] repository site you see in Figure 3.4, this repository has a huge amount of botnet binaries with more than three hundred samples.

The Malware used was available from a public repository, it could happen that the C&C server would be already known and, consequently, it may be disabled. Thus the virtual machine cannot obtain the answers from the requests to the C&C server. When an OSSEC-Client rule is triggered, an alert is sent, informing what is happening on the computer like a new resource installation or a service initialization.



Index of /publicDatasets

Name	Last modified	Size	Description
Parent Directory		-	
AvastChairCTU.png	2017-02-24 12:57	376K	
BAB0/	2015-10-04 13:36	-	
CTU-13-Dataset/	2016-12-26 20:13	-	
CTU-13-Extended-Dataset/	2018-08-27 11:03	-	
CTU-Malware-Capture-Botnet-1/	2018-09-04 07:55	-	
CTU-Malware-Capture-Botnet-2/	2015-12-16 10:43	-	
CTU-Malware-Capture-Botnet-3/	2015-12-16 10:44	-	
CTU-Malware-Capture-Botnet-4/	2018-09-10 21:20	-	
CTU-Malware-Capture-Botnet-5/	2015-12-16 10:48	-	
CTU-Malware-Capture-Botnet-6/	2016-06-15 10:42	-	
CTU-Malware-Capture-Botnet-7/	2015-12-16 10:58	-	
CTU-Malware-Capture-Botnet-8/	2015-12-16 10:58	-	
CTU-Malware-Capture-Botnet-9/	2015-12-16 10:59	-	
CTU-Malware-Capture-Botnet-10/	2015-12-22 09:31	-	
CTU-Malware-Capture-Botnet-11/	2016-06-15 10:40	-	
CTU-Malware-Capture-Botnet-12/	2016-06-15 11:54	-	
CTU-Malware-Capture-Botnet-13/	2016-06-15 11:20	-	
CTU-Malware-Capture-Botnet-14/	2016-06-15 10:26	-	
CTU-Malware-Capture-Botnet-15/	2015-12-16 11:02	-	
CTU-Malware-Capture-Botnet-16/	2016-06-15 12:12	-	
CTU-Malware-Capture-Botnet-17-1/	2018-04-25 17:19	-	
CTU-Malware-Capture-Botnet-17-2/	2018-04-25 17:19	-	
CTU-Malware-Capture-Botnet-17/	2015-12-16 10:43	-	
CTU-Malware-Capture-Botnet-19/	2015-02-04 11:24	-	
CTU-Malware-Capture-Botnet-21/	2015-12-16 10:43	-	
CTU-Malware-Capture-Botnet-22/	2015-12-16 10:43	-	
CTU-Malware-Capture-Botnet-23/	2015-12-16 10:44	-	
CTU-Malware-Capture-Botnet-24/	2015-12-16 10:44	-	
CTU-Malware-Capture-Botnet-25-1/	2018-04-25 17:19	-	

Figure 3.4: Repository Site "www.stratosphereips.org"

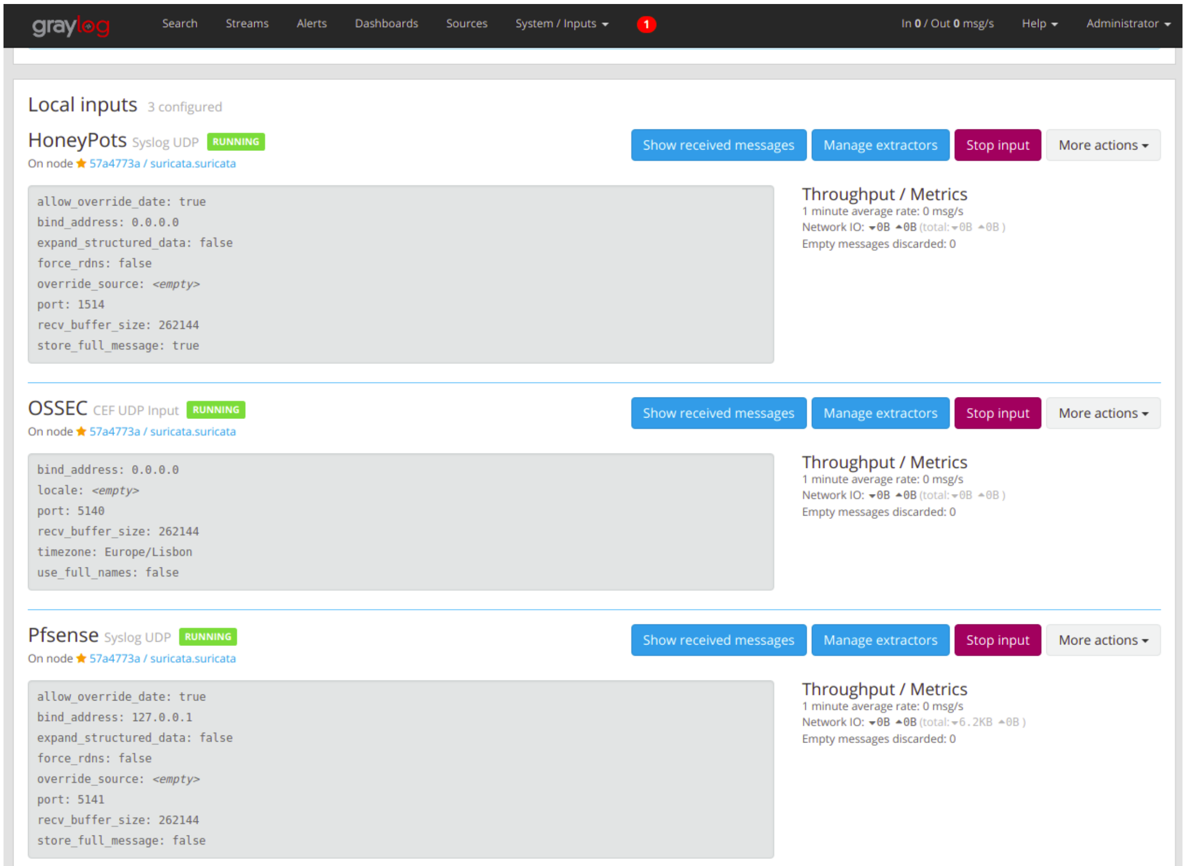
3.3 Logs Centralization

Step 3 consists on logs centralization. After the alert is sent, the messages are redirected to the OSSEC-Server, that is responsible for saving the alerts generated by OSSEC-client and sending them to Graylog. OSSEC-server is also as a possible victim for the botnet to infect and perform its actions.

Once that the virtual machine which has the OSSEC-Client is infected, it may scan the network and try to infect other devices to the botnet. HoneyPi waits for this infections to register all connection attempts and send the logs to the Graylog server. When the network scanner happens, HoneyPi sends its alerts, informing about this activity.

Dionaea, which is the Honeypot on the external network, search for activity from malware that may try to connect to the IPB network, and when it receives an attack it logs the messages to the Graylog server.

Suricata remains on the network with the PfSense, receiving a copy of traffic to perform its analyzes. Once that a Suricata rule is triggered, it sends an alert and the copy of original PfSense traffic directly to the Graylog. After sending the logs, the Graylog receives of data from all the devices simultaneously, as represented in the Figure 3.5



Local inputs 3 configured

HonеyPots Syslog UDP RUNNING

On node ★ 57a4773a / suricata.suricata

```
allow_override_date: true
bind_address: 0.0.0.0
expand_structured_data: false
force_rdns: false
override_source: <empty>
port: 1514
recv_buffer_size: 262144
store_full_message: true
```

Throughput / Metrics
1 minute average rate: 0 msg/s
Network IO: ▾0B ▴0B (total: ▾0B ▴0B)
Empty messages discarded: 0

OSSEC CEF UDP Input RUNNING

On node ★ 57a4773a / suricata.suricata

```
bind_address: 0.0.0.0
locale: <empty>
port: 5140
recv_buffer_size: 262144
timezone: Europe/Lisbon
use_full_names: false
```

Throughput / Metrics
1 minute average rate: 0 msg/s
Network IO: ▾0B ▴0B (total: ▾0B ▴0B)
Empty messages discarded: 0

Pfsense Syslog UDP RUNNING

On node ★ 57a4773a / suricata.suricata

```
allow_override_date: true
bind_address: 127.0.0.1
expand_structured_data: false
force_rdns: false
override_source: <empty>
port: 5141
recv_buffer_size: 262144
store_full_message: false
```

Throughput / Metrics
1 minute average rate: 0 msg/s
Network IO: ▾0B ▴0B (total: ▾6.2KB ▴0B)
Empty messages discarded: 0

Figure 3.5: Graylog 2 - Analyzes of data from the devices

3.4 Data Analysis

Step 4 is the phase where methods and procedures were executed for preparing the data for analysis. From August sixteen to September nine of 2018, the network traffic was stored in Packet Capture (.pcap) files from the host where the OSSEC client was installed, for future analysis. A total of 1.089.387 suricata alerts were acquired as is shown in the Figure 3.6 and in the Table 3.2. These data are the records of all alerts that it provided on the internal network traffic.

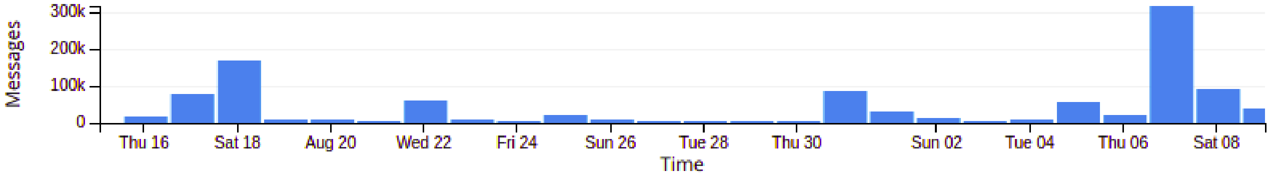


Figure 3.6: Suricata alerts

The OSSEC data client and server registered a total of 6.160 alerts, as seen in the Figure 3.7. These alerts are the records of the activities that the hosts suffered in the monitored time interval, ans the messages is shown in Table 3.3.

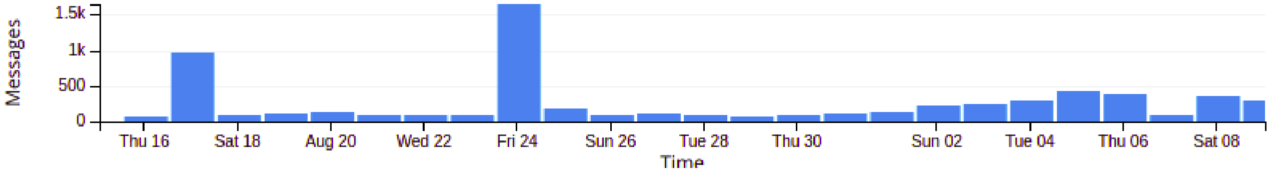


Figure 3.7: Ossec Alerts

The data from the internal honeypot were recorded a total of 30.833 alerts, seen in the Figure 3.8 and in Table 3.4.

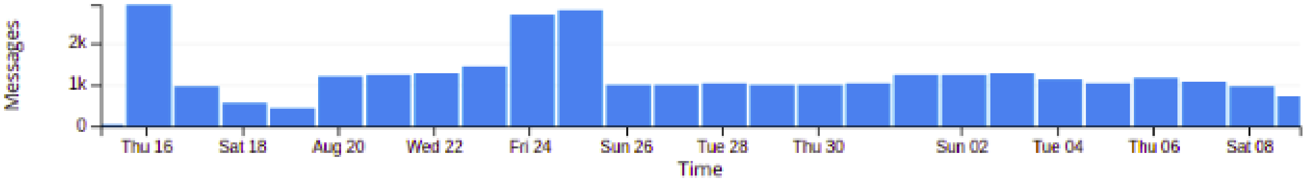


Figure 3.8: HoneyPi Alerts

The data from the external honeypot were recorded a total of 12.803 alerts, as seen in Figure 3.9 and Table 3.5.

Table 3.2: Exemple Suricata Alerts

TIMESTAMP	SOURCE	MESSAGE
2018-09-06T12:52:42.400Z	127.0.0.1	0 2018-09-06T13: 52: 42 + 01: 00 dmz-lc.estig.ipb.pt suricata 89535 [1: 2019980: 3] ET POLITICAL Possible IP Check myexternalip.com [Classification: Possible Corporate Privacy Breach] [Priority : 1] {TCP} 192.168.0.162:59137 ->78.47.139.102:80
2018-09-01T00:43:57.698Z	127.0.0.1	0 2018-09-01T01: 43: 57 + 01: 00 dmz-lc.estig.ipb.pt suricata11939 [1: 2402000: 4924] DROP E DROP Blocked Listing Source Group 1 [Classification: Mixed Attack] [Priority: 2] {TCP} 89.248.174.55:64348 ->193.136.195.94:34
2018-09-01T00:44:29.332Z	127.0.0.1	1 0 2018-09-01T01:44:29.332464+01:00 suricata dhclient 473 DHCPDISCOVER on enp3s0 to 255.255.255.255 port 67 interval 6
2018-09-01T00:44:35.502Z	127.0.0.1	0 2018-09-01T01:44:35.502554+01:00 suricata dhclient 473 DHCPDISCOVER on enp3s0 to 255.255.255.255 port 67 interval 9
2018-09-01T00:44:44.505Z	127.0.0.1	0 2018-09-01T01:44:44.504739+01:00 suricata dhclient 473 DHCPDISCOVER on enp3s0 to 255.255.255.255 port 67 interval 12
2018-09-01T00:45:06.192Z	127.0.0.1	0 2018-09-01T01: 45: 06 + 01: 00 dmz-lc.estig.ipb.pt suricata 11939 [1: 2402000: 4924] DROP ET DROP Blocked Listing Source Group 1 [Classification: Mixed Attack] [Priority: 2] {TCP} 176.119.7.26:55028 ->193.136.195.94:63017
2018-09-01T00:45:27.836Z	127.0.0.1	0 2018-09-01T01: 45: 27.836316 + 01: 00 suricata dhclient 473 DHCPDISCOVER on enp3s0 a 255.255.255.255 port 67 interval 3
2018-09-06T12:53:15.286Z	127.0.0.1	0 2018-09-06T13:53:15.286566+01:00 suricata dhclient 472 DHCPDISCOVER on enp3s1 to 255.255.255.255 port 67 interval 7

Table 3.3: Exemple Ossec Alerts

TIMESTAMP	SOURCE	MESSAGE
2018-09-01T00:53:39.667Z	127.0.0.1	0 2018-09-01T01:53:39+01:00 dmz-lc.estig.ipb.pt dhcpd - DHCPREQUEST for 192.168.0.184 from 00:0c:29:4c:5a:af (ossec-virtual-machine) way in 1
2018-09-01T00:53:39.673Z	127.0.0.1	0 2018-09-01T01:53:39+01:00 dmz-lc.estig.ipb.pt dhcpd - DHCPACK on 192.168.0.184 to 00:0c:29:4c:5a:af (ossec-virtual-machine) way in 1
2018-09-01T02:07:10.000Z	ossec-virtual-machine	OSSEC HIDS: [18107, 3] Windows Logon Success
2018-09-06T13:10:04.000Z	ossec-virtual-machine	OSSEC HIDS: [18103, 5] Windows error event.
2018-09-01T02:43:24.611Z	127.0.0.1	0 2018-09-01T03:43:24+01:00 dmz-lc.estig.ipb.pt dhcpd - DHCPREQUEST for 192.168.0.184 from 00:0c:29:4c:5a:af (ossec-virtual-machine) way in 1
2018-09-01T02:43:24.611Z	127.0.0.1	0 2018-09-01T03:43:24+01:00 dmz-lc.estig.ipb.pt dhcpd - DHCPACK on 192.168.0.184 to 00:0c:29:4c:5a:af (ossec-virtual-machine) way in 1
2018-09-01T03:42:50.544Z	127.0.0.1	0 2018-09-01T04:42:50+01:00 dmz-lc.estig.ipb.pt dhcpd - DHCPACK on 192.168.0.184 to 00:0c:29:4c:5a:af (ossec-virtual-machine) way in 1

Table 3.4: Example HoneyPi Alerts

TIMESTAMP	SOURCE	MESSAGE
2018-09-01T00:48:43.000Z	HoneyPi	HoneyPi kernel: [1605127.513264] IN=eth0 OUT= MAC=ff:ff:ff:ff:ff:ff:00:0c:29:72:3c:ec:08:00:45:00:00:e5:47:82:00:00:80:11:6f:94 SRC=192.168.0.162 DST=192.168.0.255 LEN=229 TOS=0x00 PREC=0x00 TTL=128 ID=18306 PROTO=UDP SPT=138 DPT=138 LEN=209
2018-09-01T00:51:34.000Z	HoneyPi	HoneyPi kernel: [1605297.907795] Under-voltage detected! (0x00050005)
2018-09-01T00:52:25.000Z	HoneyPi	HoneyPi kernel: [1605348.931823] IN=eth0 OUT= MAC=01:00:5e:00:00:fb:00:0c:29:72:3c:ec:08:00:45:00:00:69:3f:66:00:00:01:11:d7:d8 SRC=192.168.0.162 DST=224.0.0.251 LEN=105 TOS=0x00 PREC=0x00 TTL=1 ID=16230 PROTO=UDP SPT=5353 DPT=5353 LEN=85
2018-09-01T00:54:39.000Z	HoneyPi	HoneyPi kernel: [1605483.028480] Voltage normalised (0x00000000)
2018-09-01T00:54:33.000Z	HoneyPi	HoneyPi kernel: [1605476.788442] Under-voltage detected! (0x00050005)
2018-09-01T00:56:33.000Z	HoneyPi	HoneyPi kernel: [1605597.428885] Under-voltage detected! (0x00050005)
2018-09-01T00:59:38.000Z	HoneyPi	HoneyPi kernel: [1605782.549562] Voltage normalised (0x00000000)
2018-09-01T01:02:07.000Z	HoneyPi	HoneyPi kernel: [1605931.188601] IN=eth0 OUT= MAC=b8:27:eb:ba:2f:e3:00:0c:29:55:35:40:08:00:45:00:00:4c:00:00:40:00:38:11:ad:12 SRC=195.22.17.7 DST=192.168.0.201 LEN=76 TOS=0x00 PREC=0x00 TTL=56 ID=0 DFPROTO=UDP SPT=123 DPT=59111 LEN=56

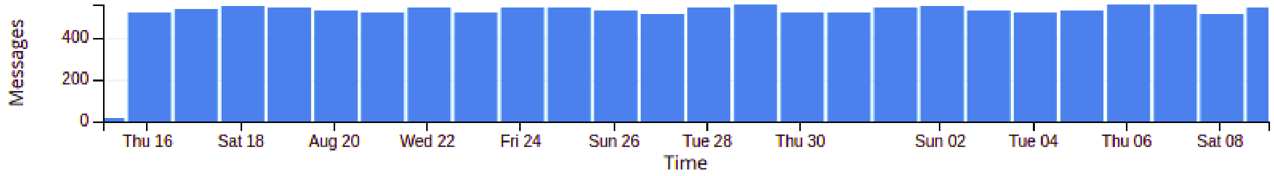


Figure 3.9: Example Dionaea Alerts

The data sent to the Graylog usually uses pre-defined formats by their systems, making

Table 3.5: Example Dionaea Logs

TIMESTAMP	SOURCE	MESSAGE
2018-09-01T01:17:01.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine CRON [10545]:pam_unix(cron:session): session opened for user root by (uid=0)
2018-09-01T01:23:19.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine dhclient: DHCPACKof 192.168.0.203 from 192.168.0.254
2018-09-01T01:23:19.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine Network Manager[962]: (eth0): DHCPv4 state changed renew ->renew
2018-09-01T01:23:19.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine Network Manager[962]: domain search 'estig.ipb.pt.'
2018-09-01T01:23:19.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine Network Manager[962]: domain search 'ipb.pt.'
2018-09-01T01:23:19.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine dbus[721]: [system]Activating service name= 'org.freedesktop.nm_dispatcher' (using servicehelper)
2018-09-01T01:23:19.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine Network Manager[962]: domain name 'labcom.estig.ipb.pt'
2018-09-01T01:23:19.000Z	sysadmin-virtual-machine	sysadmin-virtual-machine dhclient: DHCPREQUEST of 192.168.0.203 on eth0 to 192.168.0.254 port 67 (xid=0x462bee02)

them different from each other as can be seen on the following example.

- Suricata: 2018-09-06T12:52:42.400Z 127.0.0.1 0 2018-09-06T13:52:42+01:00 dmz-lc.estig.ipb.pt suricata 89535 [1:2019980:3] ET POLICY Possible IP Check myexternalip.com [Classification: Potential Corporate Privacy Violation] [Priority: 1] TCP 192.168.0.162:59137 -> 78.47.139.102:80

- OSSEC: 2018-09-06T16:50:06.000Z ossec-virtual-machine OSSEC HIDS: [18107, 3] Windows Logon Success.

- HoneyPi: 2018-09-01T08:30:42.000Z HoneyPi kernel: [1632845.991561] IN=eth0 OUT= MAC=b8:27:eb:ba:2f:e3:00:0c:29:55:35:40:08:00:45:00:01:6b:ed: 88:00:00:40:11: 08:e2 SRC=192.168.0.254 DST=192.168.0.201 LEN=363 TOS=0x00 PREC= 0x00 TTL=64 ID=60808 PROTO=UDP SPT=67 DPT=68 LEN=343

- Dionaea: 2018-09-01T01:23:19.000Z sysadmin-virtual-machine dhclient: DHCPREQUEST of 192.168.0.203 on eth0 to 192.168.0.254 port 67 (xid=0x462bee02)

It is necessary to standardize the data to enable further analysis, such as correlation between events. For the pertinent information of the captured data, an algorithm was applied to subdivided the original message into 11 columns with a total of 1.022.493, as seen in Table 3.6. Messages that did not meet the required characteristics, such as system errors and Dynamic Host Configuration Protocol (DHCP) server requests, were also removed.

Table 3.6: Transformation of suricata messages

Field	**Description**
Source	Source (system) that generated the alert
Date	Date of the alert
Time	Time of the alert
Message	Message of suricata's alert, specifying the rule that triggered the alert
Classification	Alert Classification
Priority	Priority Level. As low is the value higher is the priority
Protocol	Protocol of the packet that triggered suricata's rule
Source IP	Source IP of the packet that triggered suricata's rule
Destination IP	Destination IP of the packet that triggered suricata's rule
Source Port	Source Port of the packet that triggered suricata's rule
Destination Port	Destination Port of the packet that triggered suricata's rule

Ossec messages were subdivided into source, date, time and message, being shown in

Table 3.8: Example of OSSEC results

Source	Date	Time	message
Ossec	2018-08-26	22:52:38.000	Windows Logon Success
Ossec	2018-08-26	06:36:26.000	Log file rotated
Ossec	2018-08-26	16:07:55.000	System time changed
Ossec	2018-08-16	22:03:25.000	Windows Logon Success
Ossec	2018-08-26	23:03:32.000	Windows Logon Success
Ossec	2018-08-17	03:53:06.000	Login session opened
Ossec	2018-08-20	23:21:18.000	Registry Entry Added to the System
Ossec	2018-08-20	23:21:18.000	Registry Entry Added to the System
Ossec	2018-08-27	21:52:36.000	Windows Logon Success

Table 3.7 and in Table 3.8 is shown an example of the result, the total of alerts are 5295.

Table 3.7: Transform Ossec data

Field	**Description**
Source	Source (system) that generated the alert
Date	Date of the alert
Time	Time of the alert
Message	Message of Ossec's alert, specifying the rule that triggered the alert

The messages of HoneyPi were divided into 6 columns, as described on Table 3.9, with a total of 14818 alerts.

Table 3.9: Transform HoneyPi Data

Field	**Description**
Source	Source (system) that generated the alert
Date	Date of the alert
Time	Time of the alert
Source IP	Source IP of system registered in log file
Destination IP	Destination IP of system registered in log file
Protocol	Protocol used during the communication that was registered

The messages from Dionaea were subdivided into 5 columns and can be seen on Table 3.10.

Table 3.10: Transform Dionaea Data

Field	**Description**
Source	Source (system) that generated the alert
Date	Date of the alert
Time	Time of the alert
Source IP	Source IP of system registered in log file
Destination IP	Destination IP of system registered in log file

3.5 Correlation

Step 5 is the correlation analysis between the obtained data to find if some of the activities found are caused by the same event. It was decided to analyze 3 correlations:

- First: Suricata x OSSEC
- Second: Suricata x Honeypot (Honeypi)
- Third: OSSEC x Honeypot (Honeypi)

Tthe first was to correlate Suricata data with OSSEC. After all the data were processed, the parameters used for this correlation were Date and Time. An algorithm was developed for this correlation where, it searches first the dates in the fields to filter by day, then the same thing was done with hour and minutes. In the field of seconds an interval of 2 seconds was stipulated for incorporating some delay in the alarm trigger time or associated execution. A total of 3112 correlated alerts with the stipulated parameters are shown in the Table 3.11.

Table 3.11: Correlation data

Field	**Description**
Date	Alert Date
Suricata Time	Time of suricata's alert
OSSEC Time	Time of OSSEC's alert
Suricata Message	Message of suricata's alert, specifying the rule that triggered the alert
OSSEC Message	Message of Ossec's alert, specifying the rule that triggered the alert
Classification	Suricata's alert Classification
Source IP	Source IP of the packet that triggered suricata's rule
Destination IP	Destination IP of the packet that triggered suricata's rule
Source Port	Source Port of the packet that triggered suricata's rule
Destination Port	Destination Port of the packet that triggered suricata's rule

The honeypot data were submitted to the same analysis parameters as Suricata and OSSEC data, with the interval of 2 seconds between the logs, but no correlation was found between them. Possible reasons for this event will be discussed further.

3.6 Implementation of Machine Learning Algorithms

Step 6 is the last step in data analysis step. In this section we explain the application of classification algorithms over the data from Suricata and OSSEC alerts. Since was not possible to know exactly what is the predominant characteristic for botnet detection, was opted for the implementation of unsupervised learning algorithms to find patterns.

For this classification, the k-means algorithm was used. It is used when we do not know the classification of the object. It uses the method of grouping (Cluntering) for this classification of objects.

A cluster is a group of data, where each cluster has similarities with each other. The k-means algorithm uses these similarities to group the data into clusters, where the number of centroids (central points of the groups) is equal to the number of clusters as seen in the Figure 3.10.

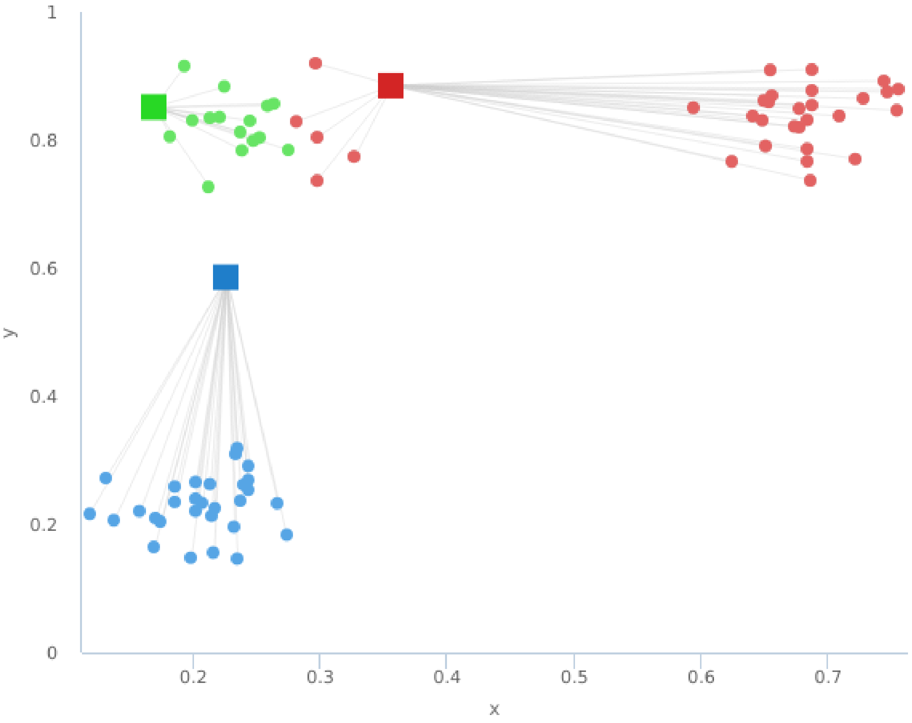


Figure 3.10: Example K-means

When the first interaction of the algorithm occurs, the average distance of all objects between the centroids is calculated. The centroids are positioned in the center of the objects belonging to each centroid. This interaction can occur changes in the centroids and objects, depicted in Figure 3.11.

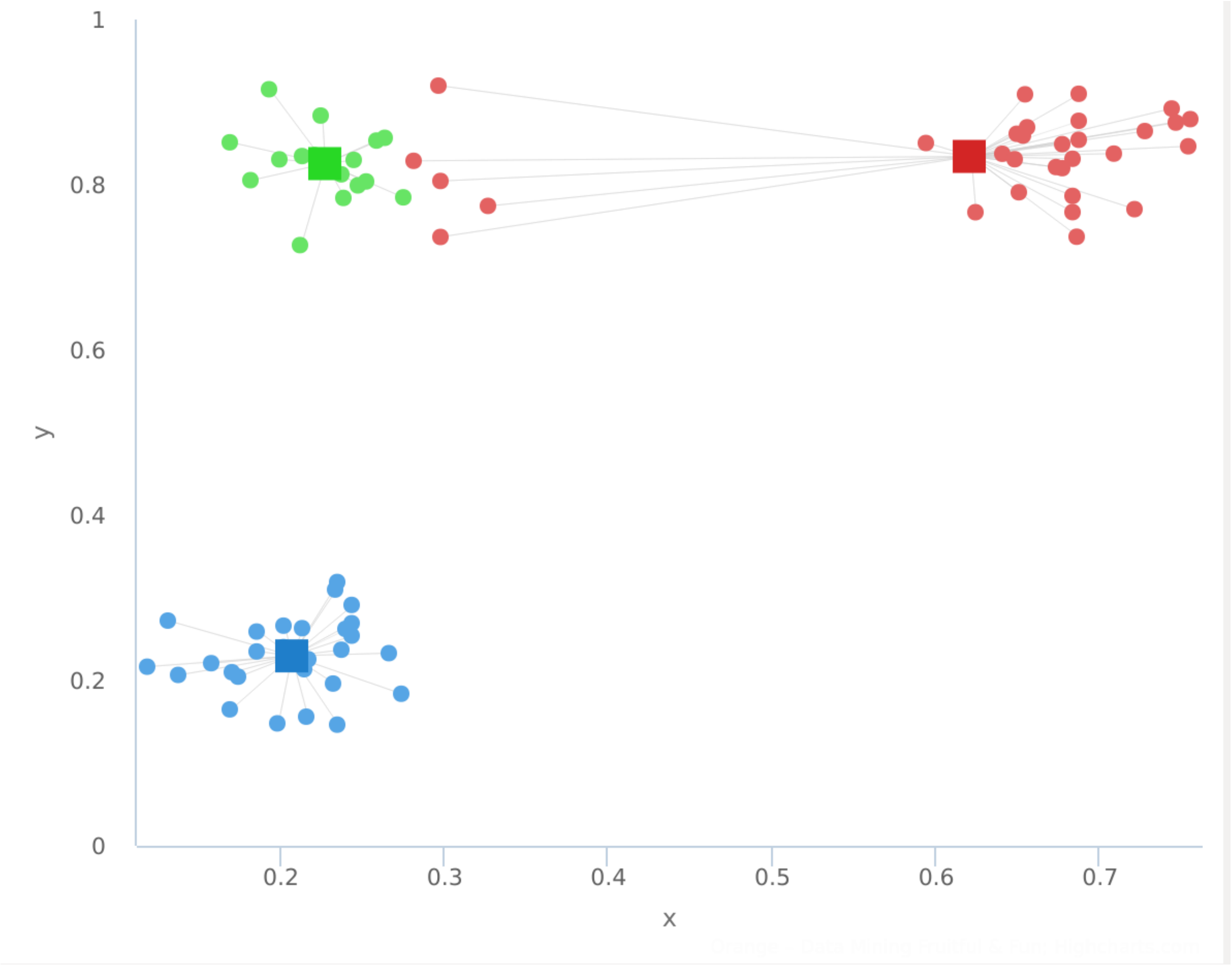


Figure 3.11: Example K-means after the first interaction

This process occurs N times where, N being the number of times the user stipulates. At the end of the process, the clusters were in the center of the objects of their respective classes as in Figure 3.12.

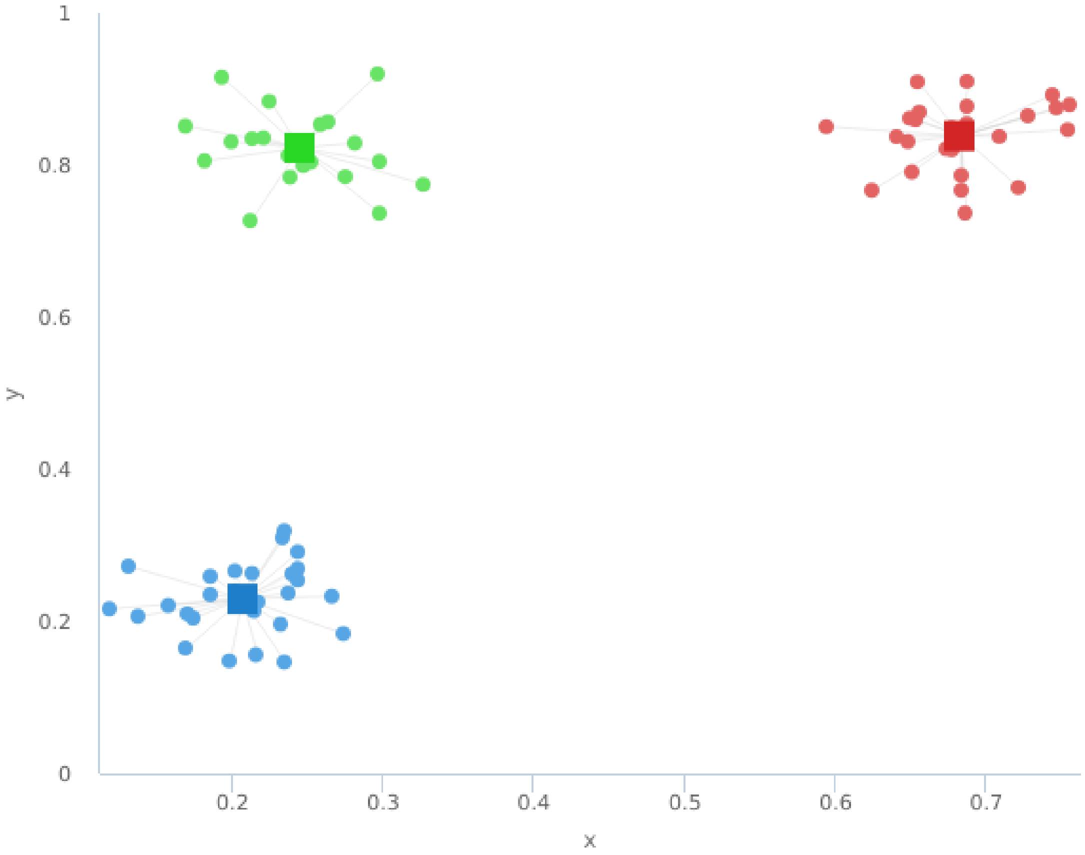


Figure 3.12: K-means completed

The software used for the implementation of k-means was Orange-canvas, with visual programming without the need for coding, an open source software with very diversified interactivity and simplicity of use [39]. Figure 3.13 shows the assembled architecture for the analysis of the data and implementation of the algorithms.

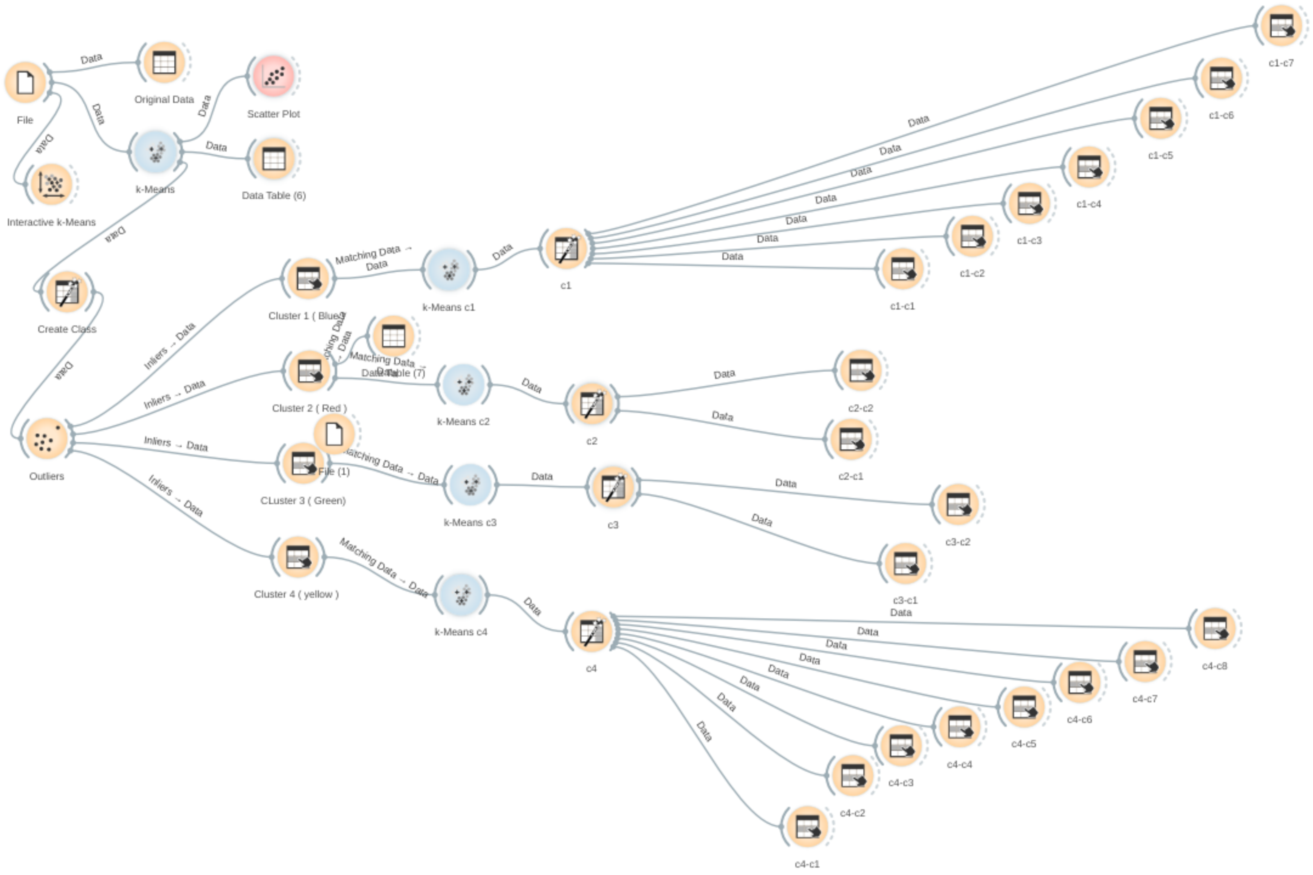
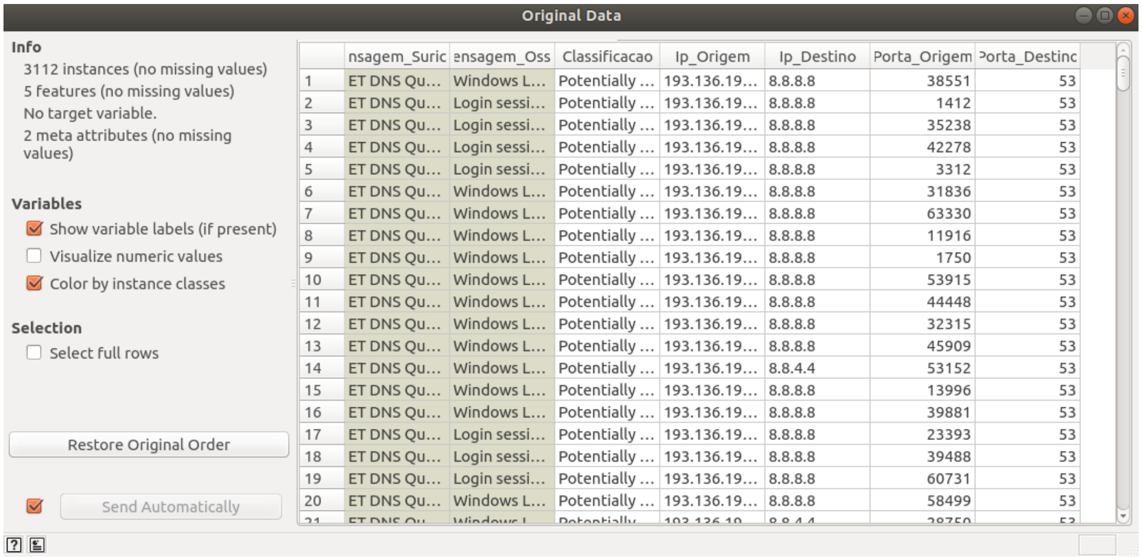


Figure 3.13: Topology Orange-Canvas

The sequences used in actions on orange-canvas are as follows.

The first step is to import the CSV file with the data to be analyzed, this data is the correlation between the Suricata and OSSEC. The features chosen for the analyzes are the Suricata Alert Classification, the source IP, the destination IP, the Source port, and the Destination port. As the orange canvas works with drag and drop, it is only necessary to connect to other components of the software.

After importing the file it is possible to see the data loaded with the Data Table component seen in Figure 3.14.



Info

3112 instances (no missing values)
5 features (no missing values)
No target variable.
2 meta attributes (no missing values)

Variables

☑ Show variable labels (if present)
☐ Visualize numeric values
☑ Color by instance classes

Selection

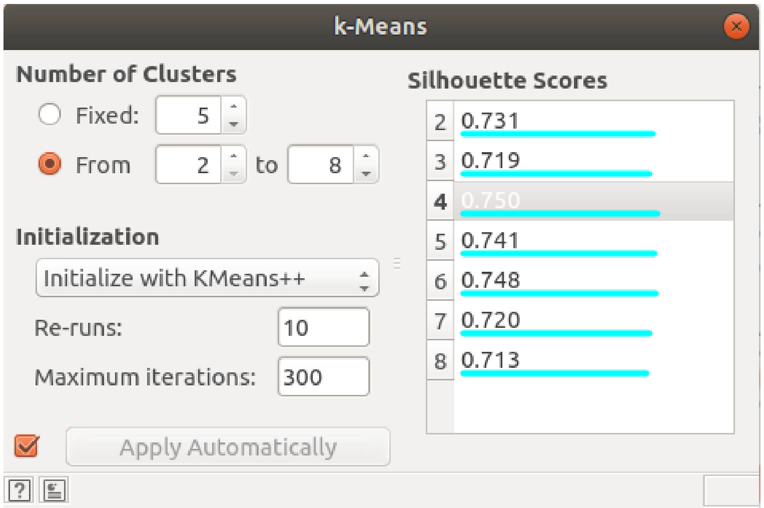
☐ Select full rows

	nsagem_Suric	ensagem_Oss	Classificacao	Ip_Origem	Ip_Destino	Porta_Origem	Porta_Destinc
1	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	38551	53
2	ET DNS Qu...	Login sessi...	Potentially ...	193.136.19...	8.8.8.8	1412	53
3	ET DNS Qu...	Login sessi...	Potentially ...	193.136.19...	8.8.8.8	35238	53
4	ET DNS Qu...	Login sessi...	Potentially ...	193.136.19...	8.8.8.8	42278	53
5	ET DNS Qu...	Login sessi...	Potentially ...	193.136.19...	8.8.8.8	3312	53
6	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	31836	53
7	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	63330	53
8	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	11916	53
9	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	1750	53
10	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	53915	53
11	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	44448	53
12	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	32315	53
13	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	45909	53
14	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.4.4	53152	53
15	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	13996	53
16	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	39881	53
17	ET DNS Qu...	Login sessi...	Potentially ...	193.136.19...	8.8.8.8	23393	53
18	ET DNS Qu...	Login sessi...	Potentially ...	193.136.19...	8.8.8.8	39488	53
19	ET DNS Qu...	Login sessi...	Potentially ...	193.136.19...	8.8.8.8	60731	53
20	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.8.8	58499	53
21	ET DNS Qu...	Windows L...	Potentially ...	193.136.19...	8.8.4.4	28750	53

Figure 3.14: Original Data Table

The second step is the application of the K-means algorithm. The features chosen for the grouping are the source ports and destination ports.

For implementation of K-means is needed to choose the number of clusters for its classification. The software Orange-canvas has a function to apply the k-means and calculates the most appropriate number of cluster. For our scenario the quantity chosen was 4, as shown in Figure 3.15.



Number of Clusters

○ Fixed: 5
● From 2 to 8

Initialization

Initialize with KMeans++

Re-runs: 10

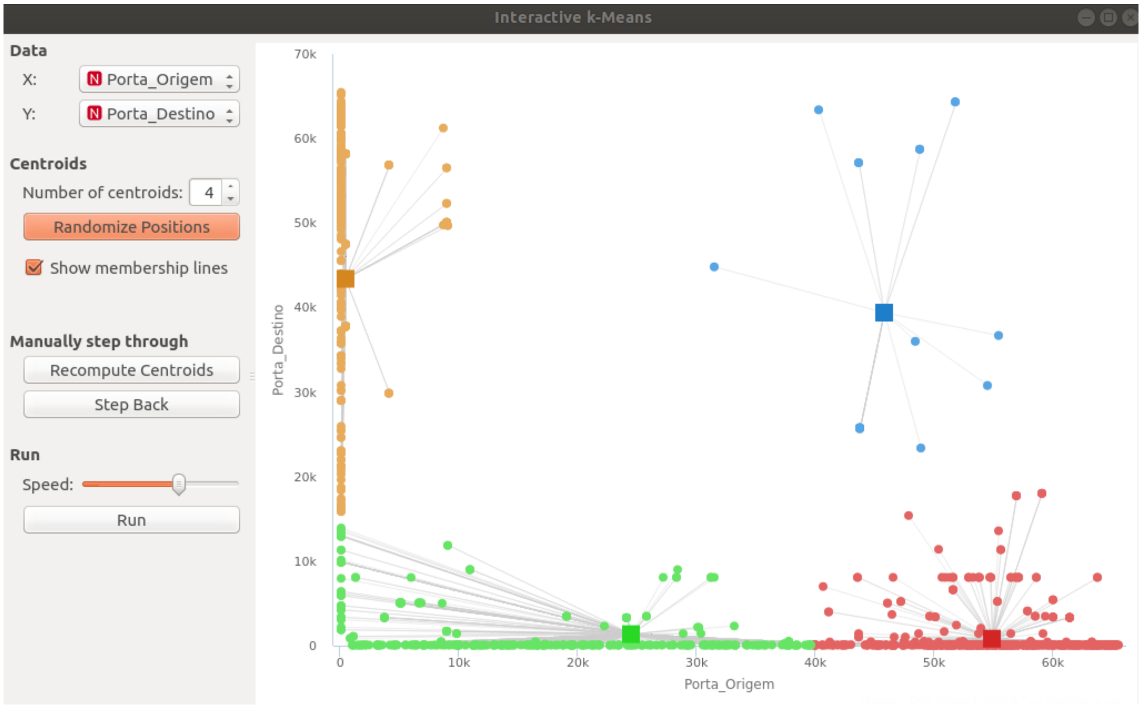
Maximum iterations: 300

Silhouette Scores

2	0.731
3	0.719
4	0.750
5	0.741
6	0.748
7	0.720
8	0.713

Figure 3.15: Clusters Orange-Canvas

The interactive k-means component allows visualization of the implementation of the algorithm, seen in Figure 3.16, together with the steps that it takes to implement.



Data

X: N Porta_Origem

Y: N Porta_Destino

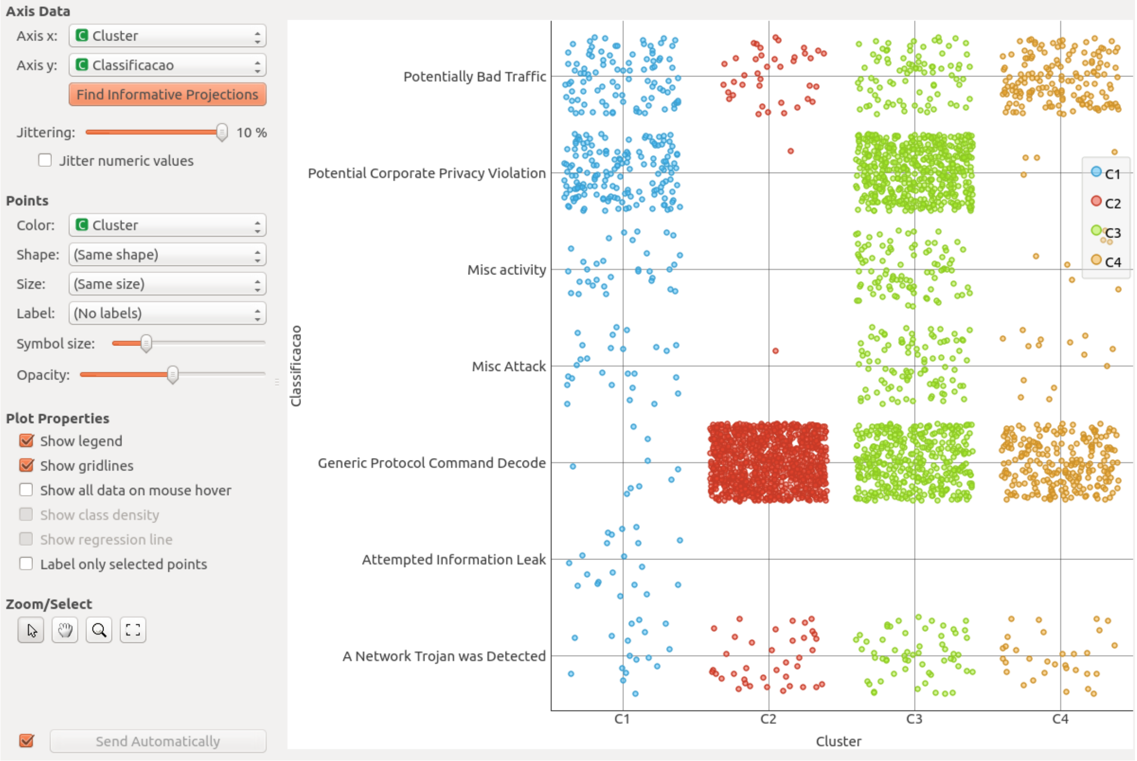
Centroids

Number of centroids: 4

☑ Show membership lines

Figure 3.16: Interactive k-Means

After K-means is performed in the Scatter Plot component it is possible to see the features and other attributes, such as the comparison of each cluster with the alert classification, as in Figure 3.17



Axis Data

Axis x: Cluster

Axis y: Classificacao

Jittering: 10 %

☐ Jitter numeric values

Points

Color: Cluster

Shape: (Same shape)

Size: (Same size)

Label: (No labels)

Symbol size:

Opacity:

Plot Properties

☑ Show legend

☑ Show gridlines

☐ Show all data on mouse hover

☐ Show class density

☐ Show regression line

☐ Label only selected points

Zoom/Select

☑ Send Automatically

Find Informative Projections
Potentially Bad Traffic
Potential Corporate Privacy Violation
Misc activity
Misc Attack
Generic Protocol Command Decode
Attempted Information Leak
A Network Trojan was Detected
Classificacao
C1
C2
C3
C4
Cluster
C1
C2
C3
C4

Figure 3.17: Scatter Plot Classification x Cluster

The third step was the creation of a variable class with the created clusters, with the Create Class component. It was opted to use this component due to the fact that later on the decision tree algorithm will need a variable class for its execution.

The fourth step are the outliers. They are data that belong to a certain cluster but are far from the centroids. Due to this dispersion they can be errors in the algorithm or isolated actions. For this, we use the Outliers component, then classifies the components as similar or different from the main class. Data that is considered similar falls into the inliers category and or others into the outliers.

After the outliers are removed, it is possible to see the result in the Data Table (2) component.

The fifth Step is the separation of each cluster for a more detailed analysis. For this separation is used the component Select Rows, that uses rules that will be applied in each line and the obtained results are redirected.

The Sixth Step is the application of the K-means algorithm again in each cluster, subdividing the characteristics found to find better patterns.

- Cluster 1: In cluster 1 the number of clusters for the second application of k-means was a total of 7.
- Cluster 2: In cluster 2 the number of clusters for the second application of k-means was a total of 2.
- Cluster 3: In cluster 3 the number of clusters for the second application of k-means was a total of 2.
- Cluster 4: In cluster 4 the number of clusters for the second application of k-means was a total of 8.

The results found in the subdivisions will be presented in the next chapter.

Chapter 4

Results

In this chapter are discussed the results obtained in the machine learning pixes described on, the previous chapter, considering the Outliers and Inliers.

4.1 Inliers

The first results presented are all the clusters and sub-clusters found from the K-means algorithm. After the first application of the algorithm, it was possible to observe the creation of 4 main clusters. Its main features are a range of source ports and destination ports, which can be viewed in the Figure 4.1.

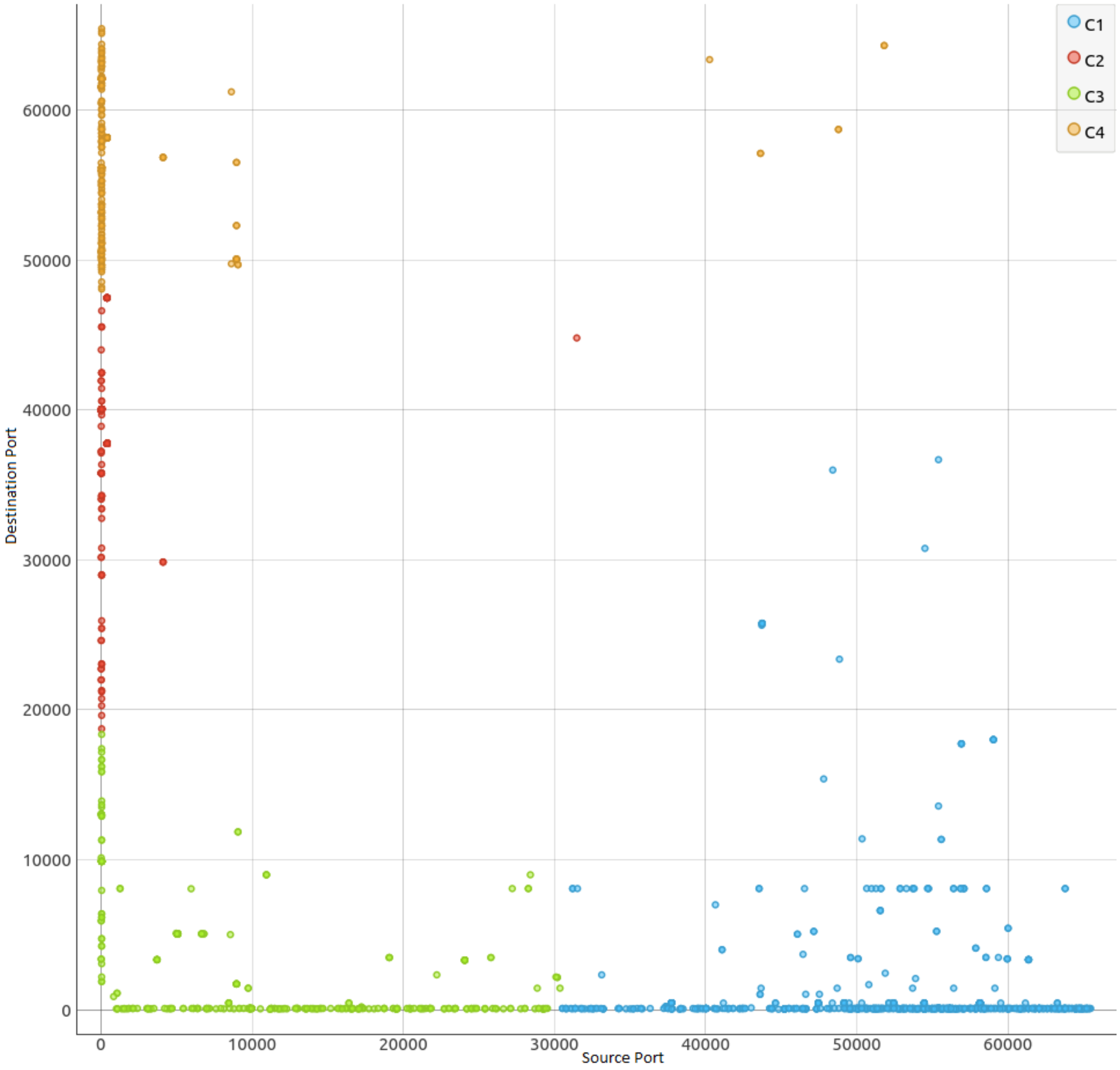


Figure 4.1: Source Port x Destination Port

Follows an individually analysis of each cluster, commenting its characteristics and actions. In tables will be shown the classifications of the alert, the source IP, destination IP, source port and destination port.

The classification of alerts is divided into 7 categories:

- A Network Trojan was Detected - this alert is triggered when a known malware connection is detected on the network.
- Attempted Information Leak - handles signatures of potentially harmful information

collection attempts. Information leaks or acknowledgment attacks that are classified as leaked, attempted data are not positive evidence that an attempt to collect information was successful. Instead, they are a sign that an attempt has been made.

- Generic Protocol Command Decode - Checks for packets that are not decode as the standard specifies.
- Misc Attack - Diverse attacks known by IDS, cataloged and blacklisted.
- Misc Activity - diversified activity found in the network, such as anomalies generating a large number of false positives.
- Potential Corporate Privacy Violation - This alert is triggered when any activity that has as a relation with the violation of privacy as collection of information of the user, IP address valid.
- Potentially Bad Traffic - This alert is triggered when network traffic is potentially malicious as connections from command and control servers to known botnets.

In the first implementation of the K-means a total of 4 chuster was obtained, and in each of the clusters the K-means was applied again for an in-depth analysis of the data.

4.1.1 Cluster 1

As depicted on Figure 4.1, Cluster 1 is displayed in blue with a total of 743 alerts, where the port connection ranges are:

- Source Port: 53 a 30227
- Destination Port: 22 a 16216

In order to further restrict the ports in which the connexions were made, we apply the algorithm again, generating a total of 7 new clusters as seen in Figure 4.2.

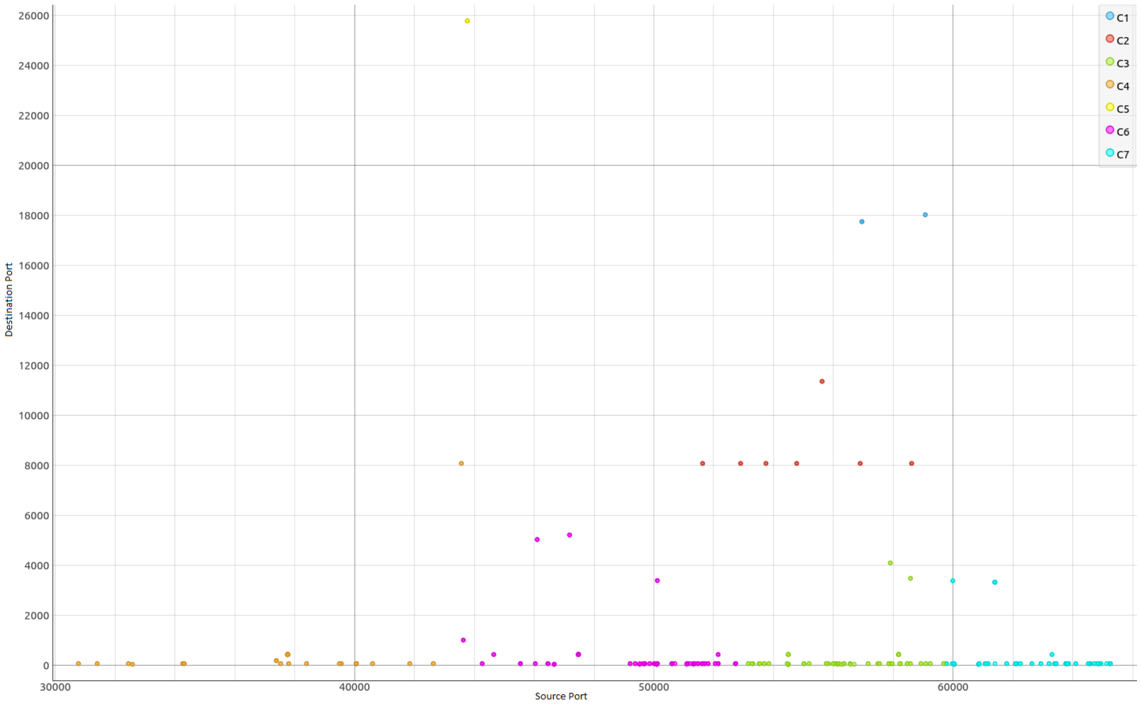


Figure 4.2: Sub Clusters 1

Cluster 1.1 has only one alert classification with a total of 6 connections, between 2 source IPS and 1 destination IP as seen in Table 4.1.

Table 4.1: Cluster 1.1

Classification	Source IP	Destination IP	Source Port	Destination Port
Misc Attack	78.128.112.74	193.136.195.94	56964	17743
Misc Attack	5.188.206.248	193.136.195.94	59081	18018

- Misc Attack:
  - The IP 78.128.112.74 is listed as malicious by the Collective Intelligence Network Security COLLECTIVE INTELLIGENCE NETWORK SECURITY (CINS).
  - The IP 5.188.206.248 is on the black list of the Dshield group.

Cluster 1.2 has 2 alerts classification with total of 14 connections, between 3 source IPS and 5 destination IPS, as seen in Table 4.2.

Table 4.2: Cluster 1.2

Classification	Source IP	Destination IP	Sorce Port	Destination Port
A Network Trojan was Detected	193.136.195.94	119.59.124.163	51641	8080
A Network Trojan was Detected	193.136.195.94	95.110.231.207	56907	8080
A Network Trojan was Detected	193.136.195.94	178.79.172.45	58624	8080
A Network Trojan was Detected	192.168.0.162	119.59.124.163	52910	8080
A Network Trojan was Detected	192.168.0.162	62.75.145.252	54784	8080
Misc Attack	146.185.222.51	193.136.195.94	55632	11359

- A Network Trojan was Detected:
  - The IPS 119.59.124.163 and 119.59.124.163 are associated with a command and control server of the Dridex botnet, but are already deactivated.
  - The IPS 95.110.231.207, 178.79.172.45 and 62.75.145.252 are associated with a command and control server of the Heodo botnet, but are already disabled.
- Misck Attack: 146.185.222.51 is in the black list of the Dshield group.

Cluster 1.3 has 2 alert classifications with a total of 167 connections, between 4 source IPS and 10 destination IPS seen in Table 4.3.

Table 4.3: Cluster 1.3

Classification	Source IP	Destination IP	Source Port	Destination Port
Misc activity	192.168.0.162	8.8.8.8	56279	53
Misc activity	192.168.0.162	8.8.4.4	56157	53
A Network Trojan was Detected	193.136.195.94	46.163.78.94	54503	443
Potential Corporate Privacy Violation	192.168.0.162	78.47.139.102	58237	80
Potential Corporate Privacy Violation	192.168.0.162	216.146.43.71	53166	80
Potential Corporate Privacy Violation	192.168.0.162	216.146.43.70	58248	80
Potential Corporate Privacy Violation	192.168.0.162	162.88.96.194	57863	80
Potential Corporate Privacy Violation	192.168.0.162	162.88.100.200	59708	80
Misc Attack	146.185.222.12	193.136.195.94	57912	4105
Misc Attack	5.188.10.103	193.136.195.94	58586	3486
Generic Protocol Command Decode	193.136.195.94	104.17.107.77	58191	443

- Misc activity: IP 192.168.0.162 made requests for Google DNS server (8.8.8.8 and 8.8.4.4) for some malicious Fully Qualified Domain Name.

- Network Trojan was Detected: IP 46.163.78.94 is associated with a command and control server of the Heodo Botnet, but is already disabled.

- Potential Corporate Privacy Violation:

  - The IPS 216.146.43.71, 216.146.43.70, 162.88.96.194 and 162.88.100.200 are associated with an Internet service that aims to discover the valid IP of a given host. This measure is used to ensure that the Botnet can communicate with the real IP and maintain an active connection with the host.

  - The IP 78.47.139.102 simply returns an HTML page with a "works" message we believe is a way to verify that the host has an active connection and to get the source IP of the client.

- Misck Attack: The IPS 146.185.222.12 is in the black list of the Dshield group.
- Generic Protocol Command Decode: The IP 104.17.107.77 belongs to the WhatsApp domain.

Cluster 1.4 has 5 alert classifications with a total of 330 connections, between 3 source IPS and 7 destination IPS seen in Table 4.4.

Table 4.4: Cluster 1.4

Classification	Source IP	Destination IP	Source Port	Destination Port
Potentially Bad Traffic	193.136.195.94	8.8.8.8	32589	53
Potential Corporate Privacy Violation	193.136.195.94	78.47.139.102	30785	80
Potential Corporate Privacy Violation	193.136.195.94	162.88.96.194	34269	80
Potential Corporate Privacy Violation	193.136.195.94	216.146.38.70	34321	80
Potential Corporate Privacy Violation	193.136.195.94	162.88.96.194	38407	80
Generic Protocol Command Decode	193.136.195.94	13.107.4.50	40069	80
Generic Protocol Command Decode	193.136.195.94	104.17.107.77	37776	443
Misc Attack	80.211.154.197	193.136.195.94	42649	81
Misc Attack	45.55.0.202	193.136.195.94	37400	199
A Network Trojan was Detected	193.136.195.94	119.59.124.163	43579	8080

- Potentially Bad Traffic: Suspicious requests for Google DNS server (8.8.8.8) for some malicious Full Qualified Domain Name.

- Potential Corporate Privacy Violation:
  - IP 78.47.139.102 simply returns an HTML page with a "works" message we believe is a way to verify that the host is with an active connection and to get the source IP of the client.
  - The IPS 78.47.139.102, 162.88.96.194, 216.146.38.70, 162.88.96.194 are associated with an Internet service that aims to find out the valid IP of a given host. This measure is used to ensure that the Botnet can communicate with the actual IP and maintain an active connection with the host.
- Generic Protocol Command Decode: IPS 13.107.4.50 (windows update) and 104.17.107.77 (WhatsApp), are false positives found in the network.
- Misc Attack: IPS 80.211.154.197 and 45.55.0.202, are listed as malicious by Collective Intelligence Network Security CINS.
- Network Trojan was Detected: IP 119.59.124.163 is associated with a command and control server from Botet Drixex.

Cluster 1.5 has 1 alert classification with a total of 4 connections, between 1 source IP and 1 destination IP seen in Table 4.5.

Table 4.5: Cluster 1.5

Classification	Source IP	Destination IP	Source Port	Destination Port
Misc Attack	77.72.82.14	193.136.195.94	43781	25766

- Misc Attack: The IP 77.72.82.14 is associated with the Dshield blacklist.

Cluster 1.6 has 6 alert classifications with a total of 121 connections, between 7 source IPS and 10 destination IPS as seen in Table 4.6.

Table 4.6: Cluster 1.6

Classification	Source IP	Destination IP	Source Port	Destination Port
Generic Protocol Command Decode	192.168.0.146	169.44.82.118	47492	443
A Network Trojan was Detected	192.168.0.162	81.88.24.211	52159	443
Misc activity	192.168.0.162	8.8.8.8	49537	53
Misc activity	192.168.0.162	8.8.4.4	51110	53
Misc activity	193.136.195.94	8.8.4.4	50108	53
Misc Attack	176.119.7.54	193.136.195.94	50127	3399
Misc Attack	63.143.33.110	193.136.195.94	47195	5222
Misc Attack	77.72.83.234	193.136.195.94	43644	1020
Misc Attack	5.189.226.102	193.136.195.94	46113	5038
Potential Corporate Privacy Violation	192.168.0.162	78.47.139.102	51380	80
Potential Corporate Privacy Violation	192.168.0.162	216.146.43.71	50623	80
Potential Corporate Privacy Violation	192.168.0.162	216.146.43.70	51710	80
Potential Corporate Privacy Violation	192.168.0.162	198.27.74.146	49671	80
Potential Corporate Privacy Violation	192.168.0.162	162.88.100.200	51487	80
Potential Corporate Privacy Violation	192.168.0.162	146.255.36.1	49671	80
Potentially Bad Traffic	193.136.195.94	8.8.8.8	46682	53

- Generic Protocol Command Decode: The IP 169.44.82.118 is associated with the WhatsApp Domain, probably being a false positive

- Network Trojan was Detected: The IP 81.88.24.211 is associated with a command and control server of the Heodo botnet, but is already disabled.

- Misc activity: Malicious queries to Google DNS servers (8.8.8.8 and 8.8.4.4)

- Misc Attack:

  - The IP 176.119.7.54 is associated with BACKDOOR DoomJuice.

  - The IPS 63.143.33.110, 77.72.83.234.5 and 189.226.102 are listed as malicious by Collective Intelligence Network Security CINS.

- Potential Corporate Privacy Violation: IPS 78.47.139.102, 216.146.43.71,216,146.43.70, 198.27.74.146, 162.88.100.200 and 146.255.36.1 are associated with a service on the Internet that aims to discover the valid IP of a given host. This measure is used to ensure that the Botnet can communicate with the actual IP and maintain an active connection with the host.

- Potentially Bad Traffic: Malicious requests for known domains.

Cluster 1.7 has 5 alert classifications with a total of 100 connections, between 3 source IPS and 11 destination IPS seen in Table 4.7.

Table 4.7: Cluster 1.7

Classification	Source IP	Destination IP	Source Port	Destination Port
A Network Trojan was Detected	192.168.0.162	103.4.18.170	63313	443
Generic Protocol Command Decode	192.168.0.162	13.107.4.50	62102	80
Potential Corporate Privacy Violation	192.168.0.162	162.88.96.194	63871	80
Potential Corporate Privacy Violation	192.168.0.162	162.88.100.200	65137	80
Potential Corporate Privacy Violation	193.136.195.94	216.146.38.70	62258	80
Potential Corporate Privacy Violation	192.168.0.162	216.146.43.71	62141	80
Potential Corporate Privacy Violation	192.168.0.162	78.47.139.102	64887	80
Potential Corporate Privacy Violation	192.168.0.162	185.26.99.195	61405	3333
Potentially Bad Traffic	192.168.0.162	8.8.8.8	60049	53
Misc Attack	5.189.226.180	193.136.195.94	60000	3389

- Network Trojan was Detected:The IP 103.4.18.170 is associated with a Dridex command and control server, but is already disabled.

- Generic Protocol Command Decode:The IP 13.107.4.50 is a false positive due to windows updates.

- Potential Corporate Privacy Violation: The IPS 162.88.96.194, 162.88.100.200, 216.146.38.70, 216.146.43.71, 78.47.139.102, 185.26.99.195 are associated with a service on the Internet that aims to discover the valid IP of a particular host, this measure is used to ensure that the Botnet can communicate with the real IP and maintain an active connection with the host.

- Potentially Bad Traffic: Queries for Probably Malicious Domains.
- Misc Attack: The IP 5.189.226.180 is associated with the Dshield black list.

4.1.2 Cluster 2

Cluster 2 is visualized with the red color on Figure 4.1, with a total of 1030 alerts the connection ranges of the ports are:

- Source Port: 53 to 443
- Destination Port: 21289 to 45553

In order to be able to restrict more the ports in which the connections were made, we apply the algorithm again, generating a total of 2 new clusters seen in Figure 4.3

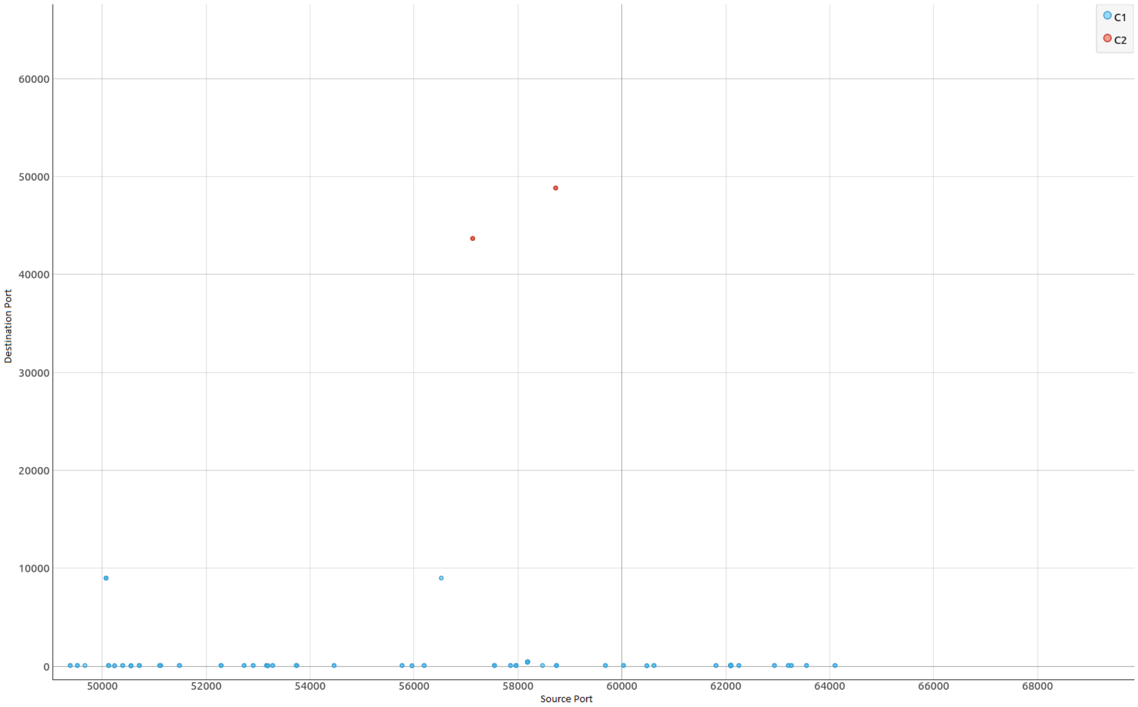


Figure 4.3: Sub Clusters 2

Cluster 2.1 has 5 alert classifications with a total of 321 connections, between 12 source IPS and 2 destination IPS seen in the Table 4.8.

Table 4.8: Cluster 2.1

Classification	Source IP	Destination IP	Source Port	Destination Port
Potential Corporate Privacy Violation	59.38.112.38	192.168.0.162	80	50719
Potentially Bad Traffic	216.146.43.71	192.168.0.162	80	52737
Potentially Bad Traffic	216.146.43.70	192.168.0.162	80	52911
Potentially Bad Traffic	216.146.43.71	193.136.195.94	80	49527
Potentially Bad Traffic	162.88.100.200	193.136.195.94	80	61816
Potentially Bad Traffic	162.88.96.194	192.168.0.162	80	57863
Potentially Bad Traffic	162.88.96.194	193.136.195.94	80	60038
Potentially Bad Traffic	162.88.100.200	192.168.0.162	80	50128
A Network Trojan was Detected	8.8.8.8	192.168.0.162	53	53195
A Network Trojan was Detected	8.8.4.4	192.168.0.162	53	60487
Misc Attack	109.239.79.181	192.168.0.162	9001	50079
Misc Attack	51.68.77.241	193.136.195.94	9001	56530
Generic Protocol Command Decode	104.17.107.77	193.136.195.94	443	58191
Generic Protocol Command Decode	13.107.4.50	192.168.0.162	80	62102

- Potentially Bad Traffic: The IPS 162.88.96.194, 162.88.100.200, 216.146.38.70 and 216.146.43.71, are associated with an Internet service that aims to find out the valid IP of a particular host, this measure is used to ensure that the Botnet can communicate with the actual IP and maintain an active connection with the host.

- The Network Trojan was Detected: Alert triggered for Domain Name System (DNS) requests to Google DNS server 8.8.8.8 to for GameOver ZeuS (GOZ) botnet domains.

- Generic Protocol Command Decode: The IP 104.17.107.77 belongs to the WhatsApp domain and and 13.107.4.50 is a false positive due to windows updates.

Cluster 2.2 has 1 alert classification with a total of 4 connections, between 2 source IPS and 1 destination IP seen in Table 4.9.

Table 4.9: Cluster 2.2

Classification	Source IP	Destination IP	Source Port	Destination Port
Misc Attack	146.185.222.35	193.136.195.94	48830	58732
Misc Attack	146.185.222.29	193.136.195.94	43671	57135

- Misc Attack: the IP 146.185.222.35 and 146.185.222.29 are associated with the Dshield black list.

4.1.3 Cluster 3

Cluster 3 is displayed in green, on Figure 4.1, with a total of 139 alerts the connection ranges of the ports are:

- Source Port: 53 to 30227
- Destination Port: 22 to 16216

In order to restrict the ports in which the connexions were made, we apply the algorithm again, generating a total of 2 new clusters as seen in Figure 4.4.

Cluster 3.1 has 5 alert classifications with a total of 58 connections, between 5 source IPS and 8 destination IPS as seen in Table 4.10

Classification	Source IP	Destination IP	Source Port	Destination Port

Generic Protocol Command Decode	193.136.195.94	52.210.42.194	16570	80
Misc Attack	82.221.105.7	193.136.195.94	17268	175
Misc Attack	80.82.77.139	193.136.195.94	30227	2152
Misc Attack	41.184.186.216	193.136.195.94	29302	23
Misc activity	193.136.195.94	8.8.8.8	20515	53
Potential Corporate Privacy Violation	193.136.195.94	216.146.38.70	12910	80
Potential Corporate Privacy Violation	193.136.195.94	162.88.100.200	13007	80
Potential Corporate Privacy Violation	193.136.195.94	78.47.139.102	14549	80
Potential Corporate Privacy Violation	193.136.195.94	216.146.43.71	29001	80
Potentially Bad Traffic	193.136.195.94	8.8.8.8	16643	53
Potentially Bad Traffic	222.186.15.66	193.136.195.94	24103	3306

Table 4.10: Cluster 3.1

- Generic Protocol Command Decode: The IP 52.210.42.194 is linked to the site "http://www.bsnett.no/"

- Misc Attack: The IPS 82.221.105.7 and 80.82.77.139 are listed as malicious by Collective Intelligence Network Security CINS.

- Misc activity: Domain Solicitations to Google DNS (8.8.8.8) considered malicious.

- Potential Corporate Privacy Violation:

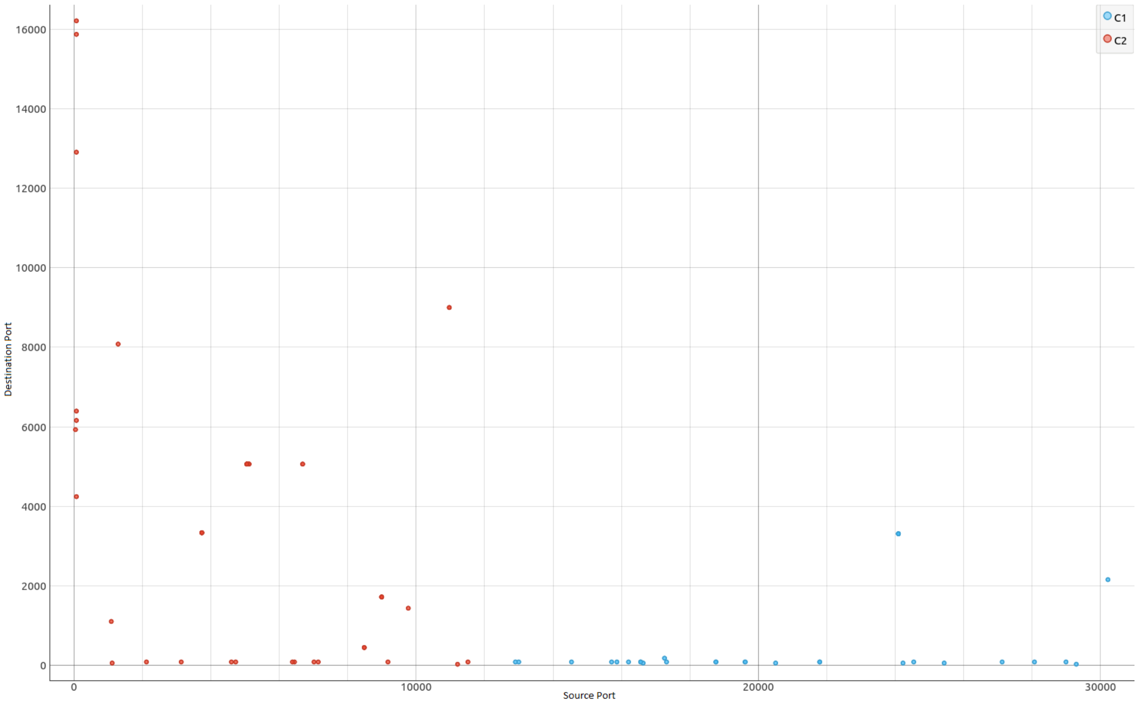


Figure 4.4: Sub Clusters 3

  - The IP 78.47.139.102 simply returns an HTML page with a "works" message we believe is a way to verify that the host is with an active connection.
  - The IPS 216.146.38.70, 216.146.43.71 and 162.88.100.200 are associated with an Internet service that aims to discover the valid IP of a given host, this measure is used to ensure that the botnet can communicate with the real IP and maintain an active connection with the host.

- Potentially Bad Traffic: IP 222.186.15.66 made attempts to attack through the MySQL server.

Cluster 3.2 has alert classification with total of 6 connections, between 14 source IPS and 9 destination IPS as seen in the Table 4.11.

Classification	Source IP	Destination IP	Source Port	Destination Port

A Network Trojan was Detected	8.8.4.4	193.136.195.94	53	5927
A Network Trojan was Detected	193.136.195.94	62.210.36.193	1299	8080
Attempted Information Leak	104.243.143.70	193.136.195.94	5060	5060
Attempted Information Leak	158.69.207.26	193.136.195.94	5063	5060
Attempted Information Leak	37.49.231.144	193.136.195.94	5122	5060
Attempted Information Leak	62.210.103.172	193.136.195.94	6691	5060
Misc Attack	37.191.196.1	193.136.195.94	11219	22
Misc Attack	71.6.233.14	193.136.195.94	1099	1099
Misc Attack	109.239.79.181	193.136.195.94	9001	1715
Misc Attack	37.49.231.144	193.136.195.94	5122	5060
Misc Attack	196.52.43.90	193.136.195.94	10978	9000
Misc activity	193.136.195.94	8.8.4.4	1124	53
Misc activity	109.239.79.181	193.136.195.94	9001	1715
Potential Corporate Privacy Violation	193.136.195.94	78.47.139.102	2127	80
Potential Corporate Privacy Violation	193.136.195.94	216.146.43.71	4734	80

Potential Corporate Privacy Violation	193.136.195.94	162.88.100.200	6394	80
Potential Corporate Privacy Violation	193.136.195.94	163.172.226.137	8492	443
Potential Corporate Privacy Violation	193.136.195.94	185.26.99.195	3745	3333
Potentially Bad Traffic	223.72.54.251	193.136.195.94	9779	1433
Potentially Bad Traffic	216.146.43.70	193.136.195.94	80	4241
Potentially Bad Traffic	162.88.100.200	193.136.195.94	80	6394

Table 4.11: Cluster 3.2

- The Network Trojan was Detected:
  - Rule triggered for DNS resolutions to Google DNS servers (8.8.8.8) to GameOver ZeuS (GOZ) botnet domains.
  - The IPS 62.210.36.193 is associated with a command and control server for the Dridex botnet, but is already disabled.
- Attempted Information Leak: The IPS 104.243.143.70, 158.69.207.26, 37.49.231.144 and 62.210.103.172 are associated with possible network scan attack.
- Misc Attack:

  - The IPS 37.191.196.1, 71.6.233.14 and 37.49.231.144 are listed as malicious by Collective Intelligence Network Security (CINS).
  - The IPS 196.52.43.90, 196.52.43.90 are associated with the Dshield black list.

- Misc activity: The IP 109.239.79.181 is associated with unusual activities in the network.
- Potential Corporate Privacy Violation:
  - IPS 78.47.139.102, 216.146.43.71 and 162.88.100.200 are associated with an Internet service that aims to find out the valid IP of a particular host. This measure is used to ensure that the Botnet can communicate with the actual IP and maintain an active connection with the host.
  - The IPS 163.172.226.137 and 185.26.99.195 have the activity related to Bitcoins.
- Potentially Bad Traffic:
  - The IP 223.72.54.251 may be scanning MSSQL.
  - The IPS 216.146.43.70 and 162.88.100.200 in Potentially Bad Traffic category are responding to DNS queries made by the infected host.

4.1.4 Cluster 4

Cluster 4 is displayed in yellow, on Figure 4.1, with a total of 1030 alerts. The input ports are in the range of:

- Source Port: 53 a 48830
- Destination Port :49389 a 64108

In order to further restrict the ports in which the connexions were made we apply the algorithm again, which generated a total of 8 new clusters as seen in the figure4.5

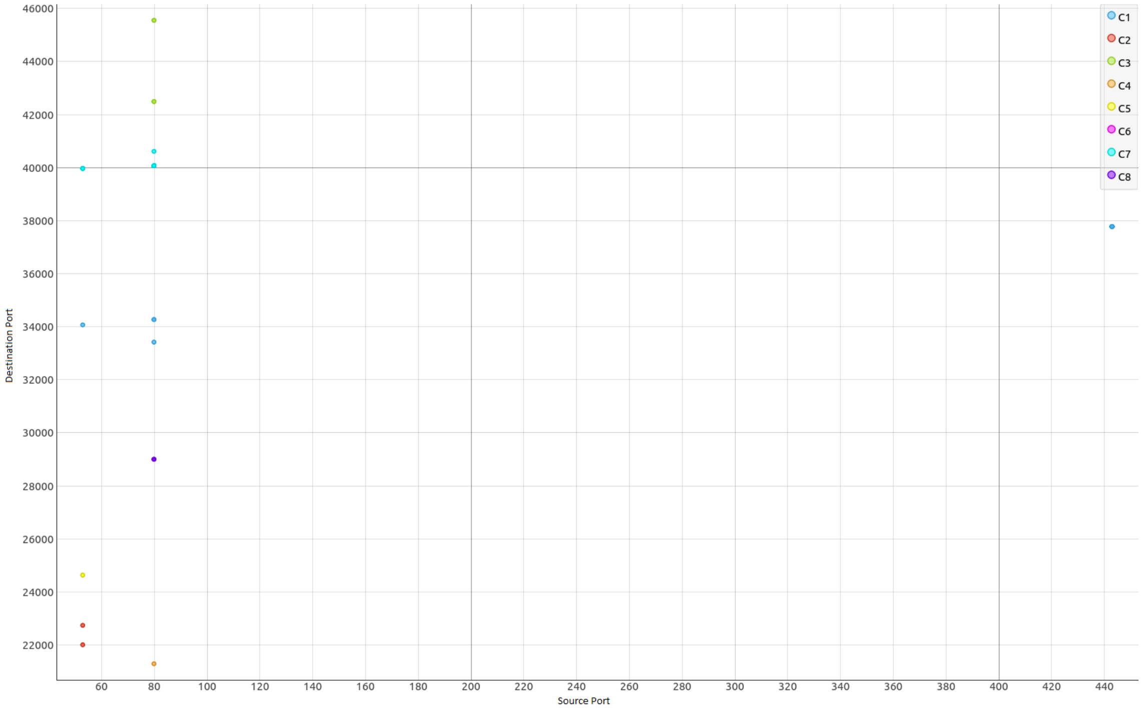


Figure 4.5: Sub Clusters 4

Cluster 4.1 has 3 alert classifications with a total of 913 connections, between 4 source IPS and 2 destination IPS as seen in the table 4.12.

Table 4.12: Cluster 4.1

Classification	Source IP	Destination IP	Source Port	Destination Port
Generic Protocol Command Decode	104.17.107.77	192.168.0.146	443	37776
A Network Trojan was Detected	8.8.8.8	193.136.195.94	53	34068
Potentially Bad Traffic	216.146.43.71	193.136.195.94	80	33418
Potentially Bad Traffic	162.88.96.194	193.136.195.94	80	33418

- Generic Protocol Command Decode: The IP 104.17.107.77 belongs to the WhatsApp domain.

- Network Trojan was Detected: Alerts are Google responses related with possible malicious domains.

- Potentially Bad Traffic: The IPS 216.146.43.71 and 162.88.96.194 are associated with an Internet service that aims to discover the valid IP of a particular host.

Cluster 4.2 has 1 alert classification with a total of 4 connections, between 1 source IP and 1 destination IP as seen in Table 4.13.

Table 4.13: Cluster 4.2

Classification	Source IP	Destination IP	Source Port	Destination Port
A Network Trojan was Detected	8.8.8.8	193.136.195.94	53	22738

- A network Trojan was Detected: DNS response from Google DNS servers about possible malicious domains.

Cluster 4.3 has 1 alert classification with a total of 7 connections, between 2 source IPS and 1 destination IP seen in Table 4.14.

Table 4.14: Cluster 4.3

Classification	Source IP	Destination IP	Source Port	Destination Port
Potentially Bad Traffic	162.88.100.200	193.136.195.94	80	45553
Potentially Bad Traffic	216.146.38.70	193.136.195.94	80	42492

- Potentially Bad Traffic: The IPS 216.146.43.70 and 162.88.100.200 are associated with an Internet service that aims to discover the valid IP of a given host. This measure is used to ensure that the Botnet can communicate with the real IP and maintain an active connection with the host.

Cluster 4.4 has 1 alert classification with a total of 2 connections, between 1 source IP and 1 destination IP seen in Table 4.15.

Table 4.15: Cluster 4.4

Classification	Source IP	Destination IP	Source Port	Destination Port
Potentially Bad Traffic	216.146.38.70	193.136.195.94	80	21289

- Potentially Bad Traffic: IP 216.146.43.70 are associated with an Internet service that aims to find out the valid IP of a given host. This measure is used to ensure that the Botnet can communicate with the real IP and maintain an active connection with the host.

Cluster 4.5 has alert classification with total of 2 connections, between 1 source IP and 1 destination IP seen in the Table 4.16.

Table 4.16: Cluster 4.5

Classification	Source IP	Destination IP	Source Port	Destination Port
A Network Trojan was Detected	8.8.8.8	193.136.195.94	53	24632

- A Network Trojan was Detected: Google DNS response for possible malicious domains.

Cluster 4.6 did not get any alerts.

Cluster 7 has 3 alert classifications with total of 95 connections, between 3 source IPS and 1 destination IP seen in the Table 4.17.

Table 4.17: Cluster 4.7

Classification	Source IP	Destination IP	Source Port	Destination Port
A Network Trojan was Detected	8.8.8.8	193.136.195.94	53	39968
Potentially Bad Traffic	162.88.96.194	193.136.195.94	80	40612
Generic Protocol Command Decode	13.107.4.50	193.136.195.94	80	40069

- Network Trojan was Detected: Google DNS response related with possible malicious domains.

- Potentially Bad Traffic: The 162.88.96.194 is associated with an Internet service that aims to find out the valid IP of a particular host. This measure is used to ensure that the Botnet can communicate with the real IP and maintain an active connection with the host.
- Generic Protocol Command Decode: The IP 13.107.4.50 represents a false positive due to Windows updates.

Cluster 8 has 1 alert classification with a total of 4 connections, between 1 source IP and 1 destination IP seen in Table 4.18.

Table 4.18: Cluster 4.8

Classification	Source IP	Destination IP	Source Port	Destination Port
Potentially Bad Traffic	216.146.43.71	193.136.195.94	80	29001

- Potentially Bad Traffic:The IPS 162.88.96.194 and 216.146.43.71 are associated with an Internet service that aims to find out the valid IP of a particular host. This measure is used to ensure that the Botnet can communicate with real IP and maintain an active connection to the host.

4.2 Outliers

Outliers are data that may be anomalies, may not fit into a category or be too far away from the desired grouping.

The Outliners data obtained in the analyzes has the number of 873 alerts in 7 categories of Suricata alerts.

- Network Trojan was Detected: A total of 80 alerts were obtained.
- Attempted Information Leak: A total of 6 alerts were obtained.
- Generic Protocol Command Decode: A total of 29 alerts were obtained.

- Misc Attack: A total of 77 alerts were obtained.
- Misc activity: A total of 90 alerts were obtained.
- Potential Corporate Privacy Violation: A total of 346 alerts were obtained.
- Potentially Bad Traffic: A total of 245 alerts were obtained.

4.3 HIDS

With the alerts column generated by HIDS, we can see the messages it has triggered on certain events. These messages are important due to the fact that they show the internal activities that the Host generated.

A total of 18 types of alerts were recorded, shown below, in a total of 3112 registered alerts.

- Login session opened: This alert is triggered when a session is started on the host.
- New dpkg (Debian Package) installed: This alert is triggered when a new package is installed in the debian distribution.
- Dpkg (Debian Package) removed: This alert is triggered when a package is removed in the debian distribution.
- Multiple Windows error events:This alert is triggered when multiple windows errors are generated.
- OSSEC agent started: This alert is triggered when the OSSEC agent starts.
- OSSEC agent disconnected: This alert is triggered when the agent disconnects.
- Registry Entry Added to the System: This alert is triggered when a log is added to the system.

- Registry Integrity Checksum Changed: This alert is triggered when some file has a change, and can be with some update or intentional change.

- Registry Integrity Checksum Changed Again (2nd time): This alert is triggered when many changes occur in some file accordingly.

- Service startup type was changed: This alert is triggered when some service changes its type of execution.

- Successful sudo to ROOT executed: This alert is triggered when the SUDO command is executed.

- System time changed: This alert is triggered when system time changes.

- Unknown problem somewhere in the system: This alert is triggered when some problem is found in the system.

- User successfully changed UID to root: This alert is triggered when any user changes the user ID.

- Windows Audit Policy changed: This alert is triggered when some windows security policy is changed.

- Windows Logon Success: This alert is triggered when some logon and registered in the systems

- Windows error event: This alert is triggered when some errors in the windows system registered.

As measured at the beginning of the chapter, all data analyzed is from the correlation between NIDS and HIDS, within 2 seconds. The IP 193.136.195.94 is the public address used of the gateway on the Network Address Translation to connect the machines on the local network to the Internet. Some activities that we can highlight among this correlation are described below:

- The first relation presented happened on the day 2018-08-17 in the interval between 06:00:07 and 06:00:08. The "New dpkg (Debian Package) prompt requested to install" was triggered at 06:00:07, stating that some package was installed on the system as early as 06:00:08. The NIDS generated the alert "ET DNS Query for .su TLD (Soviet Union) Often Malware Related" in the classification of Potentially Bad Traffic, from IP 193.136.195.94 to 8.8.8.8.

  Soon after this alert was registered another event that happened between 06:00:35 and 06:00:37, the alert of the HIDS triggered was the "New dpkg (Debian Package) installed" and the one of the NIDS was the " Network Trojan was Detected0 "with DNS origin from Google 8.8.8.8 to the infected VMWARE 193.136.195.94.

- The second relation also occurred on the day 2018-08-17 between the hour of 20:19:10 and 20:19:08. The HIDS alert "Registry Entry Added to the System" was triggered where a record was added to the system, soon after the NIDS generated the alert "ET SCAN Sipvicious User-Agent Detected (friendly-scanner)" of the classification Attempted Information Leak, of origin IP 37.49.231.144 to 193.136.195.94.

  After these alerts triggered another HIDS alert "Registry Entry Added to the System" but with the difference in the classification and alert message of the NIDS that, ET CINS Active Threat Intelligence Poor Reputation IP group 24, classification Misc Attack on the same IP cited above.

- The third report presented, occurred on 2018-09-01 in the interval between 19:39:03 and 19:39:05, the NIDS alert "ET DNS Query to a * .pw domain - Likely Hostile" was triggered regarding the classification Potentially Bad Traffic. This alert is about possible requests for malicious domains. After this event HIDS triggered the warning of "Successful sudo to ROOT executed" referring to the sudo command used by the user. After this, HIDS triggered another alert "User successfully changed UID to root", which means that the user ID has been changed.

- In the fourth relation presented, occurred on the 2018-09-04 between the schedules

21:41:26 and 21:41:28, HIDS generated the alert "Windows Logon Success" that represents a logon in the system. After the NIDS generated the alert "ET TROJAN ABUSE.CH SSL Blacklist Malicious SSL certificate detected (Dridex)" for classification "A Network Trojan was Detected". The source IP for this connection is 208.87.225.248 which is classified as an active C&C for the Dridex botnet.

- In the fifth relation presented, occurred on days 2018-08-16 and 2018-08-17, on 2018-08-16, between the hours of 11:26:25 and 11:26:23, the HIDS fired the alert "Service startup type was changed", after this alert NIDS began to receive the alert with the message "ET TROJAN DNS Reply Sinkhole Microsoft NO-IP Domain".

  On the day 2018-08-17, at 03:53:04 HIDS received the "Login session opened" alert. After this session was opened, new packages were installed, at 06:00:35, after 1 pm and 18 HIDS generated the warning "Dpkg (Debian Package) removed" we believe that the malware installed its dependencies and soon after its use the program has removed itself to leave no clues.

We were able to gain insight into the activities found on infected hosts with HIDS alerts. The vast majority of the activities found were requests for possibly malicious domains already registered by NIDS, as it already obtains an amount of updated rules the C&Care disabled, having only 1 in operation and still being able to connect, as described in Table 4.19. These command and control servers were found on the site "https://feodotracker.abuse.ch".

Table 4.19: Botnets

IP	Botnet	Status
46.163.78.94	Heodo	off-line
62.75.145.252	Heodo	off-line
62.210.36.193	Heodo	off-line
81.88.24.211	Heodo	off-line
95.110.231.207	Heodo	off-line
178.79.172.45	Heodo	off-line
192.155.83.86	Heodo	off-line
103.4.18.170	Dridex	off-line
119.59.124.163	Dridex	off-line
208.87.225.248	Dridex	on-line

The Dridex and Heodo Botnets are two versions of the botnet known as Feodo, a Trojan used to gain privileged information from the infected computer, such as banking data and system credentials. There are currently 5 versions:

- Version A: Its main feature is that it is hosted on a Web server running a proxy for port 8080/TCP, waiting for the connections and relaying the traffic to another node. Due to this way of acting, Botnet traffic hits the hots without using domain names, which makes it difficult to detect by the IDS[40].

- Version B: It is also hosted on a web server, acting with the domain names in the .ru ccTLD. The current traffic usually runs over port 80/TCP[40].

- Version C: In this version there is already a change in the URL structure used for data transmissions. It is called Geode or Emotet[40].

- Version D: It is currently known as Dridex. Its operation uses a different infrastructure from the usual command and control servers, but sharing the same logic. In Dridex the main Botnet is fragmented into small nuclei of logical botnets where each one has an identity. This fragmentation is used so that each segment is responsible by a specific activity such as. Scanning possible targets, direct attacks, theft of information, among others. For more information access the link "https://www.bitsighttech.com/blog/dridex-botnets"[40].

- Version E: It is the successor of Version C called Heodo. This Botnet is already directed to multiple actions like DDOS attacks, encrypting the host content, stealing of information, among others [41]. For more information access the link "https://fortiguard.com/encyclopedia/botnet/7630295".

With the implementation of the k-means algorithm we can observe that the alerts are repeated in almost all the clusters, due to the fact that few IPS were registered in the alerts, and only have 7 categories. Ports detected in alerts usually are associated with common services such as 443 (https) and 80 (http). In this way the Firewal believes that they are legitimate traffic, but with malicious activities behind.

Chapter 5

Conclusion and Future Work

This chapter presents the main conclusions, additional work improvements and new research vectors.

5.1 Conclusion

The main purpose of this work was to analyze how IDS detects Botnets and improve detection of the related traffic. To achieve that a network topology has been implemented to capture traffic from malware, focused on detecting the botnets.

In the topology, several components were deployed: i) a NIDS control network traffic; ii) two HIDS in virtual machines for host-specific controls; iii) a honeypot in the internal network for the detection of possible attacks from infected VM-wares; and iv) honeypot in the external network to attract possible malware from the Internet.

Considering the behavior of the Botnet threats it was necessary to overcome the adversities in the choice and implementation of the devices.

The infection phase was challenging because most of the malware found in repositories is very old and some of the Botnets were already taken down.

It was possible to correlate the alerts between HIDS and NIDS, it is possible to perceive the infections, such as installing malware and communicating with their C&C.

We used machine learning algorithms to help the classification of Botnet related traffic and understand the challenges of detecting it.

It was not possible to observe the contents of the packets, as SSL and IPSEC protocols, that are being increasingly used for the "security" of the Botnet communication, reaching directly in the IDS to detect these exchanges of malicious messages.

The honeypot did not get a very positive correlation with IDS detection, because infected hosts did not try to spread over the network or scan it. This could have happened due to the fact of during the 3 week time, that the devices were interconnected, the honeypot did not alert for malware behavior or the malware was not configured to take those actions.

5.2 Future Work

As future work we can change and implement more types of honeypot in the same topology, so we can analyze which one has a better performance.

Another future approach is to change the topology to cover more points of attack in the detection of Malware. The honeypot in the external network can be better analyzed to find correlation between the attacks received and infections on the internal network.

As during this work the network traffic was captured in the .pcap files, further analysis can be made to map the behavior of the Botnet directly, such as packet size, and the average time it communicates with C&C.

We think it can be useful to test diverse machine learning algorithms over the captured data, and, if possible,to continue to increase the data available and considered other feeds of information to correlate during the detection phase.

Bibliography

[1] M.-y. Report, "Cyber Attack Trends 2018 MID-YEAR REPORT Check Point Cyber Attack Trends: Mid-Year Report 2018", 2018. [Online]. Available: `https://research.checkpoint.com/wp-content/uploads/2018/07/Cyber-Attack-Trends-2018-Mid-Year-Report.pdf`.

[2] [Online]. Available: `https://www.spamhaus.org/news/article/772/spamhaus-botnet-threat-report-2017` (visited on 10/29/2018).

[3] [Online]. Available: `https://securelist.com/ddos-report-in-q2-2018/86537/` (visited on 10/29/2018).

[4] M. Guedes, F. Mosqueira, S. S. Cordeiro, and W. A. Pinheiro, "A Guerra Cibernética: exploração , ataque e proteção cibernética no contexto dos sistemas de Comando e Controle (C 2)", vol. 33, pp. 11–18, 2016.

[5] V. V. GALHARDI, "Detecção adaptativa de anomalias em redes de computadores utilizando técnicas não supervisionadas", PhD thesis, 2017, p. 71.

[6] M. M. Faria, A. M. Monteiro, and C. L. Paulista, "Investigação sobre Técnicas de Detecção de Intrusões em Redes de Computadores com base nos Algoritmos Knn e K-Means", no. September, pp. 2–6, 2015.

[7] *Common Types of Cybersecurity Attacks and Hacking Techniques | Rapid7*. [Online]. Available: `https://bit.ly/2J2R3SN` (visited on 10/08/2018).

[8] T. C. Silva, D. P. Lozi, G. A. T. Souza, and L. B. Cancela, “TÉCNICAS DE INVASÃO: UM ESTUDO SOBRE AS ARMAS DO MUNDO DIGITAL”, pp. 1–6, 2017.

[9] D. Chiu, S.-h. Weng, and J. Chiu, *Backdoor Use in Targeted Attacks*.

[10] *What Is a Firewall? - Cisco*.

[11] *What is the weakness of a firewall? Are you at Risk*. [Online]. Available: `https://bit.ly/2StWXR3` (visited on 10/08/2018).

[12] “Networking, Security, and the Firewall”, *Configuring Sonicwall Firewalls*, pp. 1–50, Jan. 2006. DOI: `10.1016/B978-159749250-8/50005-8`.

[13] M. Scheidel and B. Raton, “INTRUSION DETECTION SYSTEM”, vol. 2, no. 12, 2009.

[14] *Intrusion Prevention Systems: the Next Step in the Evolution of IDS | Symantec Connect Community*. [Online]. Available: `https://symc.ly/2PuTeEl` (visited on 10/09/2018).

[15] M. F. A. ASSUNÇÃO, *Honeypots e Honeynets: Aprenda a detectar e enganar os invasores.* 2009.

[16] M. H. P. C. CHAVES, “Análise de estado de tráfego de redes tcp/ip para aplicação em detecção de intrusão”, PhD thesis, 2003.

[17] S. PPatil, “Intrusion Prevention System”, *International Journal of Emerging trends in Engineering and Development Issue*, vol. 24, no. 2, pp. 577–584, 2012, ISSN: 2249-6149.

[18] *O que é um Sistema de Prevenção de Intrusão IPS - BLOCKBIT UTM*. [Online]. Available: `https://bit.ly/2Sz7D15` (visited on 10/09/2018).

[19] A. A. Abdelkarim and H. H. O. Nasereddin, “Intrusion Prevention System”, *INTERNATIONAL JOURNAL Of ACADEMIC RESEARCH*, vol. 3, no. 1, pp. 432–434, 2011.

[20] I. Mokube and M. Adams, "Honeypots: Concepts, Approaches, and Challenges", PhD thesis, 2007, pp. 321–326, ISBN: 9781595936295. DOI: http://dx.doi.org/10.1145/1233341.1233399.

[21] S. Khattak, N. R. Ramay, K. R. Khan, A. A. Syed, and S. A. Khayam, "A Taxonomy of Botnet Behavior, Detection, and Defense", vol. 16, no. 2, pp. 898–924, 2014.

[22] P. Ferreira, "Detecção de Botnets", PhD thesis, 2013.

[23] A. Karasaridis, B. Rexroad, and D. Hoeflin, "Wide-scale botnet detection and characterization", *HotBots'07 Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets*, p. 7, 2007.

[24] R. Girardi moreira and D. Furtado Gonçalves, "Rio de Janeiro 2012 Instituto Militar de Engenharia", PhD thesis, 2012.

[25] P. Correia, "Caracterizaçao Estatística de Botnets", PhD thesis, 2011.

[26] H. Singh and A. Bijalwan, "A survey on Malware, Botnets and their detection", *International Journal of Advanced Engineering Research and Science (IJAERS)*, vol. 3, no. 3, pp. 85–90, 2016.

[27] J. R. L. Amaro and Y. G. Coimbra, "Ferramenta para detecção de padrões de botnet baseado em algoritmos de agrupamento de aprendizado de máquina", PhD thesis, Rio de Janeiro: Instituto Militar de Engenharia, 2016, p. 56, ISBN: 9788522508266. DOI: 10.13140/RG.2.1.1898.8409.

[28] S. Nagendra Prabhu and D. Shanthi, "A Survey on Anomaly Detection of Botnet in Network", *International Journal of Advance Research in Computer Science and Management Studies*, vol. 2, no. 1, pp. 2321–7782, 2014.

[29] T. T. Tu, H. Y. Liao, and M. Chen, "An Advanced Hybrid P2p Botnet 2.0", p. 3, 2011, ISSN: 2010376X.

[30] M. Feily, A. Shahrestani, and S. Ramadass, "A survey of botnet and botnet detection", *Proceedings - 2009 3rd International Conference on Emerging Security Information, Systems and Technologies, SECURWARE 2009*, pp. 268–273, 2009, ISSN: 2162-2108. DOI: `10.1109/SECURWARE.2009.48`. arXiv: `0112017 [cs]`.

[31] M. Fatima and M. Pasha, "Survey of Machine Learning Algorithms for Disease Diagnostic", *Journal of Intelligent Learning Systems and Applications*, vol. 09, no. 01, pp. 1–16, 2017, ISSN: 2150-8402. DOI: `10.4236/jilsa.2017.91001`.

[32] T. O. Ayodele, "Types of machine learning algorithms", in, 2010, ISBN: 953307034X. [Online]. Available: `https://bit.ly/2yFMKce`.

[33] C. Olivier, B. Schölkopf, and A. Zien, *Semi-Supervised Learning*, 2. 2006, vol. 1, p. 524, ISBN: 9780262033589. DOI: `10.1007/s12539-009-0016-2`. arXiv: `arXiv:1011.1669v3`.

[34] S. J. Gershman and N. D. Daw, "Reinforcement Learning and Episodic Memory in Humans and Animals: An Integrative Framework", *Ssrn*, 2017, ISSN: 0066-4308. DOI: `10.1146/annurev-psych-122414-033625`. arXiv: `15334406`.

[35] P. D. Gardiner, A. Eltigani, T. Willams, R. Kirkham, L. Ou, A. Calabrese, and J. Söderlund, *Evolutionary learning in strategy-project systems*. 2018, p. 274, ISBN: 9781628254846.

[36] [Online]. Available: `https://www.investopedia.com/terms/d/deep-learning.asp` (visited on 10/29/2018).

[37] S. Santra and P. P. Acharjya, "A Study And Analysis on Computer Network Topology For Data Communication", *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, no. 1, pp. 522–525, 2013. DOI: `10.1.1.413.7099`.

[38] S. Garcia, *Malware Capture Facility Project*. [Online]. Available: `https://bit.ly/2OUW45U` (visited on 08/05/2018).

[39] [Online]. Available: `https://orange.biolab.si/` (visited on 10/29/2018).

[40] [Online]. Available: `https://feodotracker.abuse.ch/` (visited on 10/29/2018).

[41] [Online]. Available: `https://fortiguard.com/encyclopedia/botnet/7630295`.