UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR

ENGINEERING FACULTY

Title: Data Model Classification Based on Machine Learning Techniques for Detection of Anomalous Traffic

Author: Luis Alfredo Alvarez Almeida

Jury

Jury

Supervisor:

Cartagena de Indias, 2019

Data Model Classification Based on Machine Learning Techniques for Detection of Anomalous Traffic

Luis Alfredo Alvarez Almeida
Supervisor: Juan Carlos Martínez Santos

Universidad Tecnológica de Bolívar
Faculty of Engineering
Master in Engineering with Emphasis in Computer Engineer
Cartagena de Indias, D. T. y C.

2019

Data Model Classification Based on Machine Learning Techniques for Detection of Anomalous Traffic

Luis Alfredo Alvarez Almeida

Trabajo de grado para optar al título de

Máster en Ingeniería con Énfasis en Sistemas

Director: Juan Carlos Martínez Santos

Universidad Tecnológica de Bolívar
Facultad de Ingeniería
Cartagena de Indias, D. T. y C.

2019

Resumen

Como consecuencia del gran avance tecnológico que atraviesa el mundo hoy en día y el surgimiento de nuevos modelos de prestación de servicios de negocios como la Computación en la Nube. Las compañías han optado por establecer nuevas estrategias de Marketing con el propósito de aumentar la cobertura, disponibilidad de sus servicios y utilidades. Y de esta forma llegar a convertir a muchos más usuarios en clientes. Por lo tanto, es de suma importancia para las empresas implementar el uso de plataformas digitales alojadas en una nube de servicios. La nube puede ser comprendida como un conjunto de servidores desde Internet encargados de atender las peticiones de los usuarios conectados al servicio en cualquier momento y parte del mundo. Esta arquitectura permite virtualizar recursos y aplicaciones, permitiendo beneficios como: reducción de costos de inversión y mantenimientos en equipos físicos de cómputo y facilidad capacidad de ofrecer múltiples servicios. Los servicios alojados en la nube están disponibles tanto para usuarios legítimos como para los atacantes cibernéticos y debido a esto, las empresas deben recurrir a la implementación de un sistema que sea capaz de monitorear, detectar y neutralizar cualquier tipo de amenaza que busque impedir el normal funcionamiento de los servicios alojados en la nube. Uno de los principales enemigos de los servicios alojados en la nube hoy en día son los Ataques Distribuidos de Denegación de Servicio (DDoS). Los cuales son considerados como una de las principales preocupaciones y amenazas económicas para la industria digital. Su objetivo principal es tratar de suspender temporalmente

los servicios de un servidor conectado a internet o dicho de otra forma causar que un servicio o recurso sea inaccesible a los usuarios legítimos. Cuando un ataque de DDoS es lanzado, el servidor victima recibe cientos o miles de peticiones por segundos. Posteriormente, el servidor intentará responder cada una de estas peticiones y con el paso del tiempo llevará sus recursos computacionales y conexiones al límite. Por ende, los usuarios legítimos no tendrán acceso al servidor [50]. Los ataques DDoS son excesivamente peligrosos, especialmente cuando son ejecutados de forma distribuida. La manera más rentable de liberar estos ataques y causar un gran impacto es ejecutándolos a través de botnes (redes de equipos zombies). De esta manera, los atacantes utilizan grandes cantidades de equipos infectados por un virus malware, que le permite al atacante tener acceso remoto a ellos y de esta forma llevar acabo el ataque manipulando los equipos comprometidos [18]. Actualmente existen herramientas para detectar ataques DDoS, como son: los dispositivos IDS (Instruction Detection System). La función de estos dispositivos es monitorear y analizar el tráfico de red generado por usuarios legítimos o falsos, examinar la violación de políticas por parte de un usuario e identificar patrones de actividades anormales dentro de un flujo de paquetes de datos [18]. Los dispositivos IDS pueden implementar dos tipos de detección de intrusos dentro de una red de datos, como son: la detección basada en firmas y la detección basada en anomalías. La detección basada en firmas es un enfoque utilizado para identificar patrones de comportamientos normales o anómalos dentro del tráfico de red, basando su criterio de clasificación en un registros

de ataques conocidos o bien llamados firmas [23]. Es decir, una vez que el dispositivo IDS analiza el tráfico de red y cada uno de los encabezados de los paquetes de datos, este compara el comportamiento observado en dicho tráfico con un registro de patrones de comportamiento previamente conocidos con el propósito de clasificar los nuevos patrones de comportamientos observados dentro del tráfico de red entrante. Este método resulta ser un poco complejo. Ya que depende en gran medida de que los patrones reflejados en el tráfico entrante se encuentren relacionados en la base de datos de ataques conocidos por el IDS. En consecuencia, es necesario mantener actualizada la base de datos de firmas permanentemente. Por otra parte, el método de detección basado en anomalías realiza la toma de decisiones implementando modelos aprendizaje automático que pueden ser entrenados y probados utilizando como insumo el tráfico de red. A diferencia del método basado en firmas, la detención basada en anomalías busca identificar los parámetros dentro del tráfico de red que aporten un valor significativo en el proceso de toma de decisión o clasificación de la información y no por el contrario analizar todos los parámetros definidos en el tráfico de red. [30].

Palabras Claves: Máquinas de Aprendizaje Automatico, Modelos de Clasificación, Ataques DDoS, Algoritmos de Máquina de Soportes Vectoriales, Métodos de Selección de Características.

Abstract

Thank digital revolution most companies can offer their services to multiples users around the world using an architecture called cloud computing which allows virtualizing resources and applications, reducing the cost of investment. In other hands, it is necessary to mention that these resources should be protected by a robust security system, and in this way, neutralize cyberattacks on the internet there are many threats. One of the most knowns are the distributed denial of service (DDoS) attacks. Because these are considered the most serious problem in a cloud computing environment. Their principal aim is try to suspend temporarily the services of a server connected to the internet through sending numerous requests causing system failures and high computational cost and thus to be able to prevent the access of legitimate users to the services [50]. The distributed denial of services attacks is highly dangerous, especially when the attack is distributed (DDoS). Eventually, these attacks are executed through botnes (zombies host network). To achieve a greater impact on the victim the attackers use numerous agents infected by malware which can be manipulated remotely to execute the attack [18]. Actually, there are tools to detect different kind of DDoS attacks, for instance: the IDS devices (Instruction Detection System), which are used for monitoring and analyzing the network traffic generated by users or robots, examine policy violation by a user, and identify abnormal activities patterns [18]. IDS devices can implement two type of abnormal traffic detection inside the data network, such as based signatures detection and anomaly detection. The based

signatures detection is an approach used to find attack patterns based on records of known attacks [23]. For example, once IDS device analyze traffic network, it checks the header of each packet then contrast the behavior of traffic with the signatures of know attack and finally make the classification traffic (anomaly or normal) based on its own knowledge. This approach can become not efficient, because it depends highly of the signature of know attack database, whether the attack is recorded in a database, the IDS could send the alert otherwise not, thus, it is very important to keep the signatures databases updated. However, the detection anomaly approach works better than based signatures detection approach. Because, this approach implements techniques based on the machine learning algorithm, using a model that can be trained and tested for analyzing a dataset (in this case network traffic) and then make a decision. One advantages of this method is that it does not need to be updated [30]. In this document will propose a computational model based on machine learning techniques that be able to make intelligent network traffic classification.

Keywords: Machine Learning, Classification Models, DDoS Attacks, Support Vector Machine, Feature Selection.

Special Thanks

In first place, I would like to thank God for allowing me to enjoy this wonderful academic process which has made me grow personally and professionally. Also, I would like to thank my beautiful wife for supporting me in all difficult moment and being over there giving me great advices all time and keeping join me watching me stay up all night. To my parents, grandmother and brothers, because they have been one of maximum motivations to move forward in this master. Thanks to the rest of my family for understanding my absence on special dates, celebration parties, birthdays, travels, etc. and always keep praying for me. Also, I would like to thank my thesis director for knowledge transmitted, for believing in me from the beginning and because the door of his office always was opened to clear up any doubts. To my coworkers and friends, for sending me all their good energies and desires. And accompanied in this way. Thank to Laura Guerra Guerra for giving me encourage in the keys moment. To classmate Dominga Jimenez, for being my battle partner. To Cesar Redondo Manjarrez for giving me time to review the syntax of investigation articles performed during the thesis.

Contents

List of Figures

List of Tables

Chapter 1.
Introduction

Nowaday, there are many services that depend on internet network access, these services are delivered to end user through internet connection, all this is possible due to rising and developing network technology. For example services like banking, electronic commerce, social networking and newsgroup are loaded in internet cloud [79], such services are available trough websites. The internet cloud are a configurable computing resources environment that offers flexibility and help to reduce the hardware and maintenance cost, this is what we call cloud computing [19]. In this environment exist different vulnerabilities that deliver opportunities for web attackers [2], allowing them break the security polite.

DDoS (Distributed Denial-of-Service) is one of the most powerful type of attack. DDoS is having the power of disrupts and degrades the normal performance of web servers or host victims. Morover, it can prevent the user legitimate useraccess to specific service [75]. The DDoS attack decrease available machine resources, so that it cannot respond to any request made by authenticated user [38], these attack have been known by the network research community since 1980. In the summer of 1999 the Computer Incident Advisory Capability published the first Distributed DDoS at-

tack incident [15] since then every attacks have been of this type. when an attacker decides launch an attack, it can affect the victim as follows: The attacker can find some bug or weakness in web applications to break service off. He can also deplete all bandwidth or resources of victims system [19].

DDoS attacks apply the same techniques of DoS attacks, but the difference is that DDoS are performance in high scale through botnets as shown in Figure 1.1.

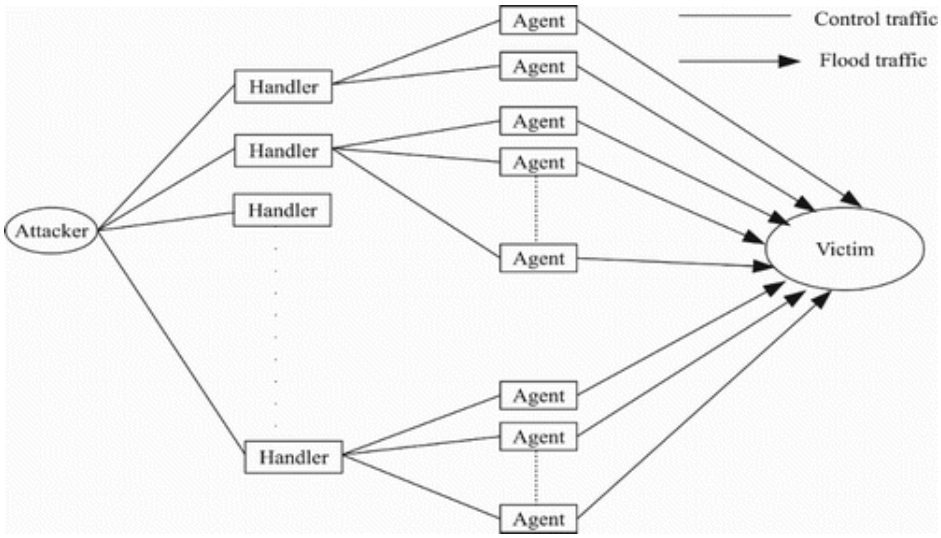

Figure 1.1: Modus Operandi of DDoS Attacks [22].

The botnet is a computer pool connected to internet which is controlled by a intruder (boot-master) and infected via malicious software called bots. There are many different reasons by which they are created: perform DDoS, spreading SPAM, conducting click-fraud scams or stealing personal user information. In addition, is important to know that the bots interact through legitimate communication0 channels as internet relay chat (IRC). Once the host is infected by bots, it will locate and connect with a IRC server, then the boot-master will use the IRC command and

control (C/C) channels to communicate and control the bots [77]. The IRC is an example of traditional botnet communication protocol to accomplish command and control once the computer target is infected [62].

Every computer connected to the internet can become an attractive target for attackers for making bots or zombies, the computers are infected through the use of worms, back-doors or Trojan horses by sending an e-mail content, a captivating link or a trust-inspiring sender address to the vulnerable machines [75], Often the request originated by a single bot is not enough, but the massive traffic sent by high number of bots is sufficient to deplete and exhaust the resources.

1.1 Justification

Many users around the world keep connected to the internet from their smart devices for the purpose of access to all digital content located on the web, for instance: social media, video platforms, electronic commerce platforms, video games community, employment, etc. Due to that, the companies have understood the potential of the digital market and for this reason, they have begun to compete in this new niche market in intense way. Launching new products and services every day to reach new customers. However, To accomplish that, the companies support all their digital strategies making use of cloud computing [79]. This is an environment of configurable computing resources that offer flexibility and it helps to reduce the costs of hardware and maintenance of a company since the company's network infrastructure

would almost be in a virtual environment [19]. In this environment, there are located many services and applications that can be accessed through Internet but in the same way, there are different vulnerabilities or weaknesses so that, the attackers can take advantage to launch a cyber attack of great impact such as DDoS attack. Because the attackers have the same opportunity to access digital platforms as legitimate users. Furthermore, taking into account that the communication protocols also presents several lacks. Among the protocols more used is TCP/IP [2]. It allows intruders to undermine the security policy of one network. The DDoS attack is the principal concern of companies in the world. because it can be capable of preventing the normal performance of either their digital platforms or server hardware. The proliferation of this type of attack today is imminent due to the exponential technological growth that develops in the 21st century. Year per year, this type of attack continues growing up as shown in Figure 2.1, from 2005 to today the amount of data sent per attack has increased significantly, reaching 1,2 terabytes (1200 Gbps). Thus, it is necessary to implement an intrusion detection system that is able to analyze, neutralize and eliminate the different threats that come from the internet.

1.2 Objectives

Having in mind the problem stated above, we have set the objectives for this thesis work.

1.2.1 General Objective

Identify the range of successful denial of service attacks in a data network using techniques based on machine learning.

1.2.2 Specific Objectives

- Select an appropriate data classification technique based on the review of the state of the art, in order to allow the grouping of network traffic into different categories.
- Consolidate a data set with defined behavior patterns that allow train and test the classification model based on the machine learning technique selected.
- Develop a machine learning classification model that fits 95% in predicting categories for new input variables.
- Identify the percentage of successful classification of the selected data classification model, through the cross-validation technique to verify the fit of the model.

Chapter 2.
Theoretical Framework

In this section, we will talk about DDoS attacks in general form beginning from the brief description and background about them, attackers' motivation, how is conformed the DDoS architecture and strategies to launch attacks, vulnerable network security protocols to attackers, defense mechanisms and DDoS tools for attacking, and techniques based on either statical and data mining used for preventing DDoS attack.

2.1 DDoS Generalities

2.1.1 Background

In this part, will present the history of DDoS attacks year after year. Initially,in the first quarter of 2013 was reported an increase the broadband average in 48,25 Gb/s by attack which is 718% more as compared to the last quarter of 2012 [51]. Since 2008 to 2013, the attacks has been increased in 1000% from 40 Gb/s to 400 Gs/s. These occur at an average rate of 3000 times at day. According to a survey by Verisign the attacks has growthed in 111%. A very relevant event occurred in October 2016. It was an attack launched using a new network mirai robots which implemented 100.000

agents against American company server called *"Dyn Mirai"*. The robots infected violated the security and affected several internet of things devices (IoT), such as: digital cameras, digital video recorder (DVR) reproductors in order to interrupt the services of companies, for example: Twitter, Netflix, The Guardian, CNN, Reddit. The report was published by DYN DNS, and this indicated that the attack potential was of 1,2 Terabits (1200 Gb/s) [38]. Futhermore, The report delivered by Dyn, the attack had a potent attack strength of 1,2 terabytes (1200 Gbps) per second, it had included 100000 malicious agent's, as shown in the Figure 2.1. has gone increasing [38].

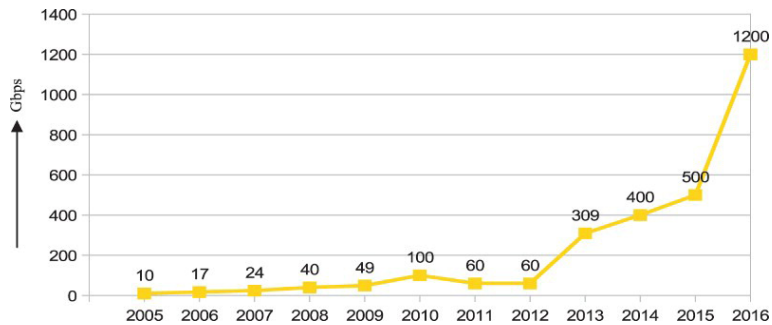

Figure 2.1: Strength of DDoS Attacks in Gbps [38].

2.1.2 Motivation

DDoS attack are difficult to detect and very dangerous as the traffic looks like normal traffic [76], on the other hands High-rate DDoS (HDDoS) attacks are easily recognized with detection methods.

Due to multiples weakness and vulnerabilities in Internet protocol, web applications and operative system make it easy to perform an attack. Surely, there are some

reasons for attacking: hactivism, extortion, personal reason, economic reasons and politic reason [56]

Incentives for attacking are [75]:

1. Financial/economical gain: the attackers who belong to this category are the most dangerous, because of these people are the most technical and the most experienced. They represent the major concern for companies.

2. Revenge: the attackers who belong to this category have suffered a personal conflict with someone, they look for a revenge through cyberattacks, possibly they have lower technical skills.

3. Ideological belief: attackers in this category are motivated by their belief, this category represents one of the major incentives for attackers to launch DDoS attack. for example these incentives have led sabotages in Estonia 2007, Iran 2009, WikiLeaks 2010.

4. Intellectual Challenge: To this category belong young people, who simply want to show off their skills and at the same time experiment to get more knowledge about how to perform DDoS attacks. Today there are different tools with which a novice can perform attacks.

5. Cyberwarfare: the attackers in this category are motivated by political interests; usually they belong to a military organization or terrorist organization, they have a very good technical skills. The power of this group of attackers

is widely powerful as they spend long time and resources to perform this activities. The impact of attack can paralyze different sectors or companies such as: executive civilian departments and agencies, private/public financial organizations (e.g.,national/commercial banks), energy/water infrastructures (e.g., [28]), and telecommunications and mobile service providers.

2.1.3 DDoS Attacks Architectures

As stated in [65], Distributed Denial of Service attack (DDoS) is an attack very well coordinated, where the attacker uses multiples infected computers to disrupt the normal performance of network. The author of attacks uses the client/server technology which allow to increase the effectiveness of the DDoS attack significantly, as it is showed in the Figure 2.2.

DDoS are different from other attacks due to the ability to deploy the threat in a distributed mode over the Internet, instead of breaking the victim's defense system for pleasure to show the skills. DDoS attacks aim to create chaos on a victim either for personal reasons, personal benefit or popularity [8]. To understand the taxonomy of DDoS attacks, is important to know a little about Internet architecture, at the beginning when the Internet were being designed, the first concern was to provide functionality not security. As second effect, many security issues have been raised and the attacker have taken advantages of it, thus the Internet design open several security issues that are taken advantage by them, these are some opportunities for launching distributed denial of service attacks [49]:

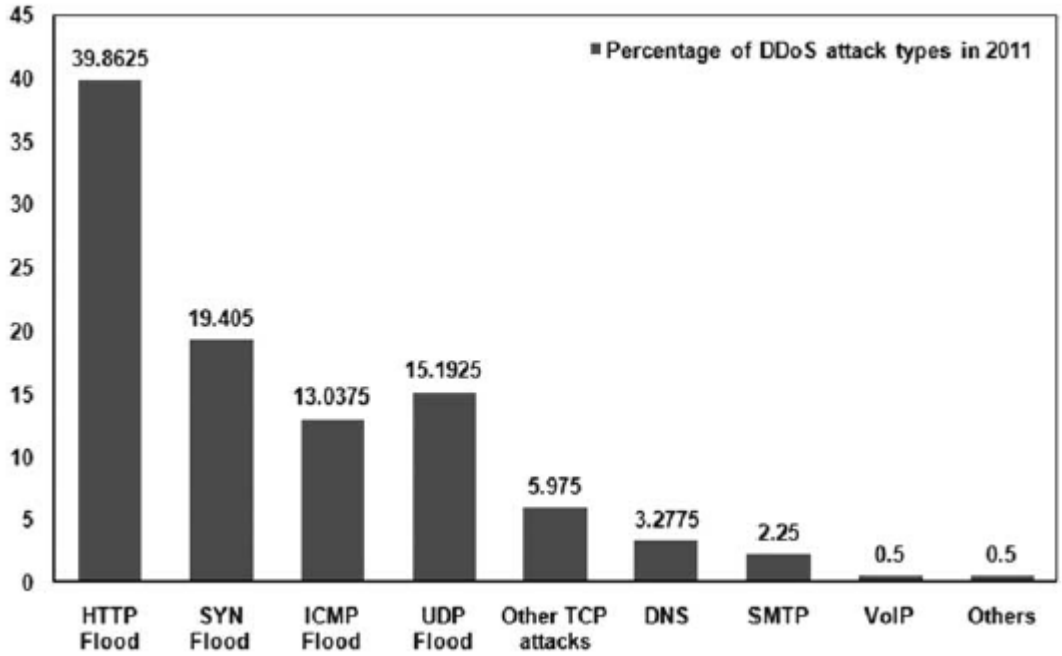

Figure 2.2: DDoS Attack Based on Protocols [8].

- Internet security is highly interdependent: The security of victims always depends on the global Internet security. The Internet was designed for functionality, thus, it does not matter how well the victims security is, it all depends on Internet communication security system.

- The Internet has limited resources: Every Internet entity such as host, network or service has limited resources that can be consumed by large number of users.

- Many against a few: If the resources of attackers are greater than the victims' resources, the attacks will be one hundred percent almost certain.

- Many times, the source address fields of packets are not validated, for this reason it is possible for attackers to spoof source address. the attackers can

take advantage of this by implementing a great mechanism of escape and evade their responsibilities

- The Internet security control is not unified, because all networks on the Internet work based on their own private policies. Thus, it is difficult to keep a very good Internet security global system.

2.1.4 DDoS Attack Strategy

There are several phases that must be taken into account before launching a DDoS attack. Firstly, the process begins when the attackers recruit hosts of target networks, searching vulnerabilities to exploit their security system, once the hosts are located and their systems are exploited, the machines are infected with a malicious code that allows the attackers remote control of machines infected, then those hosts are used for launching attack to the victim[49]. In this way the DDoS can greatly cause powerful damages in network connectivity.

This process involves component such as master, handlers, agents and victims. The agents are controlled by a master who communicates with them through handlers that are malicious programs planted into agents. Through handlers, the attacker sends commands to execute different actions and can verify which agents are working as showed in the Figure 1.1. Using several protocols such as Internet control message protocol (ICMP), TCP or UDP, attackers can connect to the handlers [19].

A bootnet attacks can be implemented in several ways, it all depends on the

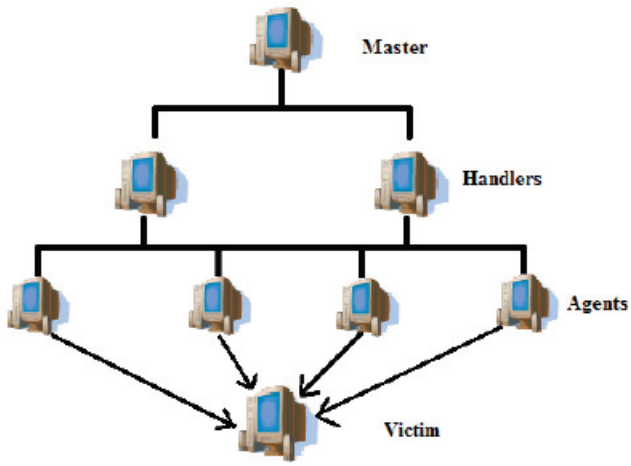

Figure 2.3: Constituents of DDoS [19].

manner with which the botmaster controls the agents. Ten years ago, the attackers would have had to implemented a multi-user, multi-channel chatting system known as IRC [68]. This is because the master (Attacker) can send instructions through the command to agents, preventing the easy trace of attacks. One of the advantages of using an IRC channel, is that allows the attackers to hide their identity due to a large volume of requests the IRC server receives. On the other hands, the principal problem when the attack process is centralized through IRC server, is that once the defender device locates the C&C (Command and Control) server, the attack can be blocked quickly and easily and the boot-net will be shutdown [75].

2.2 Classification of Remote Denial of Service Attacks

The DDoS attacks could be classified depending on the attack protocol level as showed in Figure 2.4.

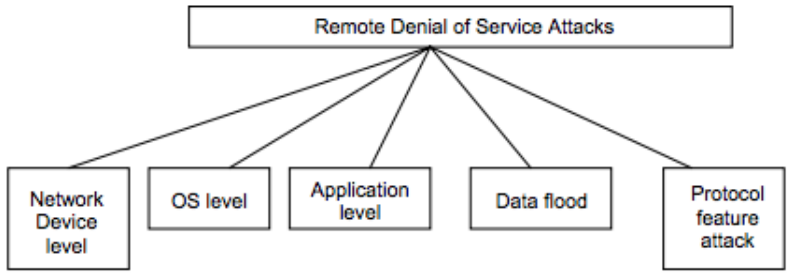

Figure 2.4: DDoS Clasification Depending on Protocol Level [19].

2.2.1 Network Device Level:

The attackers can take advantages of weaknesses of network devices software to perform DDoS attacks and can be able to deny the corporate network Internet access. The network devices level attack aims also include the deplete the hardware resources of devices, such devices can be defined as routers [22]. For example, by typing large passwords in the buffer, certain Cisco 7xx routers can be overrunning, thus approach can be used to break Cisco 7xx routers down [37].

2.2.2 OS Level

Another way that the intruders can take advantages over the victims is the way operative systems implement protocols. For instance the Ping of Death attack, which

has a much larger size compared to the maximum IP standard size can easily overrun the memory buffer [1].

2.2.3 Application-Based Attacks

Network applications scan are used to discovery network applications weaknesses, taking advantage of this, the intruders can violate the network devices applications security polices. For instance, in this category we can find the application-based attack [2].

2.2.4 Data Flooding

Once the victims are identified, the attacker is prepared for launching the attack, his major goal is flooding the host or device network by sending a big number of entries to the victim target, causing exhaust of hardware network resources. We can considered three main types of flooding: Amplification attacks, oscillation attacks, and simple flooding. The most popular flooding attack is SMURF attacks. In this attack the author spoof the source address of ping packet then specify destination IP as broadcast. Thus, the victim is flooded with pings from every computer on the amplifier network. A frequent oscillation attack uses UDP (User Datagram Protocol) protocol running in port 9 [37].

[1] http://www.insecure.org/sploits/ping-o-death.html
[2] http://service.real.com/help/faq/servg260.html

2.2.5 Attacks Based on Protocol Features

The most common attack to keep a server unavailable is TCP SYN flood attack. There is a normal communication between host and server when connection based protocol are used such as TCP. After the server receives a SYN packet from legitimate client, the server should respond to client sending SYN/ACK packet to client, finally the client should respond replying SYN packet to the server to finalize the communication or TCP three-way-handshake. One of difficulties of TCP connection is the fact of server has to keep connections half open. These half-open connections indicate that the server is waiting for a response from the client to terminate the connection and complete the greeting in three ways. In this sense, attackers can abuse of this server state to keep many half-open connections. Attackers can generate requests with packets with falsified source addresses so that the ACK packet expected by the server never arrives, thus the mechanism of the 3-way handshake is never completed [60]. The TCP SYN attack is represented in Figure 2.5

2.3 Defense Mechanisms and DDoS Tools for Attacking

Detecting DDoS attack is hard work because the distributed attacks are each day more dynamic. The DDoS defense mechanics can be classified in four types according to the activity deployed [22]:

1. Intrusion Prevention

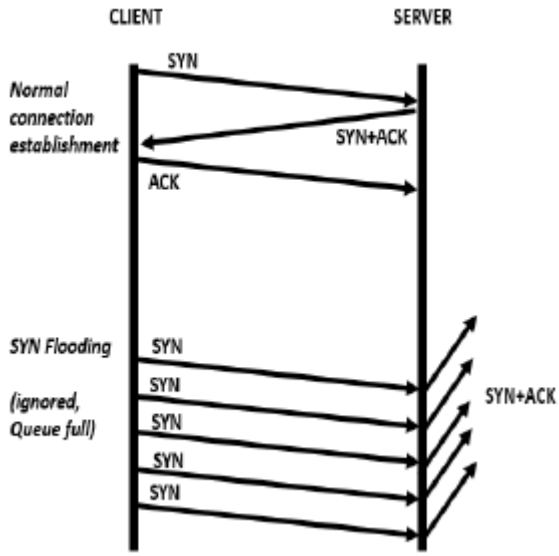

Figure 2.5: TCP SYN Flood Attack Process [60].

2. Intrusion Detection

3. Intrusion Tolerance and Mitigation

4. Intrusion Response

2.3.1 Intrusion Prevention

Intrusion Preventions try to stop all well known signature based and broadcast based DDoS attacks before they can cause damages, also it is responsible for protecting and ensuring the security of hosts connected to the network, as well as constant scanning on computers with protocol weaknesses, poorly implemented session starts and equipment that may be being used to generate DDoS attacks [29].

Attack prevention schemes not always are sufficient because each day appears novel and mixed DDoS attack types which are not registered in signatures and patch

databases. The mayor concern of instruction prevention is detect an ongoing attack and to identify malicious traffic from legitimate traffic, this detection can be executed by signatures-based detection system. The database signature is created step by step by security administrator analyzing previous attacks which use the incoming traffic to compare with attacks signatures databases, for instance SNORT [1].

2.3.2 Intrusion Detection

The last years, Instructions detection researches has been increasing, due to an increment of DDoS attacks to networks, clients and servers. Intrusion detection systems can identify DDoS attacks either by using the database of known signatures or by recognizing anomalies in system behaviors. Intrusion Detection focuses on detecting behaviors that are abnormal in comparison with normal standard behaviors.

2.3.3 Intrusion Response

By employing automated intrusion response systems the security and communication system can be reinforced and the response time will be even more efficient, since once the attack is launched, the system will send an alert to network administrator, in this way the administrator will perform the attack block in the system. Automated intrusion response systems are deployed only after a period of self-learning (for the ones that employ neural computation in order to discover the DDoS traffic) or testing (for the ones that operate on static rules) [22].

2.3.4 DDoS Tools for Launching DDoS Attack

Attackers often have different motivations for DDoS attacks, either because of personal interests or outside interests. Many times what is sought is to experiment and test data systems to validate if the security system is optimal.

Here, we can see different tools used for performing DDoS attacks.

Trin00 is widely used for performing attacks through User Datagram Protocol (UDP) protocol against one or different target source destinations, causing UDP flooding attacks and depleting bandwidth resources [15]. Also found, Tribe flood network (TFN) which is an application that implement a command line interface to establish link between boot master and target server. the protocol used in this applications are for perfoming attacks are TCP, ICM and UDP for causing flooding and deplete resources of target [21].

2.4 Types of Intrusion Detection System

Due to the increase in cybernetic network attacks, researchers have proposed a mechanism to control such attacks, as they represent a financial threat to companies offering services over the Internet. Among the different mechanisms that exist to neutralize this type of attacks are the intrusion detection systems (IDS).

Intrusion Detection Systems (IDS) are a tool that allows the identification of malicious activities within the traffic of a network [31]. The process of stopping intruders consists of carrying out an exhaustive monitoring of the traffic that passes through

the network to detect signs of violation of security policies [53].

There are two main types of Intrusion Detection System (IDS): Signature Based IDS (SBIDS) and Anomaly Based IDS (ABIDS) [55].

SBIDS is well known as it stop misuse, known attack signatures are stored in a rule file. When analyzing network traffic with this tool, anomalous events are compared with the signature database to find matches and generate an alert. The main drawback with this method is that our SBIDS will only be able to identify attacks that are stored in the signature database [31].

The number of interested researchers in ABIDS has increased significantly because this mechanism has the power to identify novel and strange attacks using machine learning algoritms [47].

ABIDS have two important advantages over signature based intrusion detection systems. Firstly the ability to indentify known attacks, for instance "zero day" attacks. Anomaly detection systems have the power to take as reference the normal network working. Finally that the afore mentioned profiles of normal activity are customized for every system, application and/or network [55].

2.4.1 Statistical

Also known as Anomaly Based Intrusion Detection System (SABIDS). Statistical modeling is the approach most used for detecting intrusions in data systems networks. The method work implementing statistical properties and performing statistical tests to determinate if observed behavior has abnormal properties or whether the observed

behavior does not match with the behavior expected [58]. As stated before, this method uses statistical properties as mean and variance to define normal profile, to then implement a test to determinate whether behavior identified from network traffic is different or equal from the expected behavior. This is made possible by assigning a score to abnormal activities, in this way when the IDS detect abnormal activities, it sends an alert to network administrator. In addition these behavior profiles are reviewed based on various techniques to detect any kind of deviation from the normal behavior. SABIDS can further be classified into following categories [17]:

- Operational Model or Threshold Metric: this model is based on the assumption that the 'X' observations obtained are analyzed and compared with defined parameters. This model is popularly applicable to metrics where experience has shown that data can behave abnormally, for example: by using an event counter to report the number of times you try to enter a password in a short period of time, where you can also define that for 10 attempts could launch an alarm event [17].

- Cases in which the Operational model or Threshold Metric works: this sub model is a good choice in two cases [58]: when there are not sufficient changes in data normal behavior and When the tolerance parameter is previously identified. Disadvantages of Operational model or Threshold Metric: they are not efficient to identify anomalies with more than one event.

- Markov Process Model or Marker Model: This can be defined as stochastic process [73]. This model is based on previous events to determine the legitimacy of present events, using an event counter metric to ensure this validation. This model classifies each observation made as a specific state and uses a transition matrix to determine the level of probability of an event such as high or low. This model is said to be particularly useful when the sequence or behavior of activities is known.

  For this model, knowing the behavioral pattern of an activity is very important but if the squence of steps changes, the model would begin to lose consistency [10].

  Therefore we can consider that this model is not good enough to work with events that are deployed in real time due to the amount of patterns that can be evidenced [55].

- Statistical Moments or Mean and Standard Deviation Model: this model basically configures a confidence interval on the basic statistical properties so that if an event is analyzed and classified outside the range stipulated by the system, it is considered as an event with abnormal properties.[36]

  One of the advantages of this method is that it can "learn" the behavior of a user without prior knowledge of it [31].

- Multivariate Model: this model is usually similar to the model of means and

standard deviation, differing from the previous one in that it is based on the correlation that exists between different parameters or factors. Therefore, we can affirm that this model is very precise and useful in cases where the parameters that affect the variability of the variable are correlated. For the implementation of this process it is also required to consume too many computational resources [58].

The mutivariable model as well as multivariate cumulative sum (MCUSUM) and multivariate exponentially weighted moving average (MEWMA) are usually used to detect anomalies in manufacturing processes [69].

- Time Series Model: To detect anomalies, this model is based on the order and time interval in which events occur in the network. This model classifies events that have a low probability of occurring as anomalies.

  This model puts as a main priority the sequence of the activities therefore this model can be a good choice when the order of the behavior of the attack on the network is known [10].

2.4.2 Knowledge Discovery in Databases and Data Mining

Knowledge Discovery in Databases is an interactive and iterative process between a human and a database that may involve background knowledge of the analyzing domain expert. Large amounts of data can be not analyzed only using data mining or machine learning algorithms because, each data set must be examined by a human

involved in the knowledge discovery process [66].

The KDD process is composed of five steps [61]:

1. Selection: whose aim is selecting the proper data samples in order to understand the application domain and identify the target data used in the knowledge discovery process.

2. Preprocessing: in this phase, the data chosen previously will be is processed in order to allow perform analysis posteriors. Other decisions are taken in this step include the handling of missing values, the identification (and potentially correction) of noise and errors in the data, the elimination of duplicates, as well as the matching, fusion, and conflict resolution for data taken from different sources.

3. Transformation: the third step consists of generating a new data subset with a new projection in a way that data mining algorithms can work on in most cases.

4. Data mining: when the data is present in a suitable format, the next goal is the search of the method that help us to work with the data transformed, such as classification, regression, or clustering. Moreover, this step involves deciding which models and parameters might be appropriate to be used (for example, models for categorical data are different than models for numerical data). Once the data mining method and algorithm are selected, data mining takes place

searching for patterns of interest in a particular representational form.

5. Evaluation and interpretation: this is the last step of the KDD process, the patterns and models extracted by the data mining algorithms are examined with respect to their validity. Furthermore, the user evaluates the utility of the found knowledge for the models and parameters selected. The person in charge can make use of visualization techniques in order to represent the knowledge acquired, so that they can take a decision about the result observed.

Data mining is well known as the computational process of analyzing big amounts of data with the purpose of extract patterns and suitable information. In the last few decades, data mining has been recognized as a powerful and versatile tool that makes possible the data analysis in a variety of fields: information technology, clinical medicine, sociology, and physics [14]. The term data mining is currently used by statisticians and database researchers [28]. Moreover, Data mining is one of the intricate phases included within the Knowledge discovery in databases (KDD) process. KDD process is the non-trivial process of identifying novel, valid, potentially useful, and ultimately understandable patterns in data. On the other hand, data mining represents the core step of the KDD process. Currently, there is a wide variety of mining algorithms that are widely used in different fields, such as statistics, recognition patterns, automatic learning machines, and databases. Basically the mining of data is a mixture of three components as follows: the model includes two fundamental factors the function and form to represent the model, the preference criterion which

depends on the data set you are working with, and the search of the algorithm where we have to specify the algorithm with which we will work [27]. This is method works better when it combines with artificial intelligent or machine learning techniques. Data mining methods are categorized into clustering, classification and associative rule mining based on detection.

- Clustering: Clusters are defined by finding natural groupings of data based on similarity metrics or probability density models. This is a technique within the discipline of machine learning whose main objective is to create groups of data or clusters that present similarity measures [27]. The elements that do not belong to a group are considered as abnormal threats or activities [5]. Among the advantages offered by this technique we can see that it is a technique that can easily work in unsupervised mode, can often be adapted to other type of complex data using clustering algorithms that can handle specific types of data and the test phase of this technique is quite fast [12].

- Classification: the intruder detention problem can be seen as a classification problem, where an instance can be assigned to a specific category. This method is one of the most used techniques in data mining, it is practically based on algorithm training with a training data set so that the system can predict or classify future data instances based on the training data. This prediction is achieved by building a decision tree [24].

- Association rule discovery: it is a very popular technique but very little used. Today it is being replaced by other data mining techniques such as classification and clustering [31]. This method finds anomalies based on the correlation of attributes is also based on rules of Boolean association to find particularities between different entities, as they analyze large amounts of factors or tributes the process become too slow [31].

2.4.3 Machine Learning Based Detection

Machine learning is an extension of artificial intelligence, among the best known techniques of machine learning are found: bayesian decision theory, multilayer perceptrons, clustering, classification trees, hidden markov models, among others [70].

Machine learning is understood as a set of algorithms capable of learning by themselves through a set of data that contains necessary information about a particular event. Algorithms based on these techniques have the ability to improve their performance and execution strategies during the development of tasks, in order to obtain more optimal results [55].

- Neuronal Network (ANNS): Artificial Neural Networks are a very old field of research in data science computing. Projects carried out with this technique began in 1940 and today are available for different purposes. Artificial Neural networks (ANNS) is a machine learning technique that has the ability to learn from a set of previous events to predict future events. Currently considered a powerful classification technique, this method is now being used to replace

statistical methods employed in expert intrusion detection systems. The results of implementing this technique are seriously comparable to signature-based intrusion detection systems [16]. The Artificial Neural network (ANNS) have the ability to classify intelligently and automatically DDoS attacks and cope with changes environments [44].

Different neural networks algorithms can be implemented for anomaly based detection, such us, Multi-layered Perceptrons, Radial Basis Function-Based, Hopfield Networks, etc [31]. For example, in [43] author implemented Linear Vector Quantization (LVQ) model of ANN for detecting DDoS attacks after testing using the same dataset the author performs tests with other model of Artificial Neural Networks (ANNS), such as: Back propagation (BP) model of ANN for comparative study. Other techniques used for identifying DDoS attacks are: Replicator Neural Networks (RNN), Linear Vector Quantization Artificial Neural Networks (LVQ-ANN), Back Propagation Neural Networks (BP-ANN), Resilient Back Propagation (RBP) Neural Networks and Time Delay Neural Networks (TDNN) are used for anomaly-based DDoS attacks detection [56].

Basically, three components are taken into account in order to build an intrusion detection system based on neural networks: The training data are obtained from audits performed on particular network systems. Training of the neural network must be performed with a part of the data obtained from the network audits, finally the neural network identifies users based on the vector distribu-

tion of data and the second part of data is use for performing neural network testing [31].

- Bayesian Networks: Bayesian networks are used in problems that contain uncertainty [34]. They can be defined as an acyclic graph, where each node represents a random state interest variable. Each node represents the state of the random variable and a condition probability table (CPT). The CPT of a node contains probabilities that the node is in a specific state given the states of its parents [40]. Bayesian networks are potentially efficient when applied to data science problems, they are also widely used along with statistical techniques to solve intruder detention problems. Among advantages we can find the following: The situation where exists missing data is easily controlled by this technique and it brings an efficient approach for avoiding the over fitting of data [32].

- Support Vector Machine (SVM): this method was proposed in 1998 by Vladimir Vapnik. Support Vector Machine (SVM) aim to find the better hyper-plane that be able to separate the data set into different class, taking into account that the better hyper-plane found is the hyper-plane that allows it to trace the distance or margin maximum between hyper-plane and data point. To accomplish it, the Support Vector Machine (SVM) takes the training data obtained from the basic input space into a higher dimensional feature space using kernels and then gets the most favourable isolating hyper-plane or a decision boundary in the form of support vectors. The Support Vector Machine (SVM) allows

to configure a parameter called penalty factor. Witch allows users to make a trade-off between the number of miss-classified samples and the width of a decision boundary [31]. This method is considered better than Artificial Neuronal Networks (ANN) and clustering methods, taking into account the speed and accuracy. It offers a robust detection and handles the missing data and over-fitting problems effectively [26].

- Classification Tree: A classification Tree is a type of prediction in machine learning, and it is also is well known as a "Decision Tree". It is a tree pattern graph similar to flow chart structure; any internal node is a test property, each branch represents test result, and final nodes of leaves represent the distribution situation of various types. There are two common algorithms used for classification trees, such as ID3 and C4.5 [59]. Moreover, there are two methods for tree construction. top-down tree 298 construction and bottom-up pruning, and both ID3 and C4.5 belong to top-down tree construction; their algorithm is described as follows [41]:

    1. All paradigms of training data are located in the root of the classification tree.

    2. If there are node does without data or the data at the node belongs to the same type, the node becomes empty leaf or all paradigms leaf of the same type; if the node contains more than one type of paradigms, it is necessary

to assess all properties of data, according to certain assessment function, and a proper property is selected. According to the value of the property is necessary to apply a process called the splitting node. It consists as follows: paradigms at the node are divided into N parts, and each part becomes a new node connecting the root node.

3. After realizing the splitting of nodes, judge whether or not these new nodes are leaves; if not, new nodes are the root of sub-tree and used to construct a new sub-tree.

4. The mentioned steps proceed continuously with the recursion method until all new nodes are leaves.

- Markov chains and Hidden Markov Models (HMMs): These models make part of the category of Markov models. A Markov chain [48] is a set of states interconnected through transition probabilities that define the topology of the model. An HMM [6] is a statistical model where the system being modeled is assumed to be a Markov process with unknown parameters. The principal challenge is based on determining the hidden parameters from the observable parameters. The HMM contains several states which represent unobservable conditions being modeled. By having different output probability distributions in each state and allowing the system to change states over time, the model is capable of representing non-stationary sequences.

2.5 Feature Subset Selection

Feature subset selection works with the purpose of removing features that are not relevant or are redundant. The subset of features selected should follow the Occam's Razor principle and also improve the according to some objective function [33]. In other words, feature subset selection is the proceedings of selecting the best features among all the features that are suitable to recognize classes. The feature selection algorithm (FSA) is defined as a computational model that is provoked by a certain definition of relevance [39].

Feature selection offered the following advantages [42]:

- It reduces the dimensionality of the feature space, and increase algorithm speed.
- It removes redundant, irrelevant or noisy data.
- The immediate effects of data analysis tasks are speeding up the running time of the learning algorithms.
- Improving data quality.
- Increasing the accuracy of the resulting model.
- Performance improvement, to gain in predictive accuracy.
- Data understanding to gain knowledge about the process that generated the data or simply visualizes the data.

Among characteristics of Feature selection algorithms, we can find (i) search organization: three types of search are possible exponential, sequential, or random. (ii) Generation of successors (subset): five different operators can be considered to generate a successor, such as forward, backward, compound, weighted, and random. (iii) Evaluation Measure: evaluation of successors can be measured through the probability of error, divergence, dependence, interclass distance, information or uncertainty and consistency evaluation.On the other hands, the Feature selection algorithms are separated into three categories as follows: [9]:

- The filters which extract features from the data without any learning involved. Filters operate independently of the classifier. This makes them very computationally efficient. They fall into in two types which are multivariate and univariate methods. Multivariate methods can be able to find relationships among the features, while univariate methods having an account each feature separately.

- The wrappers that use learning techniques to evaluate which features are useful. However, wrapper methods are more expensive to be implemented for large feature space because of that each feature set must be evaluated with the classifier that ultimately makes the feature selection process slow.

- The embedded techniques which combine the feature selection step and the classifier construction. Embedded techniques work better computationally than

wrappers. However, they perform selections according to a classifier that might not work with any other classifier. That is because the optimal set of features is acquired when the classifier is constructed and the selection is affected by the hypotheses defined by the classifier.

It is necessary to consider the following aspects when it comes to choosing a feature selection approaches Starting Point, Search Strategy, Subset Evaluation, and Stopping Criteria. On the basis of these aspects comparative analysis of feature selection techniques is shown in Table 2.1. Here, the feature selection techniques are characterized according to each feature selection technique with the purpose of guide the choice for a technique suited to the goals and resources of practitioners in the field.

2.6 Features Extraction

Features extraction realizes some transformation of original features to produce other features that are more significant. Brian Ripley [78] defined feature extraction as follows "Feature extraction is generally used to mean the construction of linear combinations of continuous features which have good discriminatory power between classes". One of the typical problems, either neural networks research and other disciplines like Artificial Intelligence consisting of finding a suitable representation of multivariate data. Thus, feature extraction can be used in this context to reduce complexity and give a simple representation of data representing each variable in feature space

Table 2.1: Comparative Analysis of Feature Selection Algorithms [33].

Method	**Type**	**Supervised**	**Linear**	**Description**
T test feature selection	Filter	No	Yes	It finds features with a maximal difference of mean value between groups and a minimal variability within each group.
Correlation-based feature selection (CFS)	Filter	No	Yes	It finds features that are highly correlated with the class but are uncorrelated with each other.
Bayesian networks	Filter	Yes	No	They determine the causal relationships among features and remove the ones that do not have any causal relationship with the class.
Information gain (IG)	Filter	No	Yes	It measures how common a feature is in a class compared to all other classes.
Genetic algorithms (GA)	Wrapper	Yes	No	They find the smaller set of features for which the optimization criterion (classification accuracy) does not deteriorate.
Sequential search	Wrapper	No	No	The heuristic-based search algorithm that finds the features with the highest criterion value (classification accuracy) by adding one new feature to the set every time.
SVM method of recursive feature elimination (RFE)	Embedded	Yes	Yes	It constructs the SVM classifier and eliminates the features based on their "weight" when constructing the classifier.
Random forests	Embedded	Yes	Yes	They create a number of decision trees using different samples of the original data and use different averaging algorithms to improve accuracy.
Least absolute shrinkage and selection operator (LASSO)	Embedded	Yes	Yes	It constructs a linear model that sets many of the feature coefficients to zero and uses the nonzero ones as the selected features.

as a linear combination of the original input variable. There are two categories for feature extraction algorithms: linear and nonlinear. The difference between linear and nonlinear problems is shown is Figure 2.6.

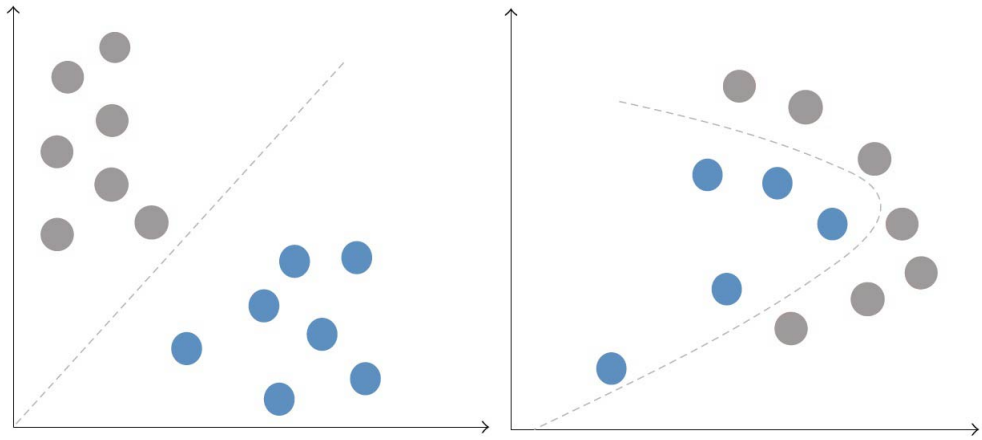

Figure 2.6: Linear and Nonlinear Classification.

The feature extraction approach most used is the Principle Component Analysis (PCA) introduced by Karl. Actually, there are many variants of PCA proposed. PCA is a simple non-parametric method used to extract the most relevant information from a set of redundant or noisy data. Also, PCA can be defined as: a linear transformation of data that minimizes the redundancy (measured through covariance) and maximizes the information (measured through the variance) [11].

Chapter 3.
State of the Art

Before starting the development of this work is necessary to perform a review of the works recently proposed by researchers in the network security field, and thus, to knows the most popular techniques used for detecting cyberattacks.

3.1 Intrusion Detection Using Neural Networks and Support Vector Machines

Mukkala [52] proposed the use of machine learning techniques for building an intruder detection system based on two machine learning algorithms, either support vector machine (SVM) and neural networks. The data set chosen for training and test the algorithms is the DARPA data set which contains 41 features and it is the first version of the KDD-1999 data set. The training and testing of each algorithm were similar. These worked processing the 41 features. Thus, the performance of the SVM and neural networks model achieved a great score very near to one. But also, but the performance of the model decreased so that, the time response increased notatly. In the SVM training and testing was implemented the radial bias kernel function (RBF) which defines the feature space in which the training set examples

will be classified [35]. During training, only 6 data points from the 7312 training set were misclassified. Finally, for the testing, SVM has used 6980 data points to achieve an accuracy of 99.50%. The training of the neural networks was conducted using a feed-forward backpropagation algorithm using the scaled conjugate gradient descent or SCG for learning. Three feed-forward neural networks were used with the following architectures: Network A: 4 layers, Network B: 3-layers, and Network C: 3-layers. The purpose of having multiple networks was to find a suitable architecture that can detect at a faster speed with a low error rate. The testing of each network showed great results. For example Network A performed with an accuracy of 99.05%; network B achieved an accuracy of 99.25%; network C performed with an accuracy of 99%.

3.2 Category-Based Intrusion Detection Using PCA

Unlike Mukkala, Gholan [74] also experimented with the DARPA data set to detect HTTP Flooding Attack using the KNN (k-nearest neighbor algorithm) classification method for the classification of the attacks. In this work, he does not use all their features for training and testing the machine learning algorithms. He proposed a method that allows examine all their attributes and make a filter for searching the features that apport most value in the decision process. PCA method was used to determine an optimal feature set to make the detection process faster. The results of accuracy acquired were between 88.5% 99.22% which is very relevant and very

similar to the obtained in [52]. This was because the number of features used was the features extracted in the PCA process. Moreover, is important to mention that the data included in DARPA is divided into four groups of attacks, such as Denial of Service (DoS), Remote to User attack (R2L), Remote to User attack (U2R), and Probing attack and for each attack the author identified the most relevant features, something similar to what was done in this thesis. Finally, it is concluded that not all features have a contribution to intrusion detection and that normal state of the network and category of the attacks can be identified using a small number of features.

3.3 Anomaly Detection for Application Layer User Browsing Behavior Based on Attributes and Features

The DDoS attacks exist in various forms as for instance HTTP flooding attack which is oriented to layer application. It has the main objective to consume the server hardware resources to avoid the connection of legitimate users by sending millions of requests per second to a victim server [71]. Xiong [45], proposed a way to combat this type of attacks through machine learning algorithms training. For this, a data set that represents either the behavior user request and server response when these two are interacting was collected using a sniffer tool in order to build a data matrix that contains the most important attributes during the connections HTTP (Hypertext Transfer Protocol) to the server either of the legitimate user or attacker. To achieve

this the author defined the following sequence steps: session identification, attribute extraction, feature construction, and feature matrix construction. This last step of the procedure the author decided to implement the same extraction features method used by Gholan [74] which is Principal Component Analysis (PCA) method which is a way of identifying inner feature patterns in high dimensional data and expressing the data in such a way to find the similarities and differences of them. In this work, the author obtained eleven attributes defined as follows in Table 3.1 then of applying the above steps. Then, the data collected were used for training and testing various machine learning algorithms model, such us: K-means, DBSCAN, and SVM for the purpose of validating the effectiveness of the proposed attributes and features. Unlike before authors, Xiong, does not use a benchmark data set he uses his own data set to build the detection intrusion system instead of a reference data set.

With the arrival of IPV6 (Internet Protocol version 6) technology New threats were discovered due to the exploitation of lacks in IPv6 architecture design offering appear news opportunities for attackers. The deployment of IPv6 in real networks indicates that a large amount of DDoS attacks can be executed in IPv6 by the same methods used in IPv4 (Internet Protocol version 4) such as flooding of ping request packets or by new methods based on new updates exist in IPv6 [72]. A number of these attacks are performed by exploiting the characteristics of protocols version six, such as TCP, UDP and ICMP. Zulkiflee [80] proposes a framework capable of select the most relevant subset of features of ipv6 packets and evaluate their importance

introducing machine learning techniques to mitigate TCP, UDP, and ICMPV6 flooding attacks. The data were collected using a test environment simulating an IPv6 network, where three types of attacks were used such as Alive6, FloodRouter, and Smurf6 to create the anomaly data. Then, the bio-inspired computing technology such as Particle Swarm Optimization (PSO) algorithm was used to apply the selection features process which is capable of identifying an optimized process and has a diversity feature that can identify unknown scenarios. In this step was achieved to obtain five relevant features that allow identifying the IPv6 network attacks. These features are TimeIntvl (duration), SrcIP (source IP), SrcPort (port source), DstPort (destiny port), and Protocol, and finally, a data mining called support vector machine (SVM) to evaluate the data set acquired. The framework proposed by Zulkiflee implement the same procedure of intrusion detection system (IDS) analyzed in this state of the art. However, it is oriented to mitigate attacks based on IPv6 architecture. Because it is a technology deployed to replace to IPv4 architecture and is necessary having into account that IDS developed for IPv4 networks does not work to IPv6 networks [25].

3.4 Machine Learning DDoS Detection for Consumer Internet of Things Devices

The Internet of Things (IoT) technology is one of the most recent research fields because the IoT devices connecting to the Internet have increased exponentially. Thus, these become in potential victims of attackers. Is important to mention that the

Table 3.1: Attributes of a Single Session [45].

Attribute	**Description**
un	the number of user requests without login
ft	the first user request time
lt	the last user request time
t	the time of user browsing web server
dq	duration between two continuous requests
c	the total number of user requests
a	the number of user requests invoking an action
m	the number of user requests invoking a method
s	the HTTP status codes of user requests
pa	the number of parameters in a request
pg	the number of user browsing a page

number of IoT devices is projected to grow 8 billion in 2017 to 20 billion in 2020 [46] taking in account the above affirmation is necessary to protect each IoT devices implementing a smart intruder detection system. Because it will be easy for the attackers to build a botnet based on them. For instance, in October 2016 the Mirai botnet was commanded by 100,000 IoT devices to conduct a DDoS attack on many popular websites such as Amazon, Netflix, Twitter, CNN, and PayPal. These important sites were broken for several hours. In this article, it is proposed to apply techniques based on machine learning algorithms to distinguish normal IoT packets from DoS attack packets, such as: K-nearest neighbors "KDTree" algorithm (KN), Support vector machine with linear kernel (LSVM), Decision Tree (DT) using Gini impurity scores, Random Forest using Gini impurity scores (RF) and Neural Network (NN). The main aim of this implementation is to evidence that packet-level machine learning DoS detection can accurately distinguish between normal and DoS attack

traffic from consumer IoT devices. The data-set used in this project was collected simulating a iot network consumer. This is composed by a router, some popular consumer IoT devices for benign traffic, and some adversarial devices performing DoS attacks. The final idea is to compare the variety performance of classifiers mentioned above using the custom data set.

Many of the works reviewed in the state of the art propose different alternatives to solve the problem of DDoS attacks. Among the most commonly used methods are machine learning techniques based on information classification which use intelligent algorithms to analyze behavior patterns within a data set. They also propose the implementation of pre-established data sets to train and test these algorithms such as DARPA, KDD99, and custom dataset. On the other hand, this research also makes use of the techniques mentioned above but it is also proposed to use two feature selection methods in order to reduce the dimension of features and obtain two different subsets in order to identify which features improve the algorithm's decision process. These two methods are $VarianceThreshold$ and $SelectkBest$ which are explained in the results section.

Chapter 4.
Design and Implementation

In this section, we are going to explain the methodology implemented for developing this project, describing step by step each phase as shown below. The main idea is to build an IDS that can receive a benchmark dataset that represents the behavior pattern of network traffic and then, using several machine learning algorithms trained, It can be able to classify the network traffic in two categories such as normal or anomalous, to finally validate each performance and select the best algorithm. To achieve that we will use an appropriate methodology that allows us to follows a series of steps to complete this process. On the other hand, it is important to mention that the development tool used in this work is the well-known python library named sklearn which is an open source machine learning library that allows using supervised and unsupervised learning. It also provides various tools for model fitting, data preprocessing and model selection and evaluation. One advantage more considerable is the short learning curve, the ability to run on multiple operating systems, and take advantage of the computational resources of the computer running it. Actually, there are several companies using this tool, such us: J.P.Morgan, Evernote, Spotify,

Inria, Télécom ParisTech, and among others [1].

4.1 Methodology

The methodology implemented in this work will be based on Knowledge Discovery in Databases (KDD) technique. This approach will allows analyze and process information large amounts of information to identify relevant patterns in data set selected [27]. The steps applied in this methodology are shown in the Figure 4.1 and will be developed below in this article .

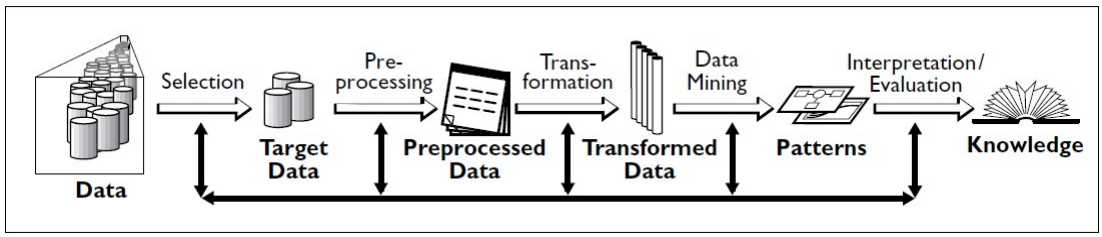

Figure 4.1: Graphic Schema of KDD Process [27].

4.2 Dataset

The data set we will use for this experiment is known as NSL-KDD [2]. This is an improved version of the KDD99 [3] data-set which was acquired from the 1998 DARPA in the intrusion detection evaluation program. In this program, They set up an environment to acquire raw TCP/IP connection data for a local-area network (LAN) simulating a typical U.S. Air Force LAN. They operated the LAN as if it was a true

[1] https://scikit-learn.org/stable/index.html
[2] https://www.unb.ca/cic/datasets/nsl.html
[3] http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

environment, but also, it was subjected to different attacks in order to obtain a data set composed of normal and abnormal activity. Thus, we can say that KDD99 is a subset of the 1998 DARPA dataset. Moreover, it was collected and distributed at the Massachusetts Institute of Technology (MIT) Lincoln Laboratory [57].

Many irregularities have been noted in KDD99 but these have been removed to create the new NSL-KDD data set. Among the irregularities we have: redundant data, empty registries, and duplicate records from the test set.

To model construction suggest has a data set ready and preprocessed. It dataset will help us to train and test our classification model. For this reason, NSL-KDD is a data set suggested to train our classification model [67].

NSL-KDD data set has been used in several network analyses by other researchers employing different classification approaches which can be either statistical techniques or machine learning and also software tools as WEKA with the principal objective to develop an effective intrusion detection system. Moreover, it contains five classes of network connection vectors and they are categorized as one normal class and four attack classes. The four attack classes fall into grouped as: [20].

1. DOS: denial of service

2. R2L: unauthorized access from a remote machine

3. U2R: unauthorized access to local super user (root)

4. Probing: surveillance and other probing

The whole NSL-KDD dataset contains around of 4.900.000 single connection records, each of which consists of 41 features (See Tables 4.1, 4.2, 4.3, and 4.4) labeled as normal or attacks.

The 41 features of each connection vector which were exposed in four tables having been classified into four categories:

- Basic features: these are obtained from the packet header, without examining the contents of the packet (duration, protocol type, service, flag and the number of bytes sent from the source to the destination and vice versa), as shows in Table 4.1.

- Content features: these are defined by analyzing the content of the TCP packet (number of unsuccessful attempts to login to the system), as shown in Table 4.2.

- Time features: these define the duration of the connection from a source IP address to target IP addresses. The connection is a sequence of data packets starting and ending at some predefined times as shown in Table 4.3.

- Host based traffic features: these are based on a window that has an interval of a given number of connections (not time intervals). This is suitable for describing attacks that last longer than the interval of the stipulated time features, as shown in Table 4.4.

Table 4.1: Basic Features of each Network Connection Vector

No.	Attribute Name	Description	Sample Data	Type
1	Duration	Length of time duration of the connection	0	Int
2	Protocol_type	Protocol used in the connection	Tcp	Categorical
3	Service	Destination network service used	ftp_data	Categorical
4	Flag	Status of the connection Normal or Error	SF	Categorical
5	Src_bytes	Number of data bytes transferred from source to destination in single connection	491	Int
6	Dst_bytes	Number of data bytes transferred from destination to source in single connection	0	Int
7	Land	if source and destination IP addresses an port numbers are equal then, this variable takes value 1 else 0	0	Int
8	ent	Total number of wrong fragments in this connection	0	Int
9	Urgent	Number of urgent packets in this connection. Urgent packets are packets with the urgent bit activated	0	Int

Table 4.2: Content Related Features of each Network Connection Vector

No.	Attribute Name	Description	Sample Data	Type
10	Hot	Number of hot indicators in the content such as: entering a system directory, creating programs and executing programs	0	Int
11	Num_failed _logins	Count of failed login attempts	0	Int
12	Logged_in	Login Status : 1 if successfully logged in; 0 otherwise	0	Bool
13	Num_comp romised	Number of "compromised' ' conditions	0	Bool
14	Root_shell	1 if root shell is obtained; 0 otherwise	0	Bool
15	Su_attempt ed	1 if "su root" command attempted or used; 0 otherwise	0	Bool
16	Num_root	Number of "root" accesses or number of operations performed as a root in the connection	0	Int
17	Num_file_c reations	Number of file creation operations in the connection	0	Int
18	Num_shells	Number of shell prompts	0	Int
19	Num_acces s_files	Number of operations on access control files	0	Int
20	Num_outbo und_cmds	Number of outbound commands in an ftp session	0	Int
21	Is_hot_logi n	1 if the login belongs to the "hot" list i.e., root or admin; else 0	0	Bool
22	Is_guest_lo gin	1 if the login is a "guest" login; 0 otherwise	0	Bool

Table 4.3: Content Related Features of each Network Connection Vector

No.	Attribute Name	Description	Sample Data	Type
23	Count	Number of connections to the same destination host as the current connection in the past two seconds	2	Int
hline 24	Srv_count	Number of connections to the same service (port number) as the current connection in the past two seconds	2	Int
25	Serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in count (23)	0	Int
26	Srv_serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in srv_count (24)	0	Int
27	Rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in count (23)	0	Int
28	Srv_rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in srv_count (24)	0	Int
29	Same_srv_rate	The percentage of connections that were to the same service, among the connections aggregated in count (23)	1	Int
30	Diff_srv_rate	The percentage of connections that were to different services, among the connections aggregated in count (23)	0	Int
31	Srv_diff_host_ rate	The percentage of connections that were to different destination machines among the connections aggregated in srv_count (24)	0	Int

Table 4.4: Content Related Features of each Network Connection Vector

No.	Attribute Name	Description	Sample Data	Type
32	Dst_host_count	Number of connections having the same destination host IP address	150	Int
33	Dst_host_srv_ count	Number of connections having the same port number	25	Int
34	Dst_host_same _srv_rate	The percentage of connections that were to the same service, among the connections aggregated in dst_host_count (32)	0.17	Int
35	Dst_host_diff_ srv_rate	The percentage of connections that were to different services, among the connections aggregated in dst_host_count (32)	0.33	Int
36	Dst_host_same _src_port_rate	The percentage of connections that were to the same source port, among the connections aggregated in dst_host_srv_c ount (33)	0.17	Int
37	Dst_host_srv_ diff_host_rate	The percentage of connections that were to different destination machines, among the connections aggregated in dst_host_srv_c	0	Int
38	Dst_host_serro r_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in dst_host_count (32)	0	Int
39	Dst_host_srv_s error_rate	The percent of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in dst_host_srv_c ount (33)	0	Int
40	Dst_host_rerro r_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_count (32)	0.05	Int
41	Dst_host_srv_r error_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_srv_c ount (33)	0	Int

The attributes of the NSL-KDD data-set are defined in the Table 4.5, which describes each one of the features of the data-set. We only focus in records of DDoS attacks. The experiment has two stages. First, preprocessing data. Second, evaluation and validation of model results.

Table 4.5: Features of NSL-KDD Dataset and their Types and Numbers [54]

Type	**Features with their numbers**
Categorical	protocol_type (1), service (2), flag (4)
Binary	land (7), logged_in (12), root_shell (14), su_attempted (15), is_host_login (21), is_guest_login(22)
Numeric	duration (1), src_bytes (5), dst_bytes (6), wrong_fragment (8), urgent (9), hot (10), num_failed_logins (11), num_compromised (13), num_root (16), num_file_creations (17), num_shells (18), num_access_files (19), num_outbound_cmds (20), count (23) srv_count (24), serror_rate (25), srv_serror_rate (26), rerror_rate (27), srv_rerror_rate (28), same_srv_rate (29) diff_srv_rate (30), srv_diff_host_rate (31), dst_host_count (32), dst_host_srv_count (33), dst_host_same_srv_rate (34), dst_host_diff_srv_rate (35), dst_host_same_src_port_rate (36), dst_host_srv_diff_host_rate (37), dst_host_serror_rate (38), dst_host_srv_serror_rate (39), dst_host_rerror_rate (40), dst_host_srv_ rerror_rate (41)

4.2.1 Preprocessing Data

Selecting the necessary information is important for our model to achieve high performance. Perform basic operations such as: removing redundant or duplicate records, analyzing missing data are of great importance.

4.2.2 Data Transformation

Basically NSL-KDD, is a data set that contains 22.544 records and 43 attributes. Before starting to train and test our model is necessary identify attributes with categorical variables which are showed in the Table 4.6. Therefore, these must be encoded to binary values [7]. The transformation of categorical variables will be perform using a get_dummies module of python. It will identify the categorical values in the attributes and creates a binary column for each category. Once encoded is done the number of columns in data set will increasing from 43 to 115 columns.

Table 4.6: Categorical Attributes

No.	Attribute Name	Description	Sample Data
0	protocol_type	Protocol used in the connection	TCP
1	service	Destination network service used	http
2	flag	Status of the connection –Normal or Error	SF

The attribute "Type" which contain several types of DoS attack as show in Table 4.7 is our target variable. It will be transform to using binary values so that now, the values for normal traffic will be identified with zero (0) and attacks traffic with one (1). This will allow the model to identify with integer values the different types of traffic. Thus we have two only two classes in 'type' attribute.

4.2.3 Data Set Normalization

Data Set Normalization is a common requirement for many machine learning estimators. Because, It is extremely important to improve the performance of the system. In this step, we achieve all attributes are formatted to the same scale. To achieve

Table 4.7: Type of DoS Attacks in NSL-KDD Dataset

No.	Type of Attack
1	neptune
2	back
3	land
4	smurf
5	pod
6	teardrop

this we use The scikit-learn preprocessing package provides the min-Max method of standardization [7]. It is a package provided by scikit-learn. Preprocessing that scale the features to lie between a given minimum and maximum value. Generally, between zero and one. The motivation is due to robustness to very small standard deviations of features and preserving zero entries in sparse data [64].

4.3 Model

For the development of the experiment, the algorithm known as Support Vector Machine (SVM) will be implemented which is a popular algorithm used for creating Instruction Detection System. This method was proposed in 1998 by Vladimir Vapnik and his principal objective is find the better hyper-plane that be able to separate the data set into different class, taking into account that the better hyper-plane found is the hyper-plane that allows it to trace the distance or margin maximum between hyper-plane and data point. To accomplish it, the Support Vector Machine (SVM) takes the training data obtained from the basic input space into a higher dimensional feature space using kernels and then gets the most favourable isolating hyper-plane

or a decision boundary in the form of support vectors. SVM allows to configure a parameter called penalty factor, Witch allows users to make a trade-off between the number of miss-classified samples and the width of a decision boundary [31].

The Support Vector Machine (SVM) algorithm need to takes a percent of data in order to train the system. Then, it will find several support vectors that represent the training data, and it can used to conform the SVM model [7]. The model will use the following kernels: Gaussian Kernel (Radial Basis Function), Polynomial, Lineal.

Given by a training set of N data points $\{Y_k, X_x\}$ where Y_k is the kth input pattern and $X_k \in R^n$ is the kth output pattern, the support vector method approach focus at building a classifier of the form:

$$y(x) = sing\left[\sum_{k=1}^{N} \alpha_k y_k \psi\left(X, X_k\right) + b\right] \tag{4.1}$$

Where α_k are positive real constants, $\psi\left(x, x_k\right)$ represent the kernel, α_k are their Lagrange multipliers, and $\boldsymbol{b}$ is a real constant. For $\psi\left(.,.\right)$ one typically has the following choices: $\psi\left(x, x_k\right) = x_k^t x$ (linear SVM); $\psi\left(x, x_k\right) = \left(x_k^t x + 1\right)^d$ (polynomial SVM of degree d); $\psi\left(x, x_k\right) = \exp\left\{- \parallel x - x_k \parallel_2^2 / \sigma^2\right\}$ (RBF SVM).

4.4 Features Selection

To carry out this phase, two methods of features selection will be used, as they are: VarianceTherdsold and SelectkBest. These are provided by the python scikit-learn library and explained below.

The method of features selection presented in this article to reduce the features

dimension is filter methods which is based on use of variable ranking techniques as first criteria. A good selection of ranking and threshold are used to evaluate the correlation between variables and remove attributes below the threshold defined. This technique must be applied before perform the test of classification model. So that, it can eliminate the less relevant variables of data set. It mean, the features that have not influence on the labeled classes become discarded automatically [13].

The ranking method used in the prepossessing phase is the Pearson correlation coefficient criterion, defined as [30]:

$$R(i) = \frac{cov(Xi, Y)}{\sqrt{var(Xi) * var(Y)}} \tag{4.2}$$

Where (cov) determinates the covariance and (var) the variance.

Covariance criterion can be used to detect the correlation between variables and can be extended to the case of two-class classification [30].The next section will shows the correlation matrix which summarizes the variable dependency in a much more succinct way. It shows the direct correlation between variable pairs, using a range between 1 and -1.

Once, the appropriate features have been selected we will use the kernels indicated above to train and test the features acquired. The percentage of samples to train and test was divided into 67% to training and 33% to testing. In addition, the each parameter of the regularization of kernels was configurated with the default value.

Chapter 5.
Results

The result of this work will be based on the application of the features selection methods mentioned above which use unique techniques to apply the features selection. Once, the attributes are chosen, these will be used to train and test the Support Vector Machine kernel. Moreover, the results obtained will be compared to finally select a model that allows classifying the traffic network in an efficient way. On the other hands, in this phase, we will describe each process performed using classification metrics and illustrations that show the behavior of models.

Methods used for selecting features

To carry out this phase, we use $VarianceThreshold$ and $SelectkBest$ feature selection methods, which are functions provided by Scikit-learn library[1].

The first method exposed here is applied using the VarianceTherdsold function provided by scikit-learn python library. This function makes it easier to evaluate the p-value parameter to select the most relevant features in all data set. This method consider that Boolean features are Bernoulli random variables, and the variance of

[1]Scikit-learn Machine Learning in Python, https://scikit-learn.org

such variables is defined by:

$$VAR(X) = p * (1 - p) \tag{5.1}$$

As the purpose of select the best features for our model, we iterated the parameter 'p' between a value of 0.6 and 0.9 to get the Table 5.1. This table have four important metrics: accuracy, precision, recall and finally cross validation score that help us to validate the system. In this part of the process, we will be able to identify the p-value appropriate for selecting the best attributes.

To generate the p-value a polynomial kernel will be used. Moreover, it will be evaluated using the classification metrics mentioned above. The box graph represented in figure 5.1 we'll help to display in a graphic way the performance of the model.

Once,the p-value is generated a cross validation shall be performed using several SVM kernels among them: Linear, Polynomial and Gaussian. These will be trained and evaluated using classification metrics. Then, in order to we draw a box graph represented in Figure 5.1 in order to visualize in a graphic way the performance of each model.

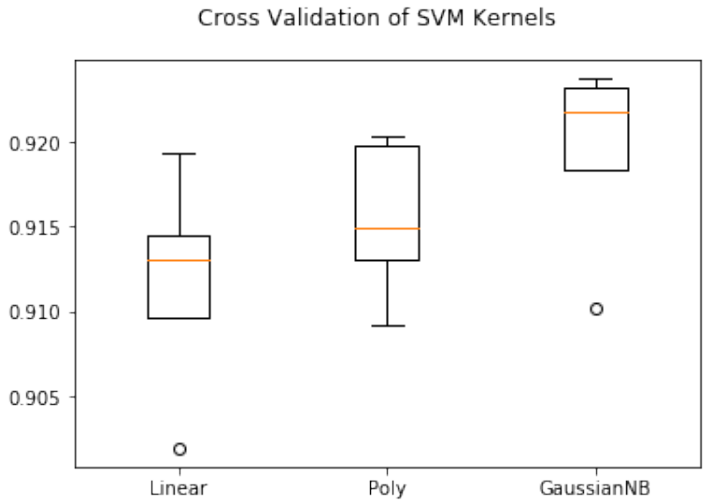

Figure 5.1: VarianceTherdsold function: Box plot of Cross Validation Score of each Model

The achieved result show us that when the Variance method iterated with p equal to 0.9, the model suffered a overfitting, because the score obtained is 0.980777 which is a perfect score. Thus, the model could predict a misclassification for the new entries. Thus, we selected p equal to 0.8, because measurements obtained with each of the metrics shows a better behavior of the model. The attributes obtained iterating with p-value equal to 0.8 are showed in the Table 5.2.

Table 5.1: Results of Changing "p" Value Parameter

p **VALUE**				
0.6	0.7	0.8	0.9	Metrics
0.820000	0.823922	0.921961	0.982745	Accuracy
0.687623	0.694466	0.954043	0.978087	Precision
0.934330	0.939873	0.837162	0.974960	Recall
0.825926	0.823995	0.926876	0.980777	Cross Validation

Table 5.2: Table of Features Selected Using VarianceTherdsold Function [20]

No	Attribute Name	Description
1	Logged_in	Login Status : 1 if successfully logged in; 0 otherwise
2	Serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in count (23)
3	Srv_serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in srv_count (24)
4	Same_srv_rate	The percentage of connections that were to the same service, among the connections aggregated in count (23)
5	Dst_host_srv_count	Number of connections having the same port number
6	Dst_host_same_srv_rate	The percentage of connections that were to the same service, among the connections aggregated in dst_host_count (32)
7	Dst_host_rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_count (32)
8	Dst_host_srv_rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_srv_c ount (33)
9	enco_host_name	Destination network service used
10	enco_printer	Destination network service used
11	enco_S3	Destination network service used

The correlation matrix represented in Figure 5.2 define a range of color to represent the the relevant values of attributes. The light blue color means that the variables are not correlated. However, the closer you get to the yellow color the greater the correlation that exists between the variables.

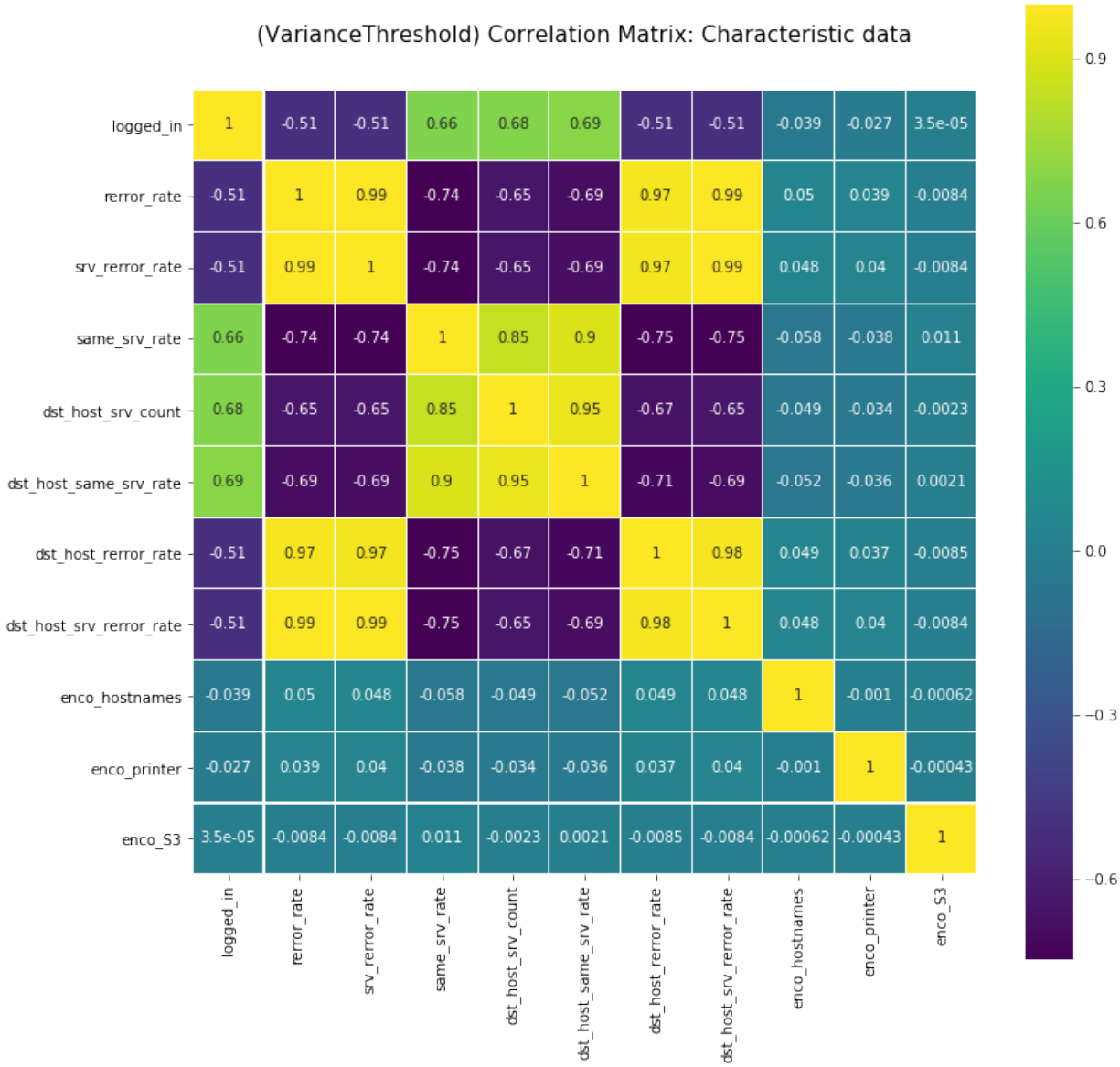

Figure 5.2: Correlation Matrix of Selected Features

The second method implemented is the "SelectkBest" function provided by scikit-learn python library which uses chi-squared stats to compute the stats between each non-negative feature and class and retrieve only the two best features. Thus, this score will be used to choose the k features with the highest values for the test chi-squared statistic from data set [63].

The attributes obtained with this function are showed in Table 5.3. The only different between VarianceTherdsold and SelectkBest method is the latter generates two attributes that there are not included in VarianceTherdsold which are: enco_RSTR and enco_whois. Once, the features have been choosed we will test and train the model.

Table 5.3: Table of Features Selected Using SelectkBest Function [20].

No	Attribute Name	Description
1	Logged_in	Login Status : 1 if successfully logged in; 0 otherwise
2	Serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in count (23)
3	Srv_serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in srv_count (24)
4	Same_srv_rate	The percentage of connections that were to the same service, among the connections aggregated in count (23)
5	enco_whois	Destination network service used
6	enco_RSTR	Destination network service used
7	Dst_host_rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_count (32)
8	Dst_host_srv_rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_srv_c ount (33)
9	enco_host_name	Destination network service used
10	enco_printer	Destination network service used
11	enco_S3	Destination network service used

The Table 5.4 shows the results of metrics (occuracy, presicion, recall, cross validation) score obtained after evaluate the model. It experiments indicates that the moment when the model works better is when it is evaluated with the polynomial kernel a unlike the VarianceTherdsold method mentioned above. To visualized the results of graphic form a box plot is drawn as defined on Figure 5.3.

Table 5.4: Metrics Models Performance Using VarianceThredsold Function

Models				
No.	Linear	Polynomial	GaussianNB	Metrics
0	0.912157	0.916078	0.912157	Accuracy
1	0.952866	0.956329	0.952866	Precision
2	0.800000	0.808021	0.800000	Recall

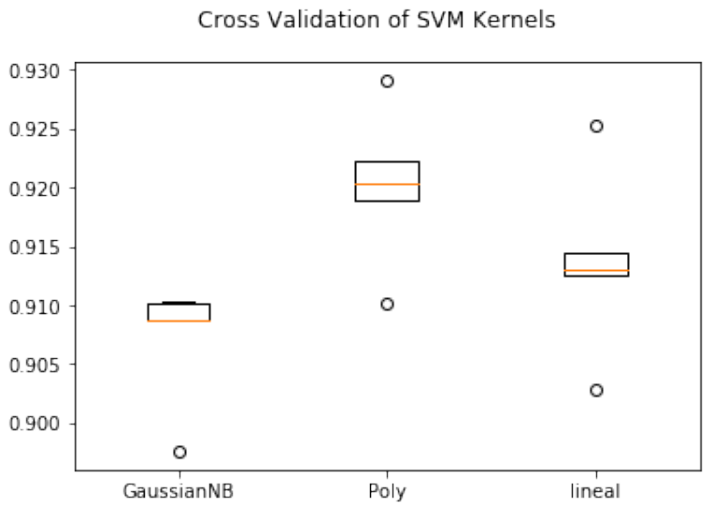

Figure 5.3: SelectkBest Function: Box Plot of Cross Validation Score of each Model

Table 5.5: Metrics Models Performance Using SelectKbest Function

Models				
No.	Linear	Polynomial	GaussianNB	Metrics
0	0.916667	0.921961	0.923725	Accuracy
1	0.949294	0.951297	0.962629	Precision
2	0.810411	0.824110	0.818630	Recall

Chapter 6.
Conclusions and Future Work

According to the state of art performed in this work. It was possible to identify different approaches to build an intelligent model to be able to classify data having into account behavior patterns. Among these technics, we could mention Neuronal Network (ANNS), Bayesian Networks, Support Vector Machine (SVM), Classification Tree, and Markov chains and Hidden Markov Models (HMMs). Because the principal idea is to build an intrusion detection system that can be able to differentiate among normal and anomalous network traffic. Thus, in this investigation, we selected the SVM algorithm to realize that work. SVM is a versatile technic since it provides different configurations called kernels in order to train and test several models. Each model offers a unique performance that we can measure using different classifications metrics such as cross-validations.

Search a data set to train and test the model chosen above is a necessary and very important task in the developing phase of this IDS. As we have mentioned in this investigation the data set selected was NSL-KDD. Because they contain valuable information about the behavior DDoS attack which helps us to understand how this attack is launched. Apart from the NSL-KDD data set, there are other data sets

named in the literature of DDoS attacks such as DARPA and KDD99. However, NSL-KDD presents advantages in front of these as for example the fact of it does not include redundant records.

In Order to achieve the result proposed was necessary to realize a search to finds a better methodology capable of shows a series of steps that help us to preprocess the data set found with the purpose to mix these with the machine learning algorithm selected and so accomplish to acquire a result very close to the point. In this case, the methodology implemented is well known as knowledge discovery in Databases (KDD).

During the performance of the IDS was possible to train and test the model and also, measure their quality using classification metrics such as accuracy, precision, recall, and cross-validations as was indicated in the results section. In total there were 3 types of SVM kernels used to evaluate the decision process such as Linear, Polynomial, and GaussianNB. Nevertheless, to select the appropriate model we made use of two metrics, such as accuracy and recall. Thus, we decided to take the polynomial model which implemented the features represented in the Table 5.2 and also, has an accuracy of 0.916078 and a recall of 0.808021. Note that model results may vary depending on the feature selection method used.

Furthermore, to have carried out all this process permitted us to participate in different academic investigation scenery for instance: be a speaker in the 1st International Congress on Advances in New Trends and Technologies in Quito, Ecuador. In this

country, we were able to share the advance of our investigation work exposing our article called "SIDS-DDoS, A Smart Intrusion Detection System for Distributed Denial of Service Attacks" published in the book called "Advances in Emerging Trends and Technologies" through Springer database [4] and at the same time, we were able to represent Universidad Tecnológica de Bolivar. Additionally, we had the opportunity to participate in the IEEE Colombian Conference on Applications of Computational Intelligence ColCACI 2019 performed in Barranquilla, Colombia where we shared new advance in our investigation such us: the article called "Evaluating Features Selection on NSL-KDD data set to Train a Support Vector Machine Based Instruction Detection System" [3] which was published by IEEE. In both events, we achieved to publish the articles mentioned above in scientific journals.

As future work derived from this thesis, there are several lines still to develop for instance: Build a custom data set that contain the network traffic acquired from custom real network architecture and so, to be able to use these data for applying the same methodology implemented in data set preprocessing selected in this work. On the other hand, build an IDS capable to work as a firewall in a real environment, and last, explore the DDoS attack launched through at levels of IoT devices.

Finally, thanks to each one of the specific objectives developed in this work and the correct use of methodology proposed, we achieved to be able to construct an IDS prototype called "Smart Intelligent Network Traffic Classifier System (SINTCS)" that meets the requirements presented in this investigation. It was published in the Na-

tional Direction of Copyright of the Colombian State which was developed in python language. Because the start of the development of this work began in 2018 the selected data set at that moment was NSL-KDD which is data set published in 2009, so it is recommended to make use of a more recent data set to test the prototype proposed, such as the case of CICDDoS2019 data set published in 2019 by the University of New Brunswick on its website. CICDDoS2019 contains benign and the most up-to-date common DDoS attacks, which resembles the true real-world data (PCAPs). It also includes the results of the network traffic analysis using CICFlowMeter-V3 with labeled flows based on the time stamp, source, and destination IPs, source and destination ports, protocols and attack (CSV files) [1].

[1] https://www.unb.ca/cic/datasets/ddos-2019.html

Bibliography

[1] Computer networks: The international journal of computer and telecommunications networking.

[2] S. K. Ajagekar and V. Jadhav. Study on web DDOS attacks detection using multinomial classifier. In *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, pages 1–5, Chennai, India, Dec. 2016. IEEE.

[3] L. Almeida and J. C. Martinez Santos. Evaluating features selection on nsl-kdd data-set to train a support vector machine-based intrusion detection system. pages 1–5, 06 2019.

[4] L. Á. Almeida and J. C. Martinez-Santos. Sids-ddos, a smart intrusion detection system for distributed denial of service attacks. In *The International Conference on Advances in Emerging Trends and Technologies*, pages 380–389. Springer, 2019.

[5] D. Barbara, N. Wu, and S. Jajodia. Detecting novel network intrusions using bayes estimators. In *Proceedings of the 2001 SIAM International Conference on Data Mining*, pages 1–17. SIAM, 2001.

[6] L. E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology. *Bulletin of the American Mathematical Society*, 73(3):360–363, 1967.

[7] Y. B. Bhavsar and K. C. Waghmare. Intrusion detection system using data mining technique: Support vector machine. *International Journal of Emerging Technology and Advanced Engineering*, 3(3):581–586, 2013.

[8] M. H. Bhuyan, H. J. Kashyap, D. K. Bhattacharyya, and J. K. Kalita. Detecting distributed denial of service attacks: methods, tools and future directions. *The Computer Journal*, 57(4):537–556, 2013.

[9] A. L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2):245–271, 1997.

[10] J. Cannady, J. Harrell, et al. A comparative analysis of current intrusion detection technologies. In *Proceedings of the Fourth Technology for Information Security Conference*, volume 96, 1996.

[11] S. Cateni, M. Vannucci, M. Vannocci, and V. Colla. Variable selection and feature extraction through artificial intelligence techniques. *Multivariate Analysis in Management, Engineering and the Science*, pages 103–118, 2012.

[12] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15, 2009.

[13] G. Chandrashekar and F. Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[14] K. J. Cios, W. Pedrycz, and R. W. Swiniarski. Data mining and knowledge discovery. In *Data mining methods for knowledge discovery*, pages 1–26. Springer, 1998.

[15] P. J. Criscuolo. Distributed Denial of Service: Trin00, Tribe Flood Network, Tribe Flood Network 2000, and Stacheldraht CIAC-2319. page 18.

[16] H. Debar, M. Becker, and D. Siboni. A neural network component for an intrusion detection system. In *IEEE symposium on security and privacy*, pages 240–250, 1992.

[17] D. E. Denning. An intrusion detection model. ieee transactions on software engineering, se-13. 1987.

[18] B. Deokar and A. Hazarnis. Intrusion detection system using log files and reinforcement learning. *International Journal of Computer Applications*, 45(19):28–35, 2012.

[19] R. V. Deshmukh and K. K. Devadkar. Understanding DDoS Attack & its Effect in Cloud Environment. *Procedia Computer Science*, 49:202–210, 2015.

[20] L. Dhanabal and S. Shantharajah. A study on nsl-kdd dataset for intrusion detection system based on classification algorithms. *International Journal of Advanced Research in Computer and Communication Engineering*, 4(6):446–452, 2015.

[21] D. Dittrich. An irc tutorial, 1999.

[22] C. Douligeris and A. Mitrokotsa. Ddos attacks and defense mechanisms: classification and state-of-the-art. *Computer Networks*, 44(5):643–666, 2004.

[23] S. Duque and M. N. b. Omar. Using Data Mining Algorithms for Developing a Model for Intrusion Detection System (IDS). *Procedia Computer Science*, 61:46–51, 2015.

[24] M. Ektefa, S. Memar, F. Sidi, and L. S. Affendey. Intrusion detection using data mining techniques. In *Information Retrieval & Knowledge Management,(CAMP), 2010 International Conference on*, pages 200–203. IEEE, 2010.

[25] O. E. Elejla, B. Belaton, M. Anbar, and A. Alnajjar. Intrusion detection systems of icmpv6-based ddos attacks. *Neural Computing and Applications*, 30(1):45–56, 2018.

[26] K. Fakieh. Survey on ddos attacks prevention and detection in cloud. 12 2016.

[27] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. The kdd process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39(11):27–34, 1996.

[28] F. Gullo. From patterns in data to knowledge discovery: what data mining can do. *Physics Procedia*, 62:18–22, 2015.

[29] B. Gupta, M. Misra, and R. C. Joshi. An isp level solution to combat ddos attacks using combined statistical based approach. *arXiv preprint arXiv:1203.2400*, 2012.

[30] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182, 2003.

[31] M. Gyanchandani, J. Rana, and R. Yadav. Taxonomy of anomaly based intrusion detection system: a review. *International Journal of Scientific and Research Publications*, 2(12):1–13, 2012.

[32] D. Heckerman. A tutorial on learning with bayesian networks. In *Learning in graphical models*, pages 301–354. Springer, 1998.

[33] Z. M. Hira and D. F. Gillies. A review of feature selection and feature extraction methods applied on microarray data. *Advances in bioinformatics*, 2015, 2015.

[34] F. V. Jensen and T. Nielsen. Bayesian networks and decision graphs springer-verlag. *New york*, 2001.

[35] T. Joachims. Svmlight is an implementation of support vector machines (svms) in c, 2000.

[36] V. Jyothsna, V. R. Prasad, and K. M. Prasad. A review of anomaly based intrusion detection systems. *International Journal of Computer Applications*, 28(7):26–35, 2011.

[37] D. Karig and R. Lee. Remote denial of service attacks and countermeasures. *Princeton University Department of Electrical Engineering Technical Report CE-L2001-002*, 17, 2001.

[38] P. Kaur, M. Kumar, and A. Bhandari. A review of detection approaches for distributed denial of service attacks. *Systems Science & Control Engineering*, 5(1):301–320, Jan. 2017.

[39] S. Khalid, T. Khalil, and S. Nasreen. A survey of feature selection and feature extraction techniques in machine learning. In *2014 Science and Information Conference*, pages 372–378. IEEE, 2014.

[40] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur. Bayesian event classification for intrusion detection. In *null*, page 14. IEEE, 2003.

[41] C. Kruegel and T. Toth. Using decision trees to improve signature-based intrusion detection. In *International Workshop on Recent Advances in Intrusion Detection*, pages 173–191. Springer, 2003.

[42] L. Ladha and T. Deepa. Feature selection methods and algorithms. *International journal on computer science and engineering*, 3(5):1787–1797, 2011.

[43] J. Li, Y. Liu, and L. Gu. Ddos attack detection based on neural network. In *Aware Computing (ISAC), 2010 2nd International Symposium on*, pages 196–199. IEEE, 2010.

[44] Y. Liu, B. Cukic, and S. Gururajan. Validating neural network-based online adaptive systems: A case study. *Software Quality Journal*, 15(3):309–326, 2007.

[45] X. Luo, X. Di, X. Liu, H. Qi, J. Li, L. Cong, and H. Yang. Anomaly detection for application layer user browsing behavior based on attributes and features. In *Journal of Physics: Conference Series*, volume 1069, page 012072. IOP Publishing, 2018.

[46] J. Manyika, M. Chui, P. Bisson, J. Woetzel, R. Dobbs, J. Bughin, and D. Aharon. Unlocking the potential of the internet of things. *McKinsey Global Institute*, 2015.

[47] M. Markou and S. Singh. Novelty detection: a review—part 2:: neural network based approaches. *Signal processing*, 83(12):2499–2521, 2003.

[48] A. A. Markov. Extension of the limit theorems of probability theory to a sum of variables connected in a chain. *Dynamic probabilistic systems*, 1:552–577, 1971.

[49] J. Mirkovic and P. Reiher. A taxonomy of ddos attack and ddos defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2):39–53, 2004.

[50] A. Mishra, B. Gupta, and R. C. Joshi. A comparative study of distributed denial of service attacks, intrusion tolerance and mitigation techniques. In *2011 European Intelligence and Security Informatics Conference*, pages 286–289. IEEE, 2011.

[51] S. M. Mousavi. *Early detection of DDoS attacks in software defined networks controller*. PhD thesis, Carleton University, 2014.

[52] S. Mukkamala, G. Janoski, and A. Sung. Intrusion detection using neural networks and support vector machines. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290)*, volume 2, pages 1702–1707. IEEE, 2002.

[53] D. B. Parker. Demonstrating the elements of information security with loss scenarios. *Information Systems Security*, 3(1):17–22, 1994.

[54] M. R. Parsaei, S. M. Rostami, and R. Javidan. A hybrid data mining approach for intrusion detection on imbalanced nsl-kdd dataset. *International Journal of Advanced Computer Science and Applications*, 7(6):20–25, 2016.

[55] A. Patcha and J.-M. Park. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer networks*, 51(12):3448–3470, 2007.

[56] K. M. Prasad, A. R. M. Reddy, and K. V. Rao. Dos and ddos attacks: defense, detection and traceback mechanisms-a survey. *Global Journal of Computer Science and Technology*, 2014.

[57] D. D. Protić. Review of kdd cup'99, nsl-kdd and kyoto 2006+ datasets. *Vojnotehnički glasnik*, 66(3):580–596, 2018.

[58] A. Qayyum, M. Islam, and M. Jamil. Taxonomy of statistical based anomaly detection techniques for intrusion detection. In *Emerging Technologies, 2005. Proceedings of the IEEE Symposium on*, pages 270–276. IEEE, 2005.

[59] J. R. Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[60] G. Ramadhan, Y. Kurniawan, and C.-S. Kim. Design of TCP SYN Flood DDoS Attack Detection Using Artificial Immune Systems. page 5.

[61] P. Ristoski and H. Paulheim. Semantic web in data mining and knowledge discovery: A comprehensive survey. *Web semantics: science, services and agents on the World Wide Web*, 36:1–22, 2016.

[62] S. Saad, I. Traore, A. Ghorbani, B. Sayed, D. Zhao, W. Lu, J. Felix, and P. Hakimian. Detecting p2p botnets through network behavior analysis and machine learning. In *Privacy, Security and Trust (PST), 2011 Ninth Annual International Conference on*, pages 174–180. IEEE, 2011.

[63] scikit-learn developers. Feature Selection, 1999.

[64] scikit-learn developers. Preprocessing data, 1999.

[65] L. D. Stein and J. N. Stewart. Securing against denial of service attacks, 2003.

[66] G. Stumme, R. Wille, and U. Wille. Conceptual knowledge discovery in databases using formal concept analysis methods. In *European Symposium on Principles of Data Mining and Knowledge Discovery*, pages 450–458. Springer, 1998.

[67] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani. A detailed analysis of the kdd cup 99 data set. In *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pages 1–6. IEEE, 2009.

[68] R. van Loon. The tribe flood network distributed denial of service attack tool., 1997.

[69] W. H. Woodall and M. M. Ncube. Multivariate cusum quality-control procedures. *Technometrics*, 27(3):285–292, 1985.

[70] S.-Y. Wu and E. Yen. Data mining-based intrusion detectors. *Expert Systems with Applications*, 36(3):5605–5612, 2009.

[71] Y. Xie and S.-Z. Yu. Monitoring the application-layer ddos attacks for popular websites. *IEEE/ACM Transactions on Networking (TON)*, 17(1):15–25, 2009.

[72] X. Yang, T. Ma, and Y. Shi. Typical dos/ddos threats under ipv6. In *2007 International Multi-Conference on Computing in the Global Information Technology (ICCGI'07)*, pages 55–55, March 2007.

[73] N. Ye, Y. Zhang, and C. M. Borror. Robustness of the markov-chain model for cyber-attack detection. *IEEE Transactions on Reliability*, 53(1):116–123, 2004.

[74] G. R. Zargar and T. Baghaie. Category-based intrusion detection using pca. *Journal of Information Security*, 3(04):259, 2012.

[75] S. T. Zargar, J. Joshi, and D. Tipper. A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks. *IEEE Communications Surveys & Tutorials*, 15(4):2046–2069, 2013.

[76] C. Zhang, Z. Cai, W. Chen, X. Luo, and J. Yin. Flow level detection and filtering of low-rate ddos. *Computer Networks*, 56(15):3417–3431, 2012.

[77] D. Zhao, I. Traore, B. Sayed, W. Lu, S. Saad, A. Ghorbani, and D. Garant. Botnet detection based on traffic behavior analysis and flow intervals. *Computers & Security*, 39:2–16, Nov. 2013.

[78] Y. Zheng, B. Vanderbeek, E. Daniel, D. Stambolian, M. Maguire, D. Brainard, and J. Gee. An automated drusen detection system for classifying age-related macular degeneration with color fundus photographs. In *2013 IEEE 10th International Symposium on Biomedical Imaging*, pages 1448–1451. IEEE, 2013.

[79] C. V. Zhou, C. Leckie, and S. Karunasekera. A survey of coordinated attacks and collaborative intrusion detection. *Computers & Security*, 29(1):124–140, 2010.

[80] M. Zulkiflee, M. Azmi, S. Ahmad, S. Sahib, and M. Ghani. A framework of features selection for ipv6 network attacks detection. *WSEAS Trans Commun*, 14(46):399–408, 2015.