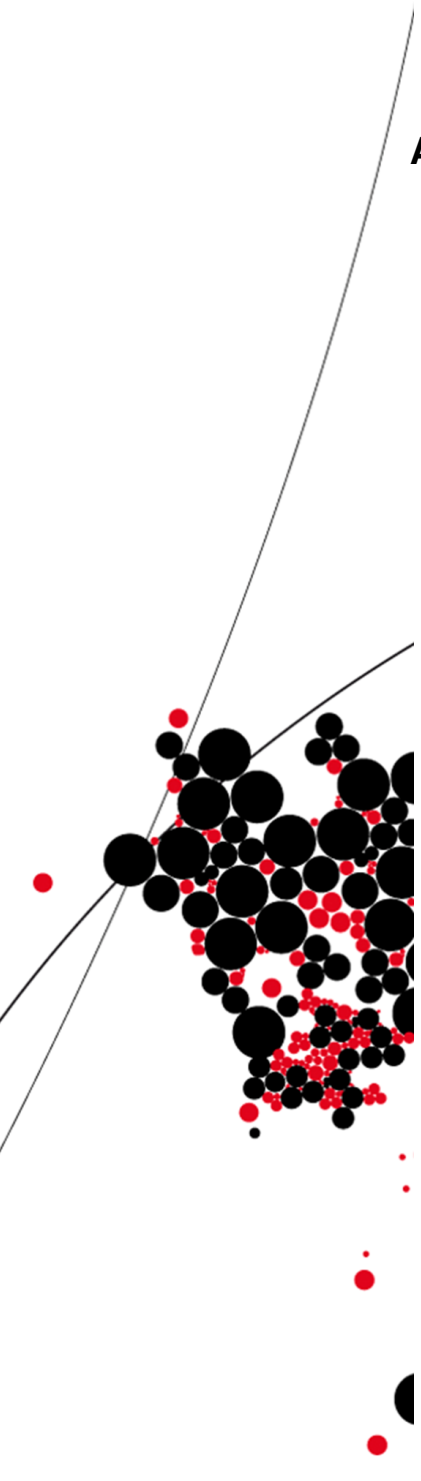
UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering,
Mathematics & Computer Science



Automatic Generation of Access Control List on Mellanox Switch For DDoS Attack Mitigation Using DDoS Fingerprints

Sridhar Bangalore Venugopal

M.Sc. Thesis

November 2019

Examination Committee:
Dr. Jose Jair C. Santanna
Dr. Aiko Pras
Dr. Andreas Peter

Design and Analysis of Communication Systems Group
Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Acknowledgment

I take this opportunity to express my sincere thanks to everyone who supported me during my thesis and health situation. First and foremost, my sincere thanks to my supervisor, Dr. Jose Jair C. Santanna, for his patience, motivation, guidance, providing constructive feedback and contributing in all aspects. During my thesis, I had hard time with my health condition and I was badly affected. My supervisor understood the critical situations and provided great support. I had to extend my thesis due to my health condition, but then time to time I had enough conversation with my supervisor. This thesis would not have been possible without his unconditional support.

I would like to express my gratitude to Gerald Schaapman, Pim Van Stam and Evert Jan Van for giving an opportunity to do thesis at NBIP. I had given access to a live infrastructure and they supported me in all aspects to finish the thesis successfully. I learnt many new things during this journey and explored things which I never did before.

I am very thankful for EIT digital members; Dr. Andreas Peter, Jan Schut and Monique Romarck for their valuable support during my course. My special thanks to, Dr. Jose Jair C. Santanna, Dr. Aiko Pras and Dr. Andreas Peter for accepting to be my committee members.

Finally, I must express my profound gratitude to my parents and my friends; Suraj Sonwalkar, Ramesh Krishnamurthy, Yuping Yan, Anna Prudnikova, Valentine Legoy, Giovanni Riva, Dron Lamichhane and Asif Khan for their support and encouragement.

Abstract

A Distributed Denial of Service (DDoS) is an attack that send a large amount of network traffic intending to disrupt online services. A successful DDoS attacks can cause significant impact in terms of financial damage and brand reputation. In 2018, Arbor security reported that forty percent of medium sized organizations protected by them were under frequent DDoS attacks. There are two main techniques to detect and mitigate DDoS attacks: signature-based detection and anomaly-based. The former is more specific and efficient in detecting *known* attacks, while the latter is more generic and capable of detecting new attacks. There are also solutions that combines these two techniques called hybrid-based. The problem is that, in the literature, there is no knowledge transfer from anomaly-based to signature-based solution addresed in this thesis. In other words, attacks detected by the anomaly-based solutions are not used for improving the signature-based (which is known to be faster). This type of improvement is suitable for attacks that happens frequently, for example, attacks performed by a botnet campaign. Our methodology relies on (after an attack is detected by the anomaly-based solution): (1) we collect *enough* samples of attack data, (2) summarize this attack data (called DDoS attack fingerprint), and (3) convert this attack summary into a signature-based solution. We used more than 200 actual attack traces to discover the minimum amount of data that contains *enough* attack information. Then, we propose an algorithm to automatically convert these attack information into Access Control List (ACL) on Mellanox switch (in a production network). Our results shows that the attack mitigation was successful through ACL's, but addition of legitimate IP addresses needs to be minimized. Also, few attacks the source IP addresses were not reduced, because they were widely distributed and for attacks with greater amount of source IP addresses the reduction was bigger. This research was performed at Nationale beheersorganisatie internet providers (NBIP) and some of our choice are in-line with NBIP.

Contents

Chapter 1

Introduction

A Distributed Denial of Service (DDoS) is a type of attack where attackers intend to prevent legitimate users from accessing the machine or network. In a DDoS attack, the incoming traffic from different infected source IP address overloads the target machine and its difficult to distinguish legitimate user traffic from malicious traffic due to traffic appear to come from a trusted source. DDoS is one of the most highlighted and dangerous attack in the Internet world due to traffic volume generated from multiple sources lasts for several hours [1].

DDoS attacks are gradually increasing. In 2018 an attacks size hit 1.7 Tbps which entered into terabit era of attacks [2]. Also, in 2018 GitHub was targeted with a DDoS attack which had a peak at 1.3 Tbps. In the same year, NetScout Arbor confirms 1.7 Tbps amplification attack on one of the customers of U.S. based service provider.

There are two important techniques found in the literature for detecting DDoS attacks, they are signature-based detection and anomaly-based detection [3]. Signature-based detection use signatures of already known attacks which are stored in a database to detect attacks [3]. This technique is efficient in detecting known DDoS attacks and less effective for new attacks. Anomaly-based detection is based on identifying the events which appears to be irregular with respect to normal system performance [3]. This technique is capable of handling new attacks that appears in the network. However, selection of threshold value to distinguish between normal traffic and malicious traffic is a critical task.

There is another technique called hybrid-based detection which works in combination of signature-based detection and anomaly-based detection [4]. This hybrid-based technique can be used to detect DDoS attack and improve the overall detection accuracy. Since both detection's works simultaneously, hybrid-based detection efficiently detects both known and unknown DDoS attacks.

1.1 Research Questions and Overall Methodology

In the state of art, there is no knowledge transfer from anomaly-based detection to signature-based detection. As anomaly-based detection have ability to detect new attacks, this information is not passed to signature-based detection which can mitigate the attack at a faster rate. By adding this "knowledge transfer" the system will become more efficient. In this thesis, we intend to connect anomaly-based with signature-based in order to improve the efficiency and minimize time of detecting the attacks. We are not implementing a new signature-based or anomaly-based detection solution, but we developed the communication between signature-based and anomaly-based detection. To meet our goal, we defined the following research questions (RQ):

- RQ1: What are the existing solutions that combine signature-base and anomaly-base to detect DDoS attack?

  The main goal for answering this question is to discover and highlight the novelty of my work. To address RQ1, we intend to understand how signature-base and anomaly-base solutions can be used to detect various DDoS attack and how those solutions are combined.

  Our methodology to answer RQ1 is by performing a literature research about hybrid-based detection system. We used five sources of information namely: Google scholar, sciencedirect, researchgate.net, International Journal of Engineering Development and Research (IJEDR) and International Journal of Computer Network and Information Security (IJCNIS). We used different set of keywords to find specific papers. In total, we found only nine papers related to hybrid-based attack detection. Although, a variety of techniques have been proposed in the literature to mitigate the DDoS attacks, to highlight the novelty of our proposal, in this document we focus specially on hybrid-based attack detection. Before focusing on hybrid-based attack detection solutions we provide a background on DDoS attacks, signature-based and anomaly-based detection solutions. These backgrounds are essential for a complete understanding of my thesis.

- RQ2: How long the traffic needs to be collected for generating the fingerprint of an attack once the anomaly has been detected?

  The goal of answering this RQ is that a critical element for connecting the anomaly-based solution with the signature based solution is related to several times: (1) the time for an anomaly-detection to notice an attack, (2) the time for collecting enough information about an attack, (3) the time to convert the

collected information into a rule for being used in a signature-based solution, and, finally, (4) the time that a signature-based solution takes to detect an attack previously detected, collected and converted into a rule. This research question (RQ2) addresses the 'enough' time to collect an attack after being noticed by an anomaly-based detection system (time 2). The significance of measuring this 'enough' time is very important in order to generate rule as early and stop the attack. This measurement of time also helps in improving efficiency by storing limited data and earlier mitigation will have less impact on the network.

Our methodology to answer RQ2 relies on using 200 real-time attack .PCAP file, which was captured previously [5] [6]. The main .PCAP file contains multiple attack vectors with different time interval. Each of these attacks was processed separately in order to determine the lowest time for collecting the source IP addresses known as being involved in the DDoS attack. At first, we determine the known source IP addresses involved in the attack using a largely used tool called DDoS Dissector [7]. This tool can process a packet capture (PCAP) data file, which contains packet data of a network and certain characteristics of network traffic flow. Then, we analyze each PCAP into different time interval, to find the lowest time for collecting the known source IP addresses involved in the attack. Our methodology may be biased to the dataset that we used. However, this dataset is the largest public available dataset with DDoS attacks.

- RQ3: How to convert summary of DDoS attack into mitigation rules automatically?

  The main goal of answering this RQ is to understand how summary of an attack used in access list generation and then deploy access list on signature-based device for mitigating the attack. To address RQ3, we intend to use summary of an DDoS attacks which shows all characteristics and used in rule generation. Once anomaly-based detection notices an attack it starts collecting a sample of the attack traffic. Based on RQ2 we know how long this sample should be collected and then get the summary of the attack. Then, this summary of the attack is converted into an rule and applied into a signature-based solution. Thus, rule's which are generated through anomaly-base is then placed to signature-base with automatic loop mechanism to mitigate the attack. This process of transferring knowledge from anomaly to signature-base helps to drop an attack.

  Our methodology to answer RQ3 is by using traffic from anomaly-based solution for generating summary of the attack and this summary is converted

into rules for signature-based solutions, with a proposed algorithm. Finally, we will assess the entire execution flow (including anomaly detection, collection, conversion, and signature-based mitigation). To access the entire flow of the solution we used different tools namely- 'DDoS Dissector'for generating summary of the attack and this tool is widely used and publicly available. Next, we used 'tcprewrite'to re-write the destination IP address to production IP in the attack PCAP and we used 'tcpreplay'to replay each attack. Next, we used 'tcpdump'for capturing the traffic and used in generating summary of the attack. At last, rule converter tool will accept summary of attack as input and generate access list. These access lists are then deployed on signature-based device.

1.2 Thesis Structure

The remainder of this thesis is organized as follows. In chapter 2, we present background information on DDoS attacks and we aim to answer RQ1 (What are the existing solutions that combine signature-base and anomaly-base to detect DDoS attack?) by describing Intrusion Detection Systems and existing solutions that combine signature-base and anomaly-base for detecting a DDoS attack. Then, in chapter 3 we aim to answer RQ2 (How long the traffic needs to be collected for generating the fingerprint of an attack once the anomaly is been detected?) by measuring time for collecting traffic which is required to generate fingerprint of a DDoS attack. After that, in chapter 4 we aim to answer RQ3 (How to convert summary of DDoS attack into mitigation rules automatically?) by explaining proposed DDoS mitigation system and rule generation process for mitigating the attacks. Finally, in chapter 5 we will present overall conclusion and future works.

Chapter 2

Background and Related Work

In this chapter, we aim to present the overall description of DDoS attacks and related work on intrusion detection systems. DDoS attacks is a huge threat to the Internet and many defense mechanisms are used for mitigating them. Attackers modifies tools for bypassing security systems, and researchers in turn alter their approaches for handling new attacks. It is important to understand DDoS attacks and its mitigation techniques. There are two important techniques found in the literature, they are signature-based detection and anomaly-based detection [3]. There is also a hybrid-based detection which is the combination of signature-base and anomaly-base techniques [4]. Each of these techniques have their own way of detecting malicious network traffic.

The organization of this chapter is as follows: First in section 2.1, we present DoS and DDoS attacks which helps in understanding the impact and consequences of this attack. There are many elements an attacker uses while initiating an DDoS attacks. Next, we present classification of DDoS attacks and describe few DDoS attacks which were used to perform the experiment. Then, we present DDoS Architecture that shows basics elements which is required to initiate an attack. There are also different classes of DDoS attacks which is discussed in this section. Then, we present, DDoS Detection methods which is used to detect various type DDoS attacks.

Next in section 2.2, we aim to describe various Intrusion Detection Systems and its related works. At first, we provide background information on Intrusion Detection System in order to understand detection process and explain different types of techniques used for mitigation of DDoS attacks. Then, we describe Signature-Based Detection, Anomaly-Based Detection and Hybrid-Based Detection, followed by Concluding Remarks in section 2.3.

2.1 Background on DDoS attacks

A Denial-of-Service (DoS) attack is an attack which intend to prevent users to access the machine or network. DoS attacks are launched from a single source which could exploit bugs and impact the system. An additional advance version of DoS attack is the Distributed Denial of Service (DDoS) attack. DDoS attacks is composed of multiple systems that performs a synchronized DoS attack on a single target machine. The primary goal of DDoS attack is to limit the access for an application or a service, therefore affect legitimate users who access those services.

Today, organizations of all types and sizes suffers from DDoS attacks which cause an impact for their day to day business. High-volume DDoS attacks are often specially designed in order to escape traditional DDoS protection [8]. Such an attack when by-passes the DDoS protection can enter into the organizations core network and exhaust the resources. Massive DDoS attack would result in loss of critical information, network performance issue, financial losses and brand damage. According to survey performed by Arbor security reported revenue loss for organizations from DDoS attacks nearly doubled in 2017 [9]. Around 10 percent of organizations experienced an attack with cost greater than 100,000 dollar and 57 percent cited damage to their brand from DDoS attack [9]. The consequences of DDoS attack are very severe, with strong defense system we can limit this type of attacks. In the next section, we describe DDoS architecture.

2.1.1 DDoS Architecture

Reviewing and understanding DDoS architecture is considered as important step for deploying appropriate mechanism for detecting the attack in the early launching stage before the attacker exhaust the resources of the victim. A DDoS attack is composed of four elements [10], as shown in figure 2.1

- The attacker.
- The handlers, are compromised hosts with a special program running on them and capable of controlling many agents.
- The attack daemon agents or zombie hosts, which are compromised hosts and responsible for generating a stream of packets for intended victim.
- A victim or target device.

The following steps take place while preparing and performing a DDoS attack:

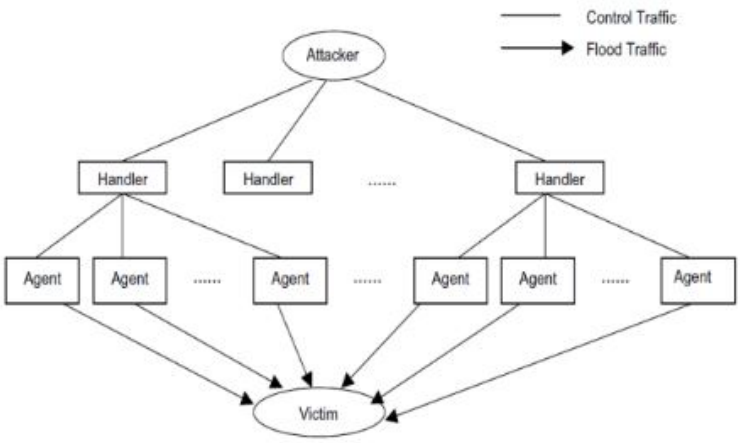


Figure 2.1: DDoS Architecture

1. ***Agents Selection:*** The attacker chooses an agent which can perform the attack and these vulnerable machines allows attacker to gain access and control the device. The attacker attempts to acquire control of these machines with advanced security tools.

2. ***Compromise:*** The attacker exploits vulnerability of the agent machines and runs attack code on it, also attacker takes necessary steps to protect the malicious code from identification and deactivation. The owners of the agent device do not have any knowledge that their device is been compromised and they will take part in a DDoS attack.

3. ***Communication:*** The attacker communicates with number of handlers to check which agents are active, when to schedule attacks. These communication between the attackers and handlers can be through various protocols such as ICMP, TCP, or UDP.

4. ***Reflectors:*** The attacker tries to damage the victim's resources by compelling third-party innocent servers or routers to launch an attack [11]. Attackers can structure their attack traffic to use reflectors for better effect.

5. ***Attack:*** During this step, attacker will initiate the attack and the duration of the attack, unique features of the attack such as the type, length, TTL can be adjusted. These various properties of the attack packets can be helpful for the attacker for avoiding detection.

All these above steps are used by the attacker to prepare and perform a DDoS attack. Each of these steps have specific way to mount the attack. In the next section, we describe classification of DDoS attacks.

2.1.2 Classification of DDoS Attacks

There are two main classes of DDoS attacks namely bandwidth depletion and resource depletion attacks [12]. The classification of different DDoS attacks is shown in the figure 2.2. As shown in figure 2.2, DDoS attack can be classified into bandwidth depletion and resource depletion attacks. Below we describe bandwidth depletion attacks and its types followed by resource depletion attacks and its types.

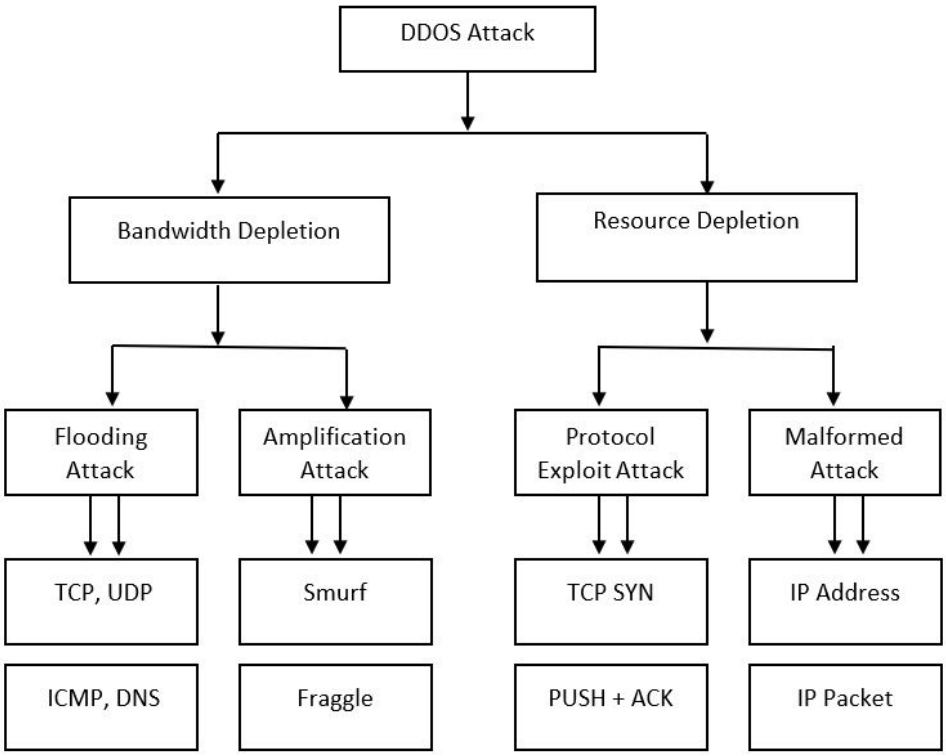


Figure 2.2: Classification of DDoS attacks

A bandwidth depletion attack is intended to flood the network with unwanted traffic which prevents normal traffic reaching to users. Bandwidth depletion is of two types which are known as flooding attacks and amplification attacks [12] [13]. Flooding attacks congest the victim system network bandwidth with IP traffic. Flooding attacks could be various types such as TCP, UDP, ICMP and DNS.

In amplification attack, an attacker is able to use an amplification factor to multiply its power [14]. This attack uses lesser resources for an attacker to cause a significantly higher level of target resources to fail. It also involves the attacker sending messages to a broadcast IP address, using this to cause all systems in the subnet reached by the broadcast address to send a reply to the victim system [13]. Amplification attacks could be various types such as Smurf and Fraggle.

A resource depletion attack is intended to drain out the resources of a victim system thus damage legitimate requests for service [13]. Resource depletion is of two

types which are known as Protocol Exploit Attack and Malformed Attack. Protocol exploit attack exploits the design of TCP communication process between a client and server. Protocol exploit attack could be various types such as TCP SYN, PUSH + ACK. Malformed attack is a type of attack in which attacker guides the zombies to send malicious IP packets in order to harm the victim system. Malformed packet attacks are of two types: IP Address and IP Packet option. In IP address attack each packet contains same source and destination IP addresses which can confuse the operating system of the victim [12]. In IP packet options attack, a malformed packet may randomize the optional fields in IP packet and set all QOS bits to one so that the victim must use more processing time to analyze the traffic [12]. As described above, every main class attack such as bandwidth depletion and resource depletion attacks are classified in to various subclasses of attacks. In the next section, we describe DDoS attacks types.

2.1.3 DDoS Attacks Types

According to top cyber security services companies who provides DDoS protection for various clients, highlighted few most common DDoS attacks [15] [16] [17]. Few them which is relevant to this thesis is described below:

1. ***DNS amplification attack:*** The DNS (Domain Name Server) attack is a reflection-based amplification attack in which attackers spoofs the target server IP address in order to send DNS requests to open DNS resolvers in the Internet. Since DNS request packets are designed in such a way to trigger a response packet which is greater than the actual request packet. Thus, DNS resolvers send responses that are amplified largely as compared to the requests for target server and thus overwhelm with large amounts of unsolicited traffic. It is very difficult to detect this type of attack, since the response traffic looks normal. One way to prevent DNS amplification attack is by tightening DNS server security and block specific DNS servers.

2. ***TCP Flood Attack:*** TCP is a connection-oriented protocol and TCP floods are very popular DDoS attacks. One of the common ways to attack is by sending many SYN packets to the victim. A SYN flooding attacks exploits vulnerabilities found in TCP protocol design. The aim of this attack is to overwhelm session tables of the targeted server. In response to every SYN packet which server receives, it responds with SYN-ACK addressed to each spoofed IP address. The traffic of SYN and SYN-ACK packets will consume bandwidth almost completely. Also, Servers opens a state for every SYN packet which arrives and they store this states in a table which is of limited size. Once the size of the

table reached maximum sessions, it then drops future request which includes legitimate connections.

3. ***UDP Flood Attack:*** UDP is a connectionless protocol and UDP Floods are very popular DDoS attacks. UDP lacks end-to-end connections and makes it vulnerable to a number of DDoS attacks. A UDP flood is a type of DDoS attack in which a large number of UDP packets are sent to a targeted server which overwhelm device capability to process and respond every request. A UDP flood primarily works by exploiting the steps that a sever performs while it responds to a UDP packet which are sent to one of its ports. In response to every UDP packet received by the server, it will utilize its resources in order to process the request. During this type of DDoS attack, an attacker will spoof the source IP address and transmit each packets. As a result, the targeted server utilizes resources to check and then respond to received UDP packet, the server's resources can quickly exhaust when a large flood of UDP packets are received.

4. ***ICMP Flood Attack:*** An ICMP flood is also known as ping flood, one of the common DDoS where an attempt to overwhelm a targeted device with ICMP echo-request packets and affect target by not providing service for legitimate users. Generally, ICMP echo-request and echo-reply messages are used to test the network device health and connectivity. The ICMP attack floods the victim's network with many echo-request packets and equal number of reply packets are responded by victim. This makes incoming and outgoing channels to consume more bandwidth and resulting in a denial of service. The harm of ICMP flood is directly proportional to the number of requests made to the victim. Unlike reflection-based DDoS attacks such as DNS amplification and NTP amplification, ICMP Flood attack traffic is symmetrical, the amount of bandwidth the victim receives is simply the sum of the total traffic that is sent.

All above attacks have different characteristics and their impact varies from each other. Each of this DDoS attack can overwhelm the target and affect legitimate connections. There are still many types of new attacks exists, but these are some of them. In the next section, we describe DDoS Detection.

2.1.4 DDoS Detection

DDoS detection process is a very important step for distinguishing malicious traffic from normal network traffic to perform effective attack mitigation. There are two main types of DDoS attack detection methods, they are signature-based and anomaly-based [3]. Signature-based detection uses predefined signatures of already known

attacks in the database to detect an attack. This detection method is efficient in detecting known DDoS attacks and less effective for new attacks [3]. Anomaly-based detection is based on identifying irregular events with respect to normal system performance. This detection method is capable of handling new attacks which appears in the network. But, selection of threshold value to distinguish between normal and malicious traffic is a tedious task [3]. Detailed detection process of signature-based detection and anomaly-based detection is described in next chapter.

Some of the DDoS detection technologies/solutions are firewalls and intrusion detection system/intrusion prevention system. At first, we describe firewall functionality followed by intrusion detection system/intrusion prevention system. Firewall is stateful devices track all connections for inspection and store them in connection table. Every packet is matched against the connection table to check if it is legitimate connection. Firewalls helps to block malicious IP address and detect the attack. There are two different types of firewall architectures: stateful and stateless firewall. A stateful firewall is can track connection states in order to allow or deny traffic based on access list. A TCP protocol maintains three-way handshake (SYN-SYN+ACK-ACK) for establishing a connection state and stateful firewall can detect those states. If a packet belongs to an existing flow it can be allowed, else it needs to undergo three-way handshake for building a new connection. This will protect client applications by maintaining active connections. New connections are actively inserted into state table and expired connections are removed from the state table maintained by the firewall.

On the other end, stateless firewall does not maintain any connection states for filtering the traffic. Instead it depends on various attributes like source IP and destination IP etc. for making the decision. Stateless firewall process single packet at a time which can be faster and uses less resources compared to stateful firewall. Since, stateless firewall does not have knowledge about connections, they apply access list on all packets passing through the firewall.

An intrusion detection system (IDS) is a device which monitors a network for malicious activity. An Intrusion Prevention System (IPS) is a device of threat prevention technology that inspects a network for malicious activity. The main difference between these two intrusions systems is, IDS can only detect the DDoS attack, but IPS can detect and mitigate DDoS attacks. Thus, firewall and IDS/IPS technology can be used in detecting DDoS attacks based on incoming traffic.

2.2 Related work on intrusion Detection systems

In this section, we aim to describe various Intrusion Detection Systems and its related works. At first, we provide background information on Intrusion Detection Sys-

tem in order to understand detection process. Then, we explain different types of techniques used for mitigation of DDoS attacks. An Intrusion Detection System (IDS) is a software application or a device which monitors a network for malicious activity and raise an alert in case of any policy violations or attacks [3]. There are two important techniques found in the literature, they are signature-based detection and anomaly-based detection [3]. There is also a hybrid-based detection which is the combination of signature-base and anomaly-base techniques [4]. Each of these techniques have their own way of detecting malicious network traffic. This section shows how signature-based and anomaly-based can be used in detecting and mitigating DDoS attacks. The organization of this section is as follows: First, we present Signature-Based Detection, then, we present Anomaly-Based Detection, followed by Hybrid-Based Detection and its related work.

2.2.1 Signature-Based Detection

A signature-based detection method is also known as rule-based detection, knowledge-based detection, pattern detection and misuse detection method [18] . This method uses predefined signatures which contains attack patterns and compares incoming traffic with signature database to identify different attacks. It is capable of detecting ***only*** known attacks. Signature-based detection process works very similar fashion to most anti-virus systems. They maintain database of signatures that triggers a particular type of attack and compare all incoming traffic with signature database. If there is no signature for an attack in the database, such attacks are not detected. Additionally, it uses classification algorithm, rule mining and cost sensitive modeling techniques to reduce the complexity of testing number of packets that need to be inspected [19].

In figure 2.3, we describe signature-based detection process for DDoS attacks. As shown in the figure 2.3, there are four elements namely packet, comparison algorithm, signature database and alert. The comparison algorithm and signature database is the core element of the detection system. Once a packet enters into detection system comparison process is initiated. If a match is found then an alert is produced, in case of no match found then the traffic flows without any problem. In this type of system signature database should be updated regularly in order to detect most recent attacks.

The main advantage of signature-based detection method is, it produces low false positive alert and with less computational power. The disadvantage of this method is not detecting unknown attacks and constant update of signature database is required.

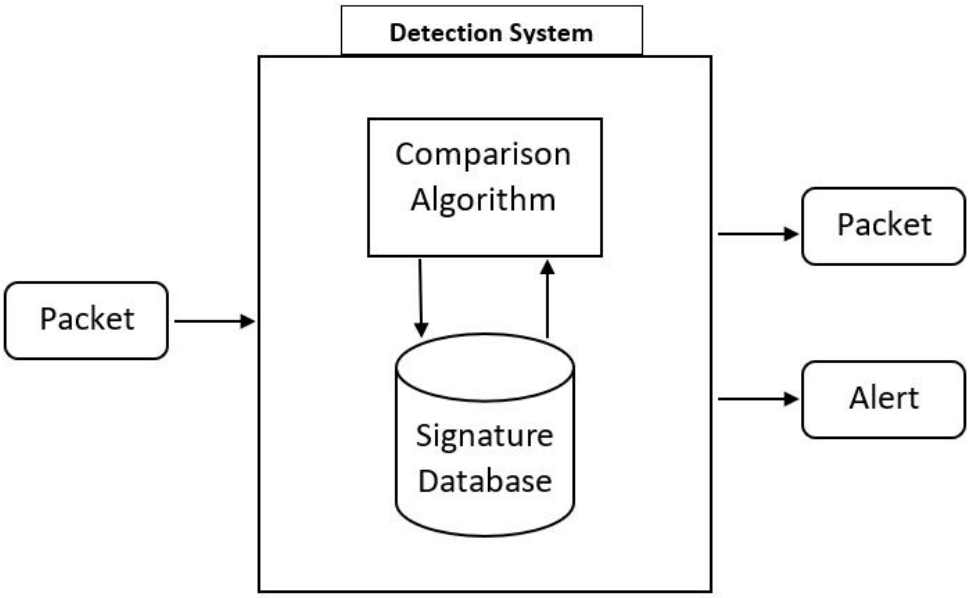


Figure 2.3: Signature-Based Detection.

Signature-Based Solutions:

In previous section we described signature-based concept and detection process. In this section we discuss existing signature-based solutions for detecting DDoS attacks.

Some of the commonly available signature-based solutions are Bro [20] and Snort [21]. Bro is a real-time network detection system, it monitors the network traffic of an intruder [20]. Drawback of this system is to create attack signature manually. Snort is an open source detection system which is based on library packet capture, which is a tool for traffic analyzer [21]. It can perform real-time traffic analysis and detect various attack but it can reduce the performance of the system for large traffic.

Khamruddin al., 2012 [22] presented signature-based DDoS attack detection system. In proposed model routers will try to mitigate different types of DDoS attack on the server. It mainly consists of three steps, at first destination router constantly monitors traffic patterns for attack detection and classification. Next, once destination router detects an attack it balances the load using Network Address Translator. Finally, for mitigating various types of attacks, the signature will be placed to upstream routers and then upstream routers apply the mitigation mechanism depending on type of attack detected. This method also reduces the traffic on the victim machine so that all legitimate users will get the services from destination machine.

2.2.2 Anomaly-Based Detection

Anomaly-based detection method is also known as behavior-based detection. This method refers to the problem of finding patterns in the traffic data that do not behave as expected and raise an alert for abnormal behavior in the traffic pattern [23]. Any deviation detected by the system is flagged and such events are investigated. It is capable of detecting unknown attacks. Anomaly-based detection process involves training and testing phase for a dataset [24]. Additionally, it uses various technique such as machine learning, data mining, Bayesian networks, clustering, computational intelligence and different classification algorithms as a base support [25] [23].

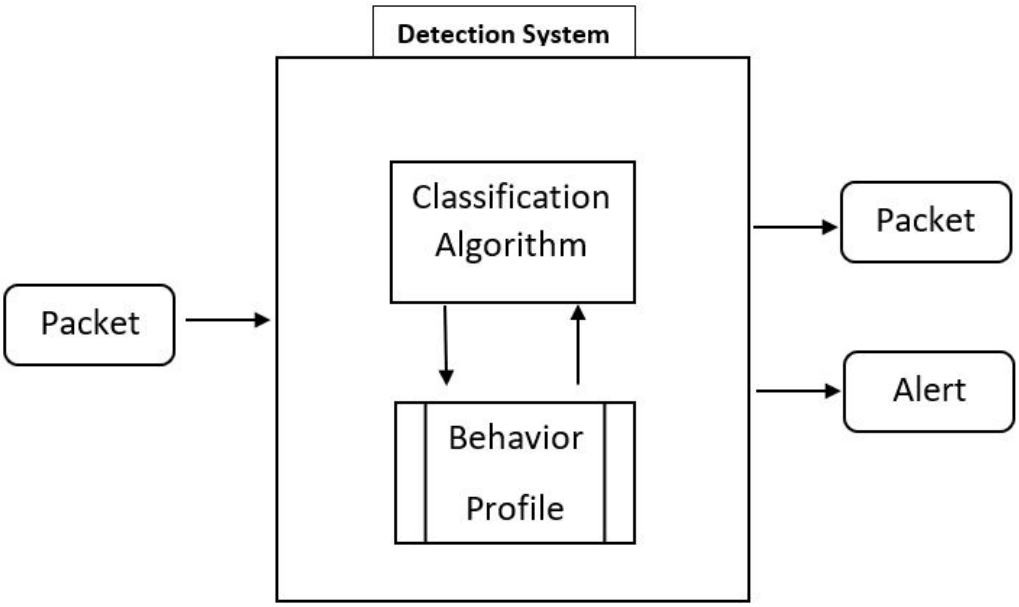


Figure 2.4: Anomaly-Based Detection.

In figure 2.4, we describe anomaly-based detection process for DDoS attacks. As shown in the figure 2.4, there are four elements namely packet, classification algorithm, behavior profile and alert. The classification algorithm and behavior profile is the core element of the detection system. It first classifies the data using different classification algorithm and creates a behavior profile which represents the normal behavior of the traffic. Once a packet enters into detection system comparison process is initiated. System starts to compare the regular traffic with the behavior profile to find any deviation. If traffic is found to be abnormal from the normal behavior, then an alert is been triggered by the system suspecting an intrusion.

The main advantage anomaly-based detection compared to signature-based detection method is: it detects unknown attacks. The disadvantage of this method it is uses trained dataset and produces high false positive alerts.

Anomaly-Based Solutions

In previous section we described anomaly-based concept and detection process. In this section we discuss existing anomaly-based solutions for detecting DDoS attacks.

Chaitanya al., 2015 [26] presented anomaly-based DDoS attack detection system. Anomaly-based detection is based on three analyzers. They analyze deviation from standard behavior of the network traffic, any deviation is flagged and raise an alarm. The testing of the setup was done with CAIDA dataset [27]. This system successfully identifies DDoS attack and reduce alarm rate.

Cabrera et al.,2001 [28] proposes a methodology for early DDoS detection through Network Management Systems. They focused on Management Information Base (MIB) traffic variables such as ip, icmp, tcp, udp and snmp which are collected from the systems participating in the Attack. A cluster of attack signatures were extracted based on three-step signature extraction method. Using this signature malicious traffic and legitimate traffic were identified. This scheme can detect statistical irregularities for different packet specific to TCP, UDP and ICMP which occurs in DDoS attacks.

Hwang al., 2003 [29] presented anomaly-based detection method which is based on multi-site correlation and alarm-matrix framework for evaluating various attack scenarios. DDoS detection here relies on protocol violated, insecure source IP and source IP redundancy. Simulation is performed using NetShield defense system with data collected from USC Information System Division [30]. This system moderately detects DDoS attacks with less false alarms.

Basant Agarwal, 2012 [23] proposed a detection technique that combines entropy of network features and support vector machine. The signature-based detection module is not used in this system, instead they use anomaly-based detection system which is based on the Entropy of network features and Support Vector Machine (SVM). This setup uses DARPA data set for evaluating the setup [31]. This system is not considered as hybrid-based detection, since there is no signature-based detection module.

In this section we discussed anomaly-based works. In the next section we will describe Hybrid-Based Detection.

2.2.3 Hybrid-Based Detection

Hybrid-based detection method is a combination of signature-based and anomaly-based detection method. This method combines signature-based and anomaly-based to improve the overall detection accuracy. It helps in detecting both known and unknown attacks. This system attempts to maximize the capability of the IDS while

reducing their drawbacks. There are many hybrid detection techniques proposed for DDoS attack detection and results vary depending upon the technology and dataset [32].

In figure 2.5, we describe hybrid-based detection process for DDoS attacks. As shown in the figure 2.5, there are four elements namely packet, signature-based detection, anomaly-based detection and alert. The signature-based detection and anomaly-based detection are the building block of hybrid-based detection. Each of them has set of elements used during detection process. Signature-based detection is composed of comparison algorithm and signature database. Anomaly-based detection is composed of classification algorithm and behavior profile. Once a packet enters into detection system, it first reaches to signature-based detection for detecting known attacks and unknown attacks are sent to anomaly-based system for detection. If the traffic found to be malicious, then an alert is been triggered by the system suspecting an intrusion.

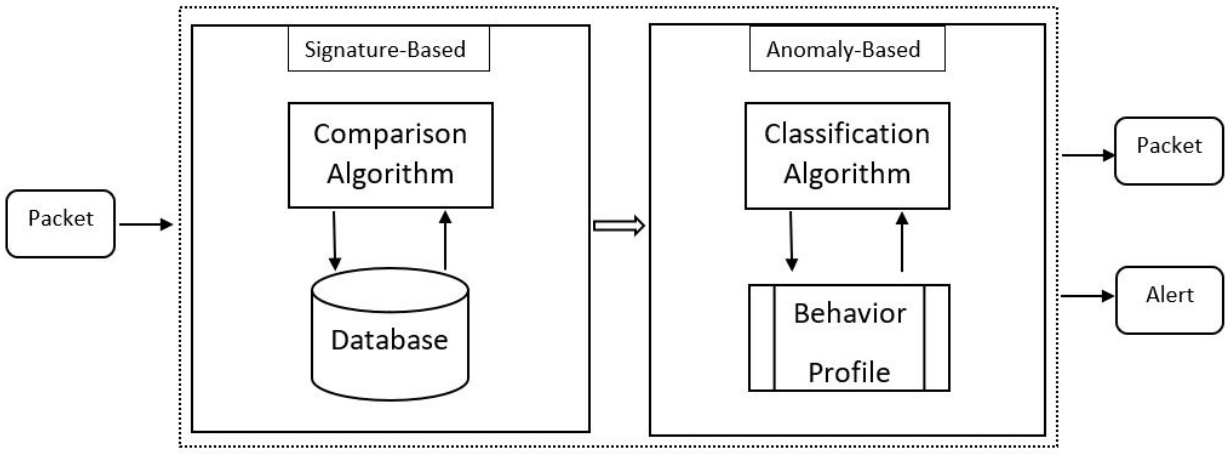


Figure 2.5: Hybrid-Based Detection.

The main advantage of hybrid-based detection method it detects DDoS attacks in efficient manner and reduces number of false positive. The disadvantage of this method it is uses trained dataset, high in complexity and computation cost.

The signature-based detection and anomaly-based detection are two well-known techniques used in a network. To obtain better detection rate and accuracy hybrid-based detection method is used. Table 2.1 shows summary of all are three techniques.

This table comprises of three features for each type of detection system namely: new attacks, false positive (FP) for known attack, false positive (FP) for unknown attack. we explain each column against each row. At first, we begin to explore new attacks, which are not easily detected in signature-based, but such attacks can be detected in anomaly-based and hybrid-based. Then, false positive for known attacks

Table 2.1: Summary of Detection Methods

Detection Method	New Attacks	FP for Known Attack	FP for Unknown Attack
Signature-Based	No	Low	High
Anomaly-Based	Yes	Varies	Varies
Hybrid-based	Yes	Low	Low

is low in case of signature-based and hybrid-based, but varies in anomaly-based. Similarly, false positive for unknown attacks is high in case of signature-based and low hybrid-based, but it varies in anomaly-based based on type of algorithm system uses.

2.2.4 Methodology for Searching Literature

In this section, we explain the methodology on finding literature about hybrid-based detection system. We decide to use five sources of information namely: Google scholar, sciencedirect, researchgate.net, International Journal of Engineering Development and Research (IJEDR) and International Journal of Computer Network and Information Security (IJCNIS). We performed this decentralized search because Google scholar did not return papers that are related to hybrid-based detection system which address DDoS attacks. We used different set of keywords to find specific papers.

At first source, in Google scholar we used “DDoS Attack Detection with Signature Generation”keywords and the top 1 paper was the one by Katkar, 2012 [33]. From the references used in this paper we got four more papers by Shanmugam, 2009 [18], Brahmi, 2010 [25], Yang al., 2010 [24] and Ding, 2009 [32]. At the second source, in sciencedirect we found a paper which deals with DDoS attack and hybrid-based detection system. The keywords “Intrusion Detection System using data mining techniques”retrieved top 1 paper by Nadiammai, 2014 [34].

For third source, in researchgate.net we used “Hybrid Intrusion Detection System for DDoS Attacks”keywords and it retrieved the paper by Cepheli al., 2016 [35]. For fourth source, in IJEDR we used “Hybrid Intrusion Detection System”keywords and it retrieved paper by Pawar B, 2015 [36]. For final search, in IJCNIS we used “Hybrid Intrusion Detection System”keywords and it retrieved paper by Tesfahun, 2015 [37]. In total, we collected nine papers related to Hybrid-based detection. Overall, a variety of techniques have been proposed in the literature and implemented by many researchers to solve the DDoS detection problem. Below are the papers review and followed by summary table.

2.2.5 Hybrid-Based Solutions

In this section we discuss existing solution for detecting DDoS attacks. Each of this paper highlights the signature-based detection and anomaly-based detection approach used in detecting DDoS attacks.

Brahmi, 2010 [25] presented hybrid-based detection system in which signature-based detection module is composed of mobile agent for detecting known attacks and anomaly-based detection module is composed of clustering-based techniques. The author used DARPA data set for evaluating the setup [31]. In this system database of signatures is updated regularly and system is capable of detecting various attacks with less false rate.

Nadiammai, 2014 [34] also proposed an effective approach for detecting DDoS attack. The signature-based detection module is composed of SNORT for detecting profile-based attacks and anomaly-based detection module is composed of efficient data adapted decision tree algorithm techniques. This setup uses KDD99 data set which is a widely used publicly available data sets for network-based anomaly detection systems and evaluation purpose [38].

Ding, 2009 [32] presents hybrid intrusion detection system that detects various intrusions. The signature-based detection module is composed of SNORT for detecting profile-based attacks and anomaly-based detection module is composed of frequent episode rule and Apriori algorithm. This setup uses KDD99 dataset for testing purpose [38]. This system performs well in the offline attack detection [32].

Katkar, 2012 [33] presents hybrid intrusion detection system with signature generation process. The signature-based detection module uses known attack signature DB and LogDB which contains all connection records and anomaly module is composed of Apriori algorithm. This setup uses KDD99 dataset for testing purpose [38]. This system focuses only resource consumption-based attacks and does not support for different attacks [33].

Pawar B, 2015 [36] proposed hybrid-based intrusion detection method by comparing attributes of each packet. The signature-based detection module uses signature database method to store all signatures of detected anomalies and anomaly-based detection module is composed of Apriori algorithm. This setup uses KDD99 dataset for testing purpose [38]. The main limitation of this system is it does not guarantee to detect unknown attacks [36].

Shanmugam, 2009 [18] proposed improved hybrid detection system. The signature-based detection module uses fuzzy inference engine and anomaly-based detection module is composed of Apriori algorithm techniques. This system uses DARPA and live dataset for evaluating the setup [31].

Yang al., 2010 [24] proposed hybrid intrusion detection system based on protocol analysis. The signature-based detection module is composed of protocol analysis,

misuse detection engine and algorithm selector and anomaly-based detection module is composed of decision tree technique. This setup uses KDD99 dataset for testing purpose [38].

Tesfahun, 2015 [37] proposed effective hybrid intrusion detection system using feature selection. The signature-based detection module uses random forests classifier and anomaly-based detection module is composed of decision tree technique. This setup uses NSL-KDD dataset which is an enhanced version of KDD99 dataset [38]. This system is not adaptive for dynamic attack scenarios [37].

Cepheli al., 2016 [35] proposed hybrid intrusion detection system using feature extraction. The signature-based detection module is molded with SNORT and anomaly-based detection module is composed of Gaussian mixture model to distinguishes normal and abnormal traffic in the data. This system uses DARPA dataset for testing the output of these detectors [31]. Detection success solely depends on the anomaly detector when unknown attacks seen [35].

Several methods listed above showed how Hybrid-based detection method can be used in detecting DDoS attacks. Table 2.2: shows the summary of above methods and point critical aspects of each method.

Table 2.2: Summary of Hybrid-based detection method

References	Signature-based	Anomaly-based	Data-Set
[25]	Mobile agent	Clustering	DARPA [31]
[34]	SNORT	Decision tree	KDD 99 [38]
[32]	SNORT	Apriori algorithm	KDD 99 [38]
[33]	Signature database	Apriori algorithm	KDD 99 [38]
[18]	Fuzzy inference engine	Apriori algorithm	DARPA, Live [31]
[36]	Signature database	Apriori algorithm	KDD 99 [38]
[24]	Protocol analysis	Decision tree	KDD 99 [38]
[37]	Random forests classifier	Decision tree	NSL-KDD [38]
[35]	SNORT	Gaussian model	DARPA [31]

2.3 Concluding Remarks

The goal of this chapter was to introduce DDoS attacks and related work on intrusion detection systems. At first, this chapter describes DDoS attack and its consequences followed by DDoS attacks types and DDoS detection. DDoS attacks is on raise and affect business by bringing down the service with huge financial and brand damage. DDoS Architecture mainly composed of attacker, handlers, attack agents and victim. The primary steps involved for preparing and performing a DDoS attack

are agents selection, compromising, communication, reflectors and crafting the attack. There are two main classes of DDoS attacks, which are bandwidth depletion and resource depletion categories. Each of these categories are having various types of attacks. Bandwidth depletion consists of flooding attacks and amplification attacks. Resource depletion consists of protocol exploit attacks and malformed attacks. Amplification and flooding attacks are most common DDoS attacks which can impact and should be mitigated well in time. These attacks can be in high volume and exhaust the resources which affects legitimate user requests. We can use various technologies such as firewalls and IDS/IPS in order to detect different classes of DDoS attack and thus limit its damage. In this thesis, we use signature-based and anomaly-based techniques for detecting and mitigating DDoS attack.

Second, we presented intrusion detection systems and related work. We described Intrusion Detection Systems and its classifications followed by various solutions available in the literature. Signature-based detection uses predefined signature to identify different attacks, this system is efficient in detecting known attacks and it functions similar to a traditional anti-virus system. The main elements of signature-based detection are comparison algorithm and signature database used to compare the incoming packets and raise an alert once an anomaly detected. The main advantage of signature-based detection system is it can detect known attacks and the disadvantage is it cannot detect unknown attacks. Existing solutions shows different approaches for signature-based detection which mainly uses Bro and SNORT for DDoS detection. Anomaly-based detection uses patterns to identify different attacks and is capable of detecting unknown attacks. The main elements of anomaly-based detection are classification algorithm and behavior profile for comparing the incoming packets and raise an alert once there is a deviation. The main advantage of anomaly-based detection system is it can detect unknown attacks and the disadvantage is it uses trained dataset and produces high false positive alerts. Existing solutions shows different approaches for anomaly-based detection which mainly uses data mining approach for DDoS detection.

Hybrid-based detection is a combination of signature-based and anomaly-based detection method used in detecting DDoS attacks. This system helps in detecting both known and unknown attacks and maximize the capability of the IDS while reducing their drawbacks. The incoming packets are processed with signature-based and anomaly-based detection system, in case of any malicious traffic an alert is created suspecting an intrusion. The main advantage of hybrid-based detection system is it can detect known, unknown attacks. We found that there is no knowledge transfer from anomaly-based detection to signature-based detection. Existing solutions shows different approaches for hybrid-based detection which mainly uses data mining and Gaussian model approach for DDoS detection. Each of these tech-

niques have their own advantage and disadvantage in detecting known and unkown attacks. Once an anomaly is detected in the network, we need to understand how long we need to collect the traffic for generating the fingerprint. In the next chapter, we will discuss time analysis on traffic collection for generating fingerprint of an attack, which shows duration required to generate summary of an attack and this summary is used for creation of access list.

Chapter 3

Time Analysis for Traffic Collection

In previous chapters we described DDoS and Intrusion Detection Systems Related Works. The background provided in previous chapters will help in understanding various concepts and detection solutions for DDoS mitigation. Literature research helps in considering existing solutions and bring novelty in current work. In this chapter we will answer **RQ2:** ***How long the traffic needs to be collected for generating the fingerprint of an attack once the anomaly is been detected?*** Measurement of anomaly time plays an important in our setup for capturing limited amount of traffic for fingerprint generation. Since DDoS attacks have different behaviors it is important to understand to which type of attacks how long we need to capture the traffic. This time can help in stopping the attack as early as possible by generating the rule, as earlier the mitigation less will be the impact on the network. Determination of time can also help in less data storage and avoids to measure complete traffic.

The rest of this chapter is organized as follows. In section 3.1 we present Proposed Solution. Then, in section 3.2 we present, Methodology. Then, in section 3.3 we present, Results. followed by Concluding Remarks in section 3.4

3.1 Proposed Solution

This section describes proposed solution for mitigating DDoS attacks and different phases involved in order to mitigate the attack. The proposed systems would overcome the limitations of existing methods and enhance overall detection rate with higher accuracy. This system is a hybrid model which consists of signature-based and anomaly-based detection technique for faster detection and mitigation. On one hand, known attacks are filtered in signature-based detection method. On the other end, any novel attack that is not detected by the signature-based is directed to anomaly-based detection which triggers an alarm and collects the abnormal traffic. After that we propose a solution that use 3 phases, namely phase 1 for collecting

enough attack traffic, phase 2 used to summarize the attack traffic and phase 3 for generating the mitigation rules. In this chapter we mainly focus phase 1 on how much traffic needs to be collected for generating the fingerprint of an attack.

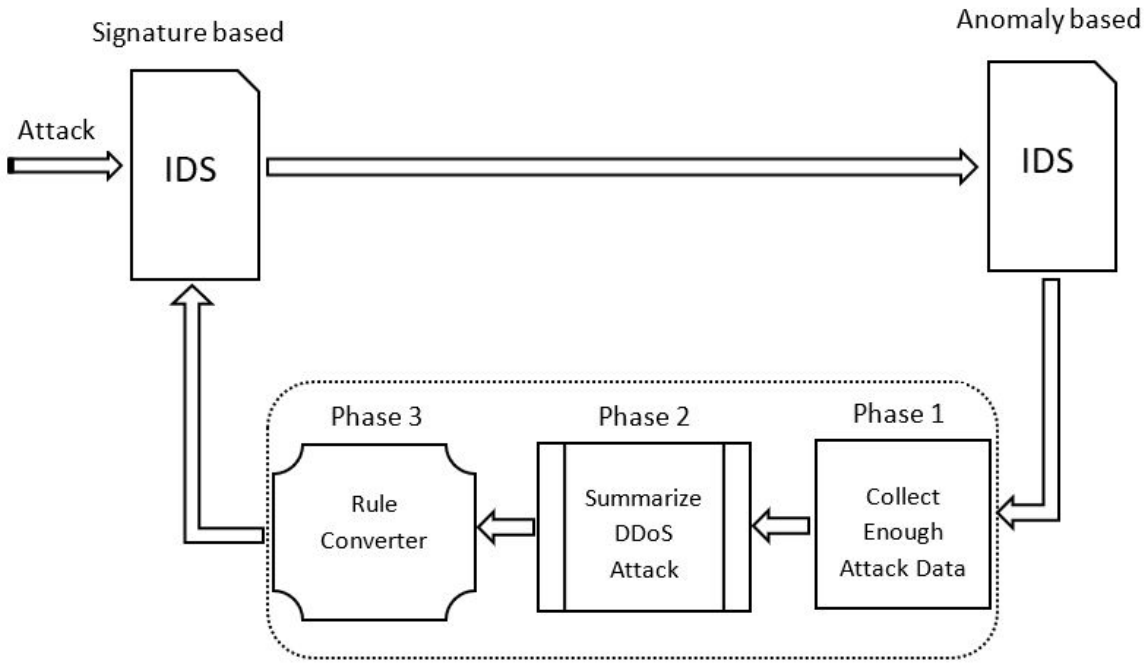


Figure 3.1: Proposed Solution.

As shown in figure 3.1, proposed model consists of three phases-

- Phase 1: We use to collect attack traffic which is detected from anomaly-based system. How long the traffic needs to be collected depends upon the type of attack. In different time range, an administrator chooses to capture the attack data.

- Phase 2: A tool which can summarize the DDoS attack and thus provide all characteristics. This information can contain source IP address, destination IP address, ports and protocol.

- Phase 3: Rule converter generates technology specific rule from obtained fingerprint. This Rule converter generates rules for various types of attacks.

Our proposed solution creates a logical connection from anomaly-based to signature-based detection. In the next section, we will describe Methodology used in determining the time for collecting enough traffic.

3.2 Evaluation Methodology

In this section, we the present the methodology used to answer the RQ2 (How long the traffic needs to be collected for generating the fingerprint of an attack once the anomaly is been detected?). Each attack is processed separately in order to determine the lowest time for seeing complete list of source IP address. Since DDoS attacks is a repetition of network traffic coming from the same source of IP addresses, we need to have unique and complete source IP list.

We make use of DDoS Dissector tool for analyzing every DDoS attack from a PCAP file [7]. This tool is widely used and available for public usage through GitHub. It can accept larger PCAP file and generates fingerprints and separates the attack vectors. DDoS Dissector process a packet capture (PCAP) data file, which contains data of a network and certain characteristics of network traffic flow. DDoS Dissector can identify and generate summary of each attack and this summary is used as an input for creating access list. We rely on the correctness of the DDoS Dissector. Therefore, we take its execution and results as correct for generating the rules. In the remaining of this section, we present dataset which is used in this analysis, followed by the steps that is required to determine the time.

3.2.1 Dataset

The dataset used in this analysis is a composition of the dataset made publicly available by [5] [6]. Both datasets are composed of DDoS attacks purchased from websites that offer DDoS as a Service, also called Booters. All these datasets were in .PCAP file and we selected 200 attacks based on source IP address count and protocol. The .PCAP file contains multiple attack vectors with different time interval and DDoS Dissector can filter each type of attack to a separate PCAP file which contains unique attack.

Figure 3.2 shows the distribution of different types of DDoS attacks. Each bar shows the number of DDoS attack. Data consists of totally 200 attacks of different types and some of the attack types which is used in this analysis are DNS attack, Chargen attack, SSDP attack, NTP attack, UDP attack, ICMP attack and TCP attack. There were four sub types of UDP based attack namely: DNS, Chargen, SSDP and NTP attacks. We considered only these seven types of attacks based on the data and some attacks were less in number. Each of this attack were of different time interval starting from 1 to 300 seconds.

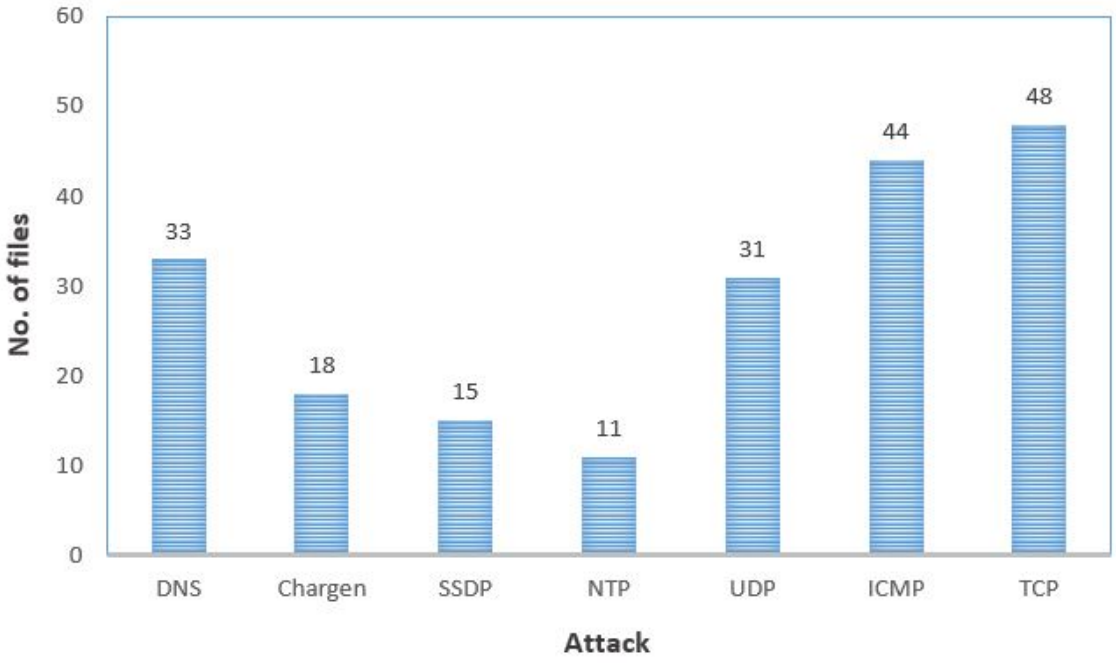


Figure 3.2: Total number of attack type.

3.2.2 Data exploration

The dataset contains various DDoS attacks and each of them are processed separately. We performed below steps in order to determine the minimum amount time for getting complete list of source IP address. These complete list of source IP will help in generating summary of the attack and create access control list to drop the attack.

Step 1: Get attack PCAP file and process it in DDoS Dissector to get specific attack. Each PCAP file can have many attack vectors and each of them is considered as specific attack.

Step 2: Determine maximum source IP count for given PCAP, which is then used for calculating in terms of percentage.

Step 3: Since there are PCAP's with different time intervals we choose to split PCAP having a specific attack in to three different levels: 1 second (used between 1 and 10 seconds), 10 seconds (used between 10 and 60 seconds), and 60 seconds (used starting from 60 seconds till the end of the network trace).

Step 4: We then process PCAP's starting from lowest time to highest in order to determine the complete list of source IP address.

We mark time in seconds and process each individual PCAP in determining source IP count that matches to the maximum source IP count value (Step 2). Once we reach maximum source IP count, we stop further processing and record the time entry. So, for an attack recorded time entry would provide complete list of source IP

address. This also avoids to analyze given PCAP for complete time interval which eventually saves the resources. This process is repeated for all the 200 attacks and generate separate values. Figure 3.3 shows an example for Chargen attack and how values are generated. At first, maximum source IP count is determined which is 1086 in this example. Then, every PCAP source IP count is recorded in table under Src-IP column. At every time interval the source IP count is matched against the maximum source IP count and percentage is calculated for source IP address. Thus, this table also shows the summary of number source IP's that are captured for every time interval.

File Content				
	Src_IP	Sec	MAX	Percent
0	576	1	1086	53.04
1	783	2	1086	72.1
2	864	3	1086	79.56
3	908	4	1086	83.61
4	940	5	1086	86.56
5	958	6	1086	88.21
6	980	7	1086	90.24
7	997	8	1086	91.8
8	1008	9	1086	92.82
9	1011	10	1086	93.09
10	1046	20	1086	96.32
11	1053	30	1086	96.96
12	1065	40	1086	98.07
13	1065	50	1086	98.07
14	1065	60	1086	98.07
15	1080	120	1086	99.45
16	1086	180	1086	100

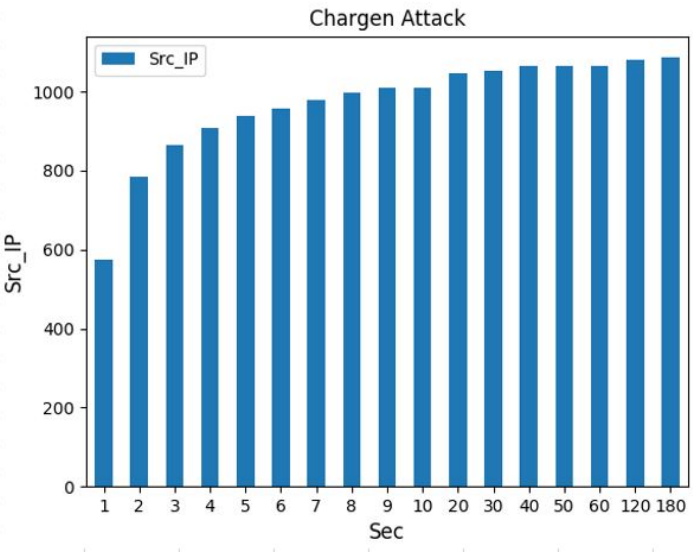


Figure 3.3: Example for Chargen attack.

3.3 Results

This section describes the result of the time analysis for different attack types, followed by discussion and observations on the result. The results in this section aims to provide answer to RQ2 (How long the traffic needs to be collected for generating the fingerprint of an attack once the anomaly is been detected?). We are in the phase of determining the time for collecting the traffic and then we convert this traffic into a fingerprint that shows summary of the DDoS attack. Next, we write mitigation rule with use of fingerprint which will be discussed in chapter 4. In figure 3.4, we present time analysis for different DDoS attacks.

Figure 3.4, is having uneven size for various DDoS attacks. The small bar shows that majority of attacks are covered in that segment and big bar depicts that attacks are spread across the bar. The figure 3.4, shows the time required to capture all

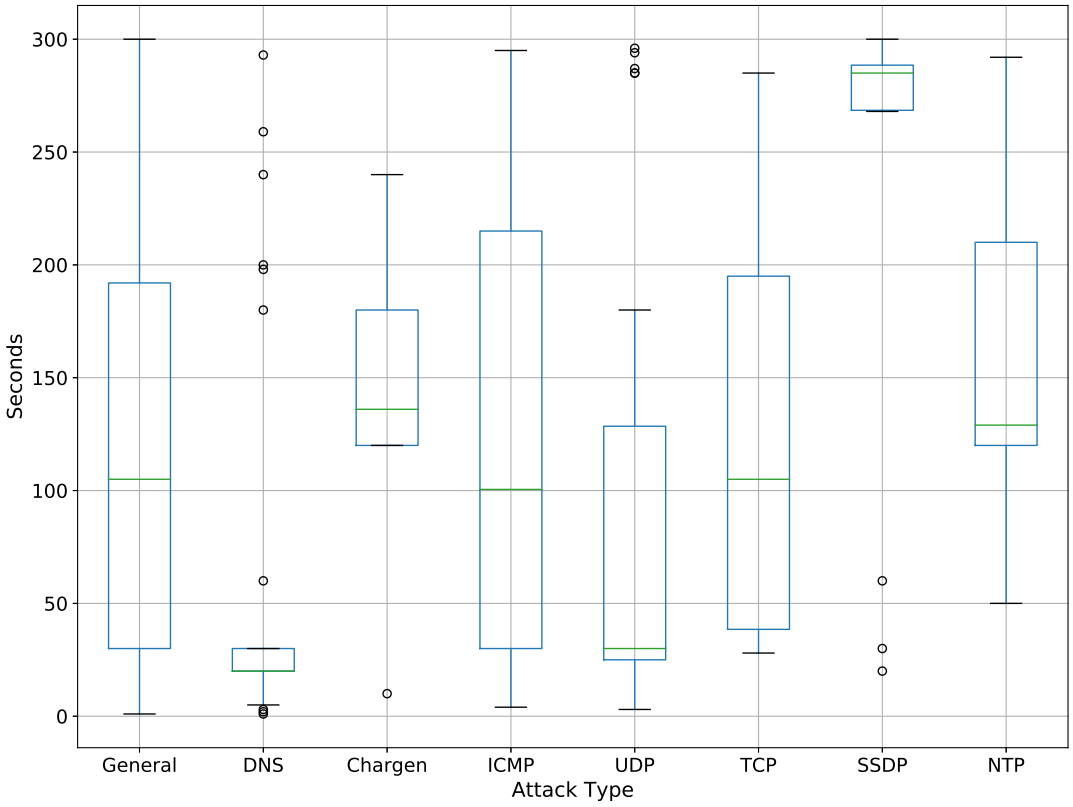


Figure 3.4: The time required to measure all the source IP's.

source IP addresses participating in (known) DDoS attacks. The results are presented in a box-plot visualization. The results were grouped per type of attack (i.e., DNS, Chargen, ICMP, UDP, TCP, SSDP, and NTP). From administrator perspective, the attack type is not known until it is measured and classified. In that case administrator can choose highest time, which is 300 seconds in this case for capturing complete source IP list. But, this approach might not feasible in some cases, because few attacks does not require 300 seconds, instead 30 seconds would be sufficient in obtaining complete source IP list. We narrow down and try to analyze the distribution per attack type. There are seven types of attacks. Following, we analyze the observations for each of them.

Considering 33 DNS attack traces, we observed that all IP addresses were collected within 30 seconds for the majority of attacks. This value is the upper-bound of the box-plot (75% of attacks–third quartile). We also observed some outliers. It means that the 30 seconds will not be sufficient to observe all the IP addresses involved in all the attack in our dataset. In this case, one way to overcoming this

problem is by creating partial rule that already has some amount of captured source IP address. If a rule is deployed then that source IP's which we measured will be blocked and the IP which we did not collect they will still keep entering into the network and then administrator again captures the traffic for obtaining remaining source IP's participating in the attack.

We considered 18 Chargen attack traces and observed that all IP addresses were collected within 240 seconds and common time range was in between 120 sec to 240 seconds. However, for ICMP attack the time range was in between 10 to 290 seconds which was longest amongst all attacks and without any outliers. For UDP attack we considered 31 attack traces and the time range was in between 10 to 180 seconds, being the second-best result after DNS attack. For TCP attack we considered 48 attack traces and the time range was in between 30 to 280 seconds, no outliers are detected across the data points. Next, for SSDP attack we used 15 traces and results was surprising. The minimum value for SSDP is 270 seconds, which we considered to be long time for capturing the attack and we assume that attacks will be repeating with the same source IP address. Everything attack is having lower time except SSDP attack. Finally, for NTP attack we used 11 attack traces and no outliers are observed in this graph, minimum time is 50 seconds and maximum time is 280 sec, One of the important observation is, if I measure an attack for 10sec I could see total number of source IP's, however it doesn't mean that I have full set of IP's. It could be a full set of source IP's or it could be that I did not reach the one barrier/next one.

During analysis of each type of attack, we observed that some of these results are very close to each other. In figure 3.4 DNS attack and SSDP attack is having small bar that is concentrated at one point, while DNS attack concentration is below 20 seconds and SSDP concentration is higher than 250 seconds. It means that I have more chance to measure all source IP's in low time for DNS attack and invest more time for measuring SSDP attack. There are also some attacks right at the beginning I could measure observe all source of an attack, but it's a very small minority. Majority of attacks shown in the graph have minimum time of less than 60 seconds. Only SSDP is surprisingly high with 270 seconds. Looking at the graph we can say that, if we measure 30 seconds, we get complete source IP list that are involved in DNS attack, which is not true for all other attacks. There are some exception with the outliers, example: we see 7 outliers in DNS attack which is not having a normal behavior. We have the position of median that counts half of the values in our dataset. It means that, if the median is as low as possible, example: DNS attack is having the best value for measuring the attack traffic, but with some exceptions. It is observed that median value is very close to minimum time of DNS, Chargen, UDP and NTP attacks. However, this is not true for ICMP, TCP and SSDP

attacks, which are crossing more than 100 seconds.

The results show three different values an administrator can select for obtaining source IP address., namely minimum, maximum and median value. Capturing for minimum time leads to less resource consumption and faster mitigation. However, this minimum time does not give complete source IP address and administrator needs to recapture the remaining traffic for complete source IP list. This method initiates mitigation process for every capture and thus contributes to multiple device memory. Capturing for maximum time leads to larger processing power and delay in the mitigation process. However, captured data is guaranteed to give complete source IP list and this method will avoid recapturing the traffic for multiple time. The administrator can select maximum time for obtaining complete source IP addresses. Since the captured data is very high, it requires larger processing power and consequently mitigation process will be much longer. Alternatively, administrator can choose a time which is in between minimum and maximum time range (median value) that guarantees to give 50% of the source IP address. The remaining IP address participating in the attack will be captured and perform the mitigation process. This method avoids multiple capture of the attack traffic for mitigation. From an operational point of view as much as low time, faster will be the mitigation.

The limitation of this approach is, it does not guarantee to capture all source IP's participating in an attack when administrator sets a particular value (in this case 30 seconds). So, if 30 seconds is considered for capturing the complete source IP list there might be possibility of not getting entire source IP list, in that case administrator should recapture the traffic. While recapturing administrator can observe remaining amount source of IP's and creates additional rule for mitigating the attack. All results which I have obtained could be biased to my dataset, if there is other dataset then there is a chance of obtaining different results.

3.4 Concluding Remarks

The goal of this chapter was to answer RQ2 (How long the traffic needs to be collected for generating the fingerprint of an attack once the anomaly is been detected?). The time analysis for collecting 'enough'traffic determines when all the (previously) known IP addresses involved in an attack are observed. To answer RQ2, we spilt this chapter into three sections. In section 3.1, we discussed about proposed solution for mitigating DDoS attacks. Our proposed solution consists of three phases: phase 1 for collecting enough attack traffic, phase 2 used for summarizing the attack traffic and phase 3 for generating the mitigation rules. We collect the attack traffic which is detected from anomaly-based system and time to collect the traffic depends upon the type of attack. Our proposed solution creates a logical

connection between anomaly-based and signature-based detection.

Then, in section 3.2 we discussed evaluation methodology Every DDoS attack is unique and they have different behaviors and characteristics. We analyzed 200 attacks with time range of 1 to 300 seconds. Every attack was separately analyzed the results where combined to an attack type. Each attack type showed different size, some attacks capture time is relatively short and few were tall. Then, in section 3.2 we discussed the results obtained during the analysis. Some of the key observations are DNS attack and SSDP attack is concentrated at one point and both are in opposite direction. The maximum time to capture traffic was 30 seconds for DNS and 300 seconds for SSDP attack. ICMP attack distribution was the longest amongst all other attacks. The median value is minimum for DNS, Chargen, UDP and NTP attacks some outliers are observed in DNS, Chargen, UDP and SSDP types of attacks. For DNS attack, if we choose the time of 30 seconds, we will be able to observe all the source IP address. However, for some attacks we will not able to see all the source IP address participating in the attack, which are considered as outliers. From the analysis we can find that in less than 50 seconds we can obtain minimum amount of source IP address for all type of attacks, except SSDP attack. The lowest value of all attack types shows something surprise results, SSDP attack minimum time is 270 seconds, which is considered to be high. To get complete list of source IP, we need to consider highest time. The overall result of this analysis shows the maximum time required to capture attack traffic is 300 seconds which guarantees to give complete source IP which is participating in the attack. However, the implication of measuring this amount of time, is that we possibly collect more attack data for generating the fingerprint. In the next chapter, we will discuss Rule Converter that generates access-control lists using DDoS fingerprint.

Chapter 4

Automatic Generation of ACL's from Summarized DDoS Attack Information

In the previous chapter, we observed that 300 seconds were 'enough' time to capture all source IP addresses involved in DDoS attacks (based on the dataset that we had access). In this chapter, first, we explain how we summarize DDoS attack data for filtering only information to be used in mitigation rules. After that, we propose an automated algorithm that converts these summaries of attacks into mitigation rules, called in this chapter Access-Control List (ACL). This algorithm takes into consideration memory limitations of the software/hardware that will deploy the ACL. Therefore, this algorithm addresses a minimization problem very relevante to network operators that will mitigate DDoS attacks and help us to answer the RQ3 defined in the introductory chapter of this thesis (***How to convert summary of DDoS attack into mitigation rules automatically?***).

The remainder part of this chapter is organized as follows. In section 4.1, we discuss how to summarize information in DDoS attack previously collected. After that, in section 4.2, we present rule converter and its requirements for automatically converting summary of attack data into ACL's. Next, in section 4.3, we present our rule generation process used in creation of ACL's. Then, in section 4.4, we present our experimental setup. Then, in section 4.5, we present the metrics used to validate our algorithm. Subsequently, in section 4.6, we present impact of ACL's Finally, in section 4.7, we present our concluding remarks.

4.1 Summarizing DDoS Attack Data

For applying the mitigation rule, we should have some specific characteristics of a DDoS attack. These characteristics contains IP address, port and protocol. A captured traffic (PCAP) also have similar information, but the data is not well organized and there are chances of repetition of IP addresses. Therefore, in this thesis we decided to use DDoS Dissector tool to summarize the DDoS attacks. DDoS dissector is clever and able to distinguish the attack. This tool is already available for public usage through github and it used by many organizations. DDoS Dissector is developed in python code and it can accept many formats of data i.e, PCAP, nfdump and netflow as input file. For this thesis, we assumed that DDoS Dissector is working as expected and no errors.

DDoS Dissector is capable of analyzing large file, but one file at a time. It first accepts PCAP file and starts the analysis by discovering the target (destination IP). It is discovered based on how many packets the destination IP receives. So, now this destination IP is selected and start analyzing the protocol through Tshark utility, which gives the protocol information. Next, analysis is in search of source port and destination port. This analysis continues for further identification and determines possible source IP participating in the attack.

The output of the analysis is the summary of characteristics, which is known as Fingerprint saved in a JSON file. In figure 4.1 shows a sample fingerprint of a UDP attack and fingerprint shows various information of the DDoS attack. The protocol in this example is UDP attack, then it shows the source IP list which is truncated output in the example. Next, it lists source ports and destination ports along with the total count of each individual ports. It also shows total number source IP address counts and total number of packets the received. There are other additional information related to the attack, such as key, pps, bps and start time. Most important information used from that fingerprint for generating the rule are source IP address, source port, destination port and protocol.

4.2 Rule Converter and its Requirements

Usually, at the operational level, the administrator extracts the main information of DDoS attack and write the mitigation rule. The rule which an administrator writes is not as complete as system generated rule. Sometimes attacks are big and administrator can miss source IP addresses while creating the rule. Therefore, the goal is to get the completeness of the rule using some automated tool.

```
{
  "protocol": "UDP",
  "src_ips": [
    "14.153.121.111",
     ......
     ......  ],
  "src_ports": [32773],
  "total_src_ports": 1,
  "dst_ports": [80],
  "total_dst_ports": 1,
  "total_src_ips": 4
  "start_timestamp": 1428877726.951922,
  "key": "54d639d921ceef8060eb804a5672749b",
  "start_time": "2015-04-12 15:28:46",
  "duration_sec": 37.718438148498535,
  "total_packets": 1289,
  "avg_pps": 34.17426763338321,
  "avg_bps": 11980.63923593264,
  "vector_filter": "(['_ws.col.Protocol']=='UDP')&(['srcport']==32773)",
  "multivector_key": "78b770adb71b23ac7ef776feba51648d"
}
```


Figure 4.1: Summary of a PCAP file.

We propose a solution that can automatically generate the mitigation rule using a DDoS fingerprint. Figure 4.2 shows our proposed solution for DDoS mitigation, which is involving three main phases. In previous chapter we focused on phase 1 for collecting enough data for a specific period of time. Then, in previous section we discussed about phase 2, which is DDoS Dissector used for generating the fingerprint. In this section we discuss about phase 3, which is the rule converter. This rule is device specific and minimizes the number of fields from summary of DDoS attacks. Access-control list (ACL) deployment at faster rate will increase the effectiveness of the mitigation system.

Rule converter generates ACL for mitigating the DDoS attack and it is very important phase in blocking the DDoS attack. The rule converter has 2 main requirements for access-control list creation: (1) once an anomaly is detected, 'enough' data should be collected (this was done in phase 1); (2) collected data should be converted to a fingerprint (.JSON file), which shows the summary of a DDoS attack (this was done in phase 2); and (3) these conversions must also consider which fields the hardware/software has. Attack data is collected for a specific time when an anomaly is detected, then we use fingerprinting tool for generating the summary of attacks, which shows characteristics of a DDoS attack. Example of fields that the DDoS summary contains: protocol, source IP (src-ips), source ports (src-ports) and destination ports (dst-ports). A fingerprint can contain more than one .JSON file. Each .JSON file contains a protocol (TCP,UDP and ICMP). The rule converter

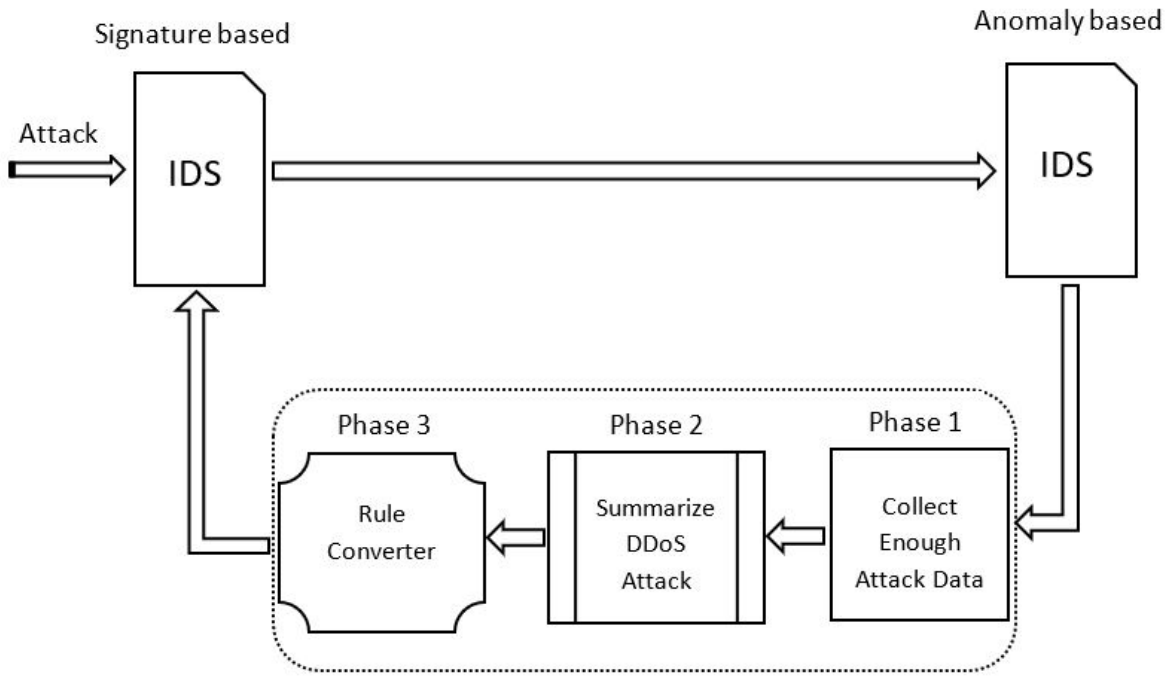


Figure 4.2: Proposed Solution.

must be able to translate the summary of attack to a device specific rule, but these conversions should be in such a way that it would fit with the limitation of software and hardware.

4.3 Rule Generation Process

For generating access-control lists (ACL) we use as input the summary of the attack (collected in the previous step). Although the attack summary may contain many network features, we use only the list of source IP addresses, source ports and destination ports. This is a limitation imposed by the type of device we use to validate our ACL's.

There are two ways for creating an access-control lists (ACL). The first way is individual mapping, which for every source IP address in the attack summary there will be one entry in the ACL. The problem of this approach is that the software/hardware that receives ACL has limited memory and DDoS attacks have a large number of IP addresses. To reduce the number of ACL entries, there is a second way, which is subnet-based method. In this way a range of IP addresses is grouped into a segment with a mask field. The problem of this approach is that this aggregation will likely to include legitimate IP addresses, which are not participating in the attack. Following we describe whether we use one or the other approach.

In our algorithm, we group the IP addresses that are in the same range aim-

ing for the smaller subnet size. The reason for this smaller grouping is to reduce the addition of legitimate IP addresses in the final ACL. For example, the three IP addresses 130.89.14.1, 130.89.14.2, 130.89.14.3 will be grouped into a single /24 subnet 130.89.14.0. We consider three rounds of interaction. The first, tries to group IP addresses in subnets between /24 and /31. For the remaining IP addresses that were not grouped, we try to group in subnets between /16 and /23. Finally, for the remaining IP addresses, we consider those for an individual mapping approach (/32 subnet).

The attack summary contains a single value of IP protocol (e.g., ICMP, TCP, UDP). Therefore, we add this information to the ACL. The attack summary also contains either only one source port or only one destination port. Therefore, we add one of this two characteristics to the ACL. Usually, this port defines the type of the attack (although this is not important to the operator). For example, if the source port is 389 we consider the attack is based of LDAP. If the destination port is 389 we assume that the target system is the LDAP.

4.4 Experimental Setup

This section describes Experimental Setup in order to mitigate the attack. A live setup was created at NBIP(Nationale beheersorganisatie internet providers) for mitigating various DDoS attacks. This live setup is the blueprint of proposed model with some additional devices in place.

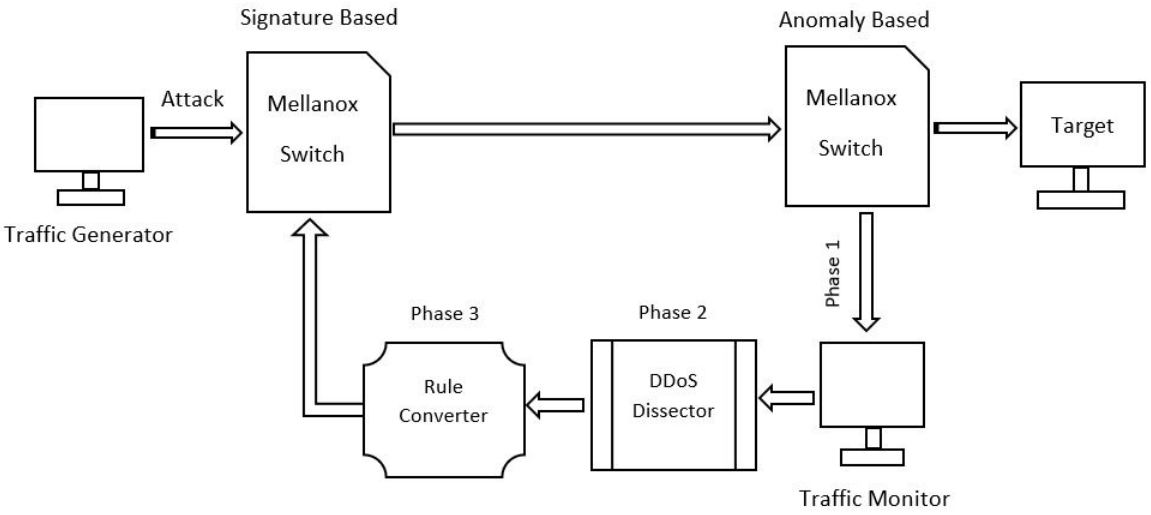


Figure 4.3: Experimental Setup

As shown in figure 4.3, Experimental setup consists of following things. The technology used for signature-based and anomaly-based is Mellanox SN2010 switch

which uses Onyx software and it is accessed via command line. There are two linux based machines which acts as traffic generator and traffic monitors. Traffic generator and traffic monitors are running operating system 'Ubuntu 18.04.1'with Intel(R) Xeon CPU E3 processor and 8 GB of RAM. These machines are having around 160 GB disk space and can store limited files. Traffic generator will initiate the attack to a specific target and Traffic monitors will monitor the attack and capture the traffic for generating fingerprint using DDoS Dissector. The target is the real server where application is hosted. We use DDoS Dissector tool to generates the fingerprint of an attack which is observed from anomaly-based detection. This tool is assumed to be functioning as expected and not it is not my actual contribution.

For this analysis we considered composition of the dataset made publicly available by [5] [6]. These datasets consist of DDoS attacks, which were purchased from websites that offer DDoS as a Service. Every PCAP's destination IP was rewritten to production IP using 'tcprewrite'. Tcprewrite is a tool, that can rewrite the packets stored in a PCAP format [39]. Each attack was replayed with having closed boundary with 'tcpreplay'. Once a PCAP file is rewritten, it can be replayed back with 'tcpreplay'tool [40]. Traffic generator uses tcpreplay tool for re-initiating the attack and traffic monitors captures the traffic for generating PCAP. This PCAP will be used to generate fingerprint using DDoS Dissector. Finally, rule converter will then take fingerprint as an input and generate individual mapping and subnet-based access list. These access lists are then deployed on signature-based Mellanox device. Each access list mitigation behavior was observed through 'tcpdump'command and then verifying logs at Mellanox device too. Main attributes considered during the access list creations were protocols, source IP addresses, destination IP addresses, source port and destination ports. These are the only fields Mellanox can support for ACL creation.

4.4.1 Limitations of Mellanox

There are many restrictions in Mellanox device which does not support for mitigating larger DDoS attacks. Some of the limitations are as listed below:

- BGP Flowspec is not supported directly on Mellanox SN2010. BGP Flowspec is having a granular approach for mitigating DDoS attacks. It matches a particular flow and effectively install dynamic actions which can drop traffic or place it to a different forwarding instance for further analysis.
- Mellanox Access-list works only for ingress direction, applying rule in egress won't filter the traffic. Rules can be applied only in inbound direction for filtering the traffic.

- Mellanox supports maximum of 9000 ACL entries. Since, there is a tight restriction, it is difficult to add large set of ACL entries.

- Mellanox allows only one access-list to bind its interface. When there are different categories of access-control lists, we cannot group all those access-control lists to an interface. We can only bind one group of access-control lists to an interface. If this group is changed, then we need to bind the new group to the interface.

- Mellanox does not support Object-Group feature. Often in production network the number of access-control list count is very high and it's difficult to manage those access-control lists. The Object-Group feature helps to classify IP (individual IP and subnets) and services into groups and use those groups in the access-control lists. This eventually reduce the number of access-control list entries.

- Mellanox does not support packet tracer and traffic injecting options. For validating the access-control lists traffic needs to enter the Mellanox device, but there is no option for injecting the traffic from Mellanox. Packet tracer feature is also missing, which initiate a virtual packet.

- Mellanox SN2010 does not maintain connection table, which is a stateless device. Since Mellanox is not support for stateful traffic an explicit access-control list needs to be defined for inbound and outbound traffic. This is very helpful for TCP traffic for initiating three-way handshake process.

- Mellanox does not support for combination of ports in an access-control list, i.e It wont accepts more than one individual port, but it accepts port range. So, if there are multiple ports with different ranges, the separate access-control list we need to define.

- Traffic logs are visible only for few seconds, which then clears from the buffer. There is no option for storing historical logs which helps to determine the traffic flows.

- Mellanox does not support ICMP codes and DNS query options while adding an ACL. Device is strictly controlled with limited fields.

Since DDoS mitigation system requires different attributes options while mitigating an attack and Mellanox comes with very tight restrictions, considering this drawbacks available options are explored on Mellanox device and it was used in mitigating the DDoS attack.

4.5 Results

To evaluate the rule converter, we consider the reduction of source IP addresses as a metric. Since, there is a limit in the hardware for ACL entries, its important to have reduction in number of source IP addresses from DDoS fingerprints.

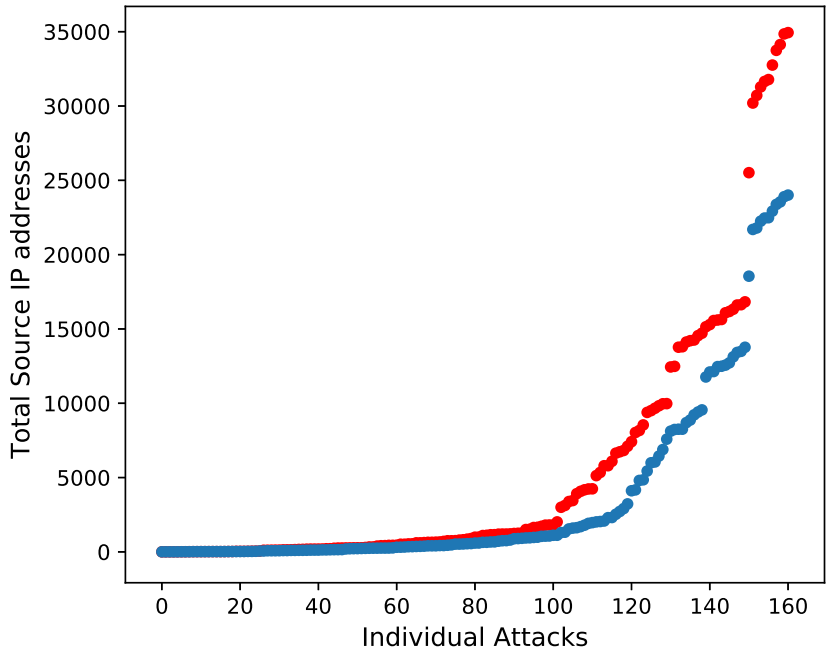


Figure 4.4: Distribution of Source IP Addresses per Individual attack for fingerprint and rule converter

Figure 4.4 shows the distribution of source IP addresses from DDoS fingerprints and after rule converter. The X-axis indicates the individual DDoS attacks, while Y-axis shows the total source IP addresses involved in each DDoS attack and rule converter. Fingerprint contains all source IP addresses and rule converter shows reduction of those source IP address. The rule converter aims to group the similar IP segment for reducing count of individual IP address. It is observed that, at the starting point source IP addresses count is overlapping among fingerprint and rule converter. The reason of overlap is that, the individual IP address which are present in the DDoS attacks are not contiguous and these IP address cannot be group under a segment.

In the figure 4.4 if the count of source IP is lower than 950 source IP addresses, then we basically observer the overlap. In our dataset, we have nearly 100 such attacks having overlapping and surprisingly we discovered some of the attack source IP addresses cannot be reduced because they are widely distributed. For the attacks that have less than 2000 IP address, the reduction is minimum (1500 to 1700). However, the limitation of hardware is still suitable and ACL's without reduction is

acceptable, because of the device capacity (9000 ACL entries).

The upper right of the figure 4.4 shows large difference among fingerprint and rule converter source IP addresses. For example, in attack number 160, the number of source IP addresses in fingerprint is 34852 and rule converter shows 23897 source IP addresses. The graph also highlights that, if there is a large number of source IP address present in a DDoS attack, then the difference among fingerprints and rule converter is bigger. In another words, rule converter minimize the number of source IP address from the actual count of fingerprint source IP addresses. However, attacks numbered from 130 to 160, which have more than 9000 source IP address, would not be suitable for deploying the ACL into the device (9000 ACL entries is the limit), even after rule converter reduces this source IP addresses by subnetting.

The reduction is actually required from attack number 130 onwards. For all these attacks, we have a problem of reduction, which is higher than 9000. Considering the limit of the hardware (9000 ACL entries), the attacks that have reduction higher than 9000 and this reduction is still not sufficient to fit in hardware. This is not even considering the damage after adding legitimate IP address during ACL creation. One way to overcome this problem is changing the algorithm to do more aggregation, until it reaches the limit of the hardware. However, the implication of this change, possibly include much more legitimate traffic.

4.5.1 Mellanox Access Control List

Each flow is uniquely represented by the following five attributes (protocol, source IP address, destination IP address, source port and destination port). General syntax which is considered for rule creation is:

Sequence-number {seq-number} Action {permit | deny} Protocol {tcp | udp | icmp} Source-IP {[mask] | [any]} Destination-IP {[mask] | [any]} Source-Port {0-65535} Destination-Port {0-65535} Log {optional}

Below are the list of ACL entries, which can be used for blocking the malicious traffic. These ACL is in-line with Mellanox device syntax.

Example of 'Greedy'Rule: seq-number 47 deny udp any any eq-destination 1900 log

Example of Single Host: seq-number 47 deny udp 50.198.204.14 mask 255.255 .255.255 100.100.100.100 mask 255.255.255.255 eq-source 19 log

Example of Subnet-Based Rule: seq-number 16 deny udp 2.229.80.160 mask 255.255.0.0 100.100.100.100 mask 255.255.255.255 eq-source 19 log

Above examples shows three ACL types and each of the access list begins with a unique sequence number. Each sequence number represents an entry in access list table with any overlaps. Two or more access list with similar IP address is possible, but access list with same sequence number is not acceptable. Access list also represent an important action field, whether to allow or deny the traffic. Next, each access list also needs protocol field in order to classify type of traffic followed by source and destination IP address. In access list we can include both source port and destination port. Finally, every access list entry can be logged by giving 'log'keyword at the end of the access list.

4.5.2 Mellanox Access Control List Configuration

We have three important steps to configure an access list and fourth step is used to verify if the access list is defined properly as per need.

Step 1: Enter config mode-
Mellanox > enable
Mellanox# configure terminal

Step 2: Appropriate access list creation-
Mellanox(config)# ipv4 access-list NAME
(config ipv4 access-list NAME)# seq-number 47 deny udp 50.198.204.140 mask 255.255.255.255 100.100.100.100 mask 255.255.255.255 eq-source 19 log

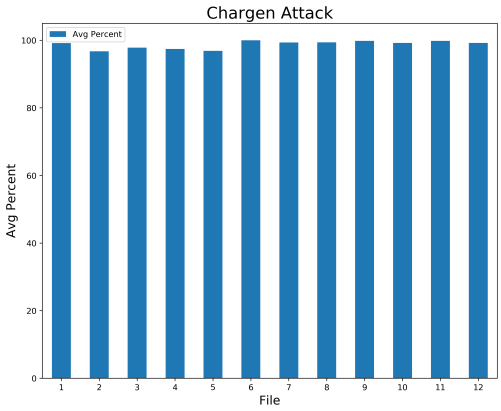
Step 3: Bind the created access list to an interface-
Mellanox(config)# interface ethernet 1/5
Mellanox (config interface ethernet 1/5)# ipv4 port access-group NAME

Step 4: Verify access list creation-
Mellanox (config ipv4 access-list NAME)# show access-lists

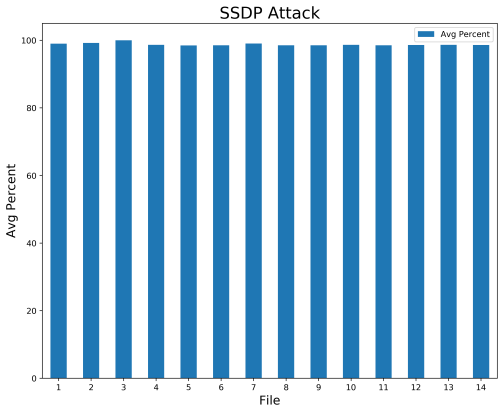
In step 1, we have to enter into global configuration mode to make access list implementation. In step 2, we have to define an access list name and followed by access list parameters. In step 3, we should bind the access list to an interface of the device for proper filtering. Attaching access list to interface which is not participating in the traffic flow will not have any impact. Finally, we can verify an access list entry with the help of 'show access list'command, which will list out all access list from the table.

4.6 Impact of ACL

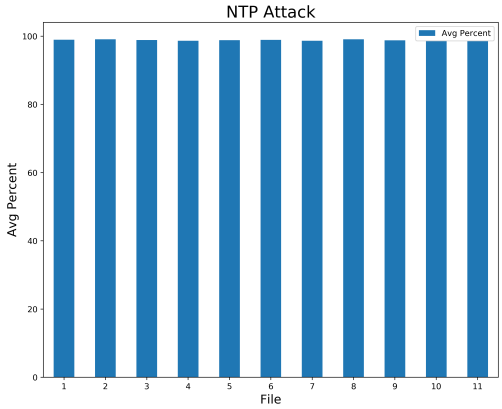
This section describes the consequence of ACL's, which is created using individual mapping and subnet-based approach.



(a) Chargen Attack



(b) SSDP Attack



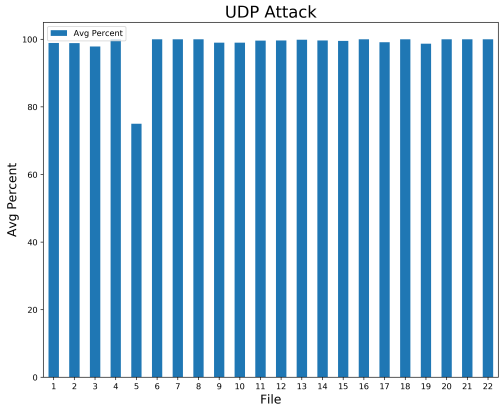
(c) NTP Attack

Figure 4.5: UDP Amplification Attacks

There is addition of legitimate IP addresses during ACL creation, that can block the IP address which are not participating in the attack. Figure 4.5 illustrates UDP amplification attacks such as Chargen, SSDP and NTP attacks. A key observation is almost every file which was simulated showed theoretical false positive (worst case) above 98%, which means a large set of non-malicious IP address are blocked potentially. At an average theoretical false positive value for Chargen attack is 98.72%, SSDP attack is 98.77% and NTP attack is 98.83%.

Figure 4.6 shows the average theoretical false positive for an UDP based attack. A key observation almost every file which was simulated showed theoretical false

positive around 98%, except one file which is file number 4 having 75%. At an average theoretical false positive value for UDP attack is 98.39%.



(a) UDP Attack

Figure 4.6: UDP Attack

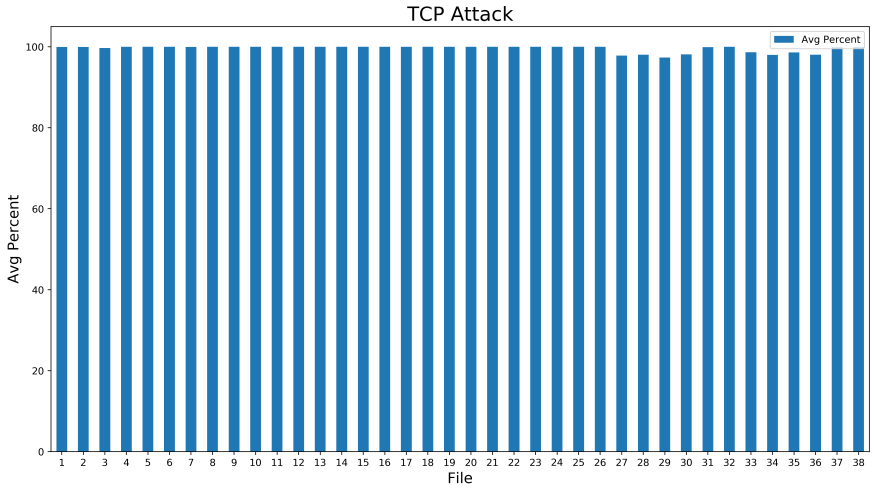


Figure 4.7: TCP Attack

Figure 4.7 shows the average theoretical false positive for an TCP based attack. A key observation almost every file which was simulated showed theoretical false positive around 99%. At an average theoretical false positive value for TCP attack is 99.56%.

Figure 4.8 shows the average theoretical false positive for an ICMP based attack. A key observation almost every file which was simulated showed theoretical false positive around 99% At an average theoretical false positive value for ICMP attack is 99.39%.

The analysis shows that all attacks are have high percentage of theoretical false positive rate. Although, number of rules are reduced in one side, number of additional IP's are increasing on the other side.

There are many additional IP address considered while performing subnet-based access list creation. These additional IP addresses may have two different scenarios. At first, these additional IP addresses could be part of normal traffic. Second, these additional IP addresses may not be participating or involving in the traffic for that moment.

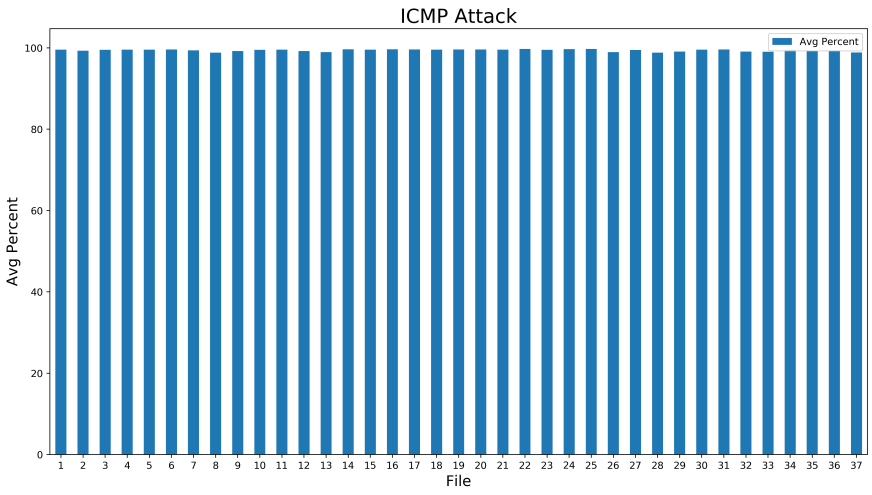


Figure 4.8: ICMP Attack

A table is created for every subnet-based access list and ‘theoretical’false positive calculation is performed automatically for every attack. Using these ‘theoretical’false positive values, a graph is plotted for every type of attack. Table 4.1 mainly shows IP address, mask and total IP address belonging to a desired subnet, upon these field the difference and average is calculated.

4.6.1 Analysis of various type of Rules on Mellanox

Access list creation is a important step for mitigating the DDoS attack and various types of rule creation method can be used. Each rule type have positive and negative characteristics and in table 4.2 four important factors are considered for each type of rules and explained as follows:

Accuracy: The greedy rule accuracy is very less when compare to other two types of rules. The main cause for this is the rule behavior, which allows entire IP range to access the network. On the other end, single host is considering every IP

Table 4.1: Rule Table

IP	Mask	Count	Total IP	Difference	Diffpercent	Average
58.211.15.51	255.255.0.0	2	65536	65534	99.996	97.80
60.195.62.250	255.255.0.0	3	65536	65533	99.995	97.80
60.217.226.148	255.255.224.0	2	8192	8190	99.975	97.80
112.216.44.68	255.255.0.0	3	65536	65533	99.995	97.80
115.28.142.183	255.255.128.0	4	32768	32764	99.987	97.80
115.29.48.82	255.255.0.0	2	65536	65534	99.996	97.80
117.239.22.97	255.255.0.0	2	65536	65534	99.996	97.80
124.206.20.248	255.255.255.240	2	16	14	87.5	97.80
190.85.130.90	255.255.128.0	2	32768	32766	99.993	97.80
211.53.135.150	255.255.128.0	2	32768	32766	99.993	97.80
211.168.41.232	255.255.255.224	2	32	30	93.75	97.80
220.165.80.250	255.255.255.248	2	8	6	75	97.80
220.169.61.13	255.255.0.0	2	65536	65534	99.996	97.80
220.172.191.50	255.255.128.0	2	32768	32766	99.993	97.80

Table 4.2: Rule Type

Types of Rules	Accuracy	Complexity	Impact	Consequence
'Greedy'Rule	- -	+ +	- -	+ +
Single Host	+ +	- -	+ +	- -
Subnet Based	+ -	- -	- -	+ -

which is participating in the attack and thus create a one to one rule, without addition of non-malicious hosts. Finally, subnet-based rule reduces the accuracy by addition of non-malicious hosts into the rule.

Complexity: Complexity here is in terms of rule creation and how easy it is to create. One of the easiest rule creations method is the greedy rule, it produces single access list. On the other end, single host rule method generates multiple access list according to number of malicious IP address and subnet-based rule method also creates multiple access list, but with slightly less rules than single host method.

Impact: The greedy rule method adds many non-malicious hosts in the access list thus impact highly. In single host method only malicious hosts considered for creation of access list and there is no malicious hosts added in the access list so that impact is less. In subnet-based rule method there will be many non-malicious hosts for creation of access list, thus impact is more when non- malicious hosts looses the access for the network.

Consequence: In greedy rule method the rule count is significantly reduced compared to single host and subnet-based method. In single host method more number

of rules are created, it creates a separate rule for every IP address. In subnet-based method IP addressed are group to a particular subnet and then create an access list which is generally lesser than single host rule method.

All these three rules have positive and negative characteristics, but choosing appropriate rule type depends on the situation and network behavior. If there is large memory that supports single host method access list method then we can easily choose. On the other end, if there is memory issues on the device and rules needs to be optimized, then we can choose subnet based rule method. We can also choose greedy rule method to block specific services which are not generally routable over internet. Overall, according to need we can choose one of these three methods and we can mitigate the attack.

4.6.2 Lessons Learned Using Static Rules

One way of mitigating the DDoS attacks, is through capturing the attack data and then converting that data into a fingerprint. Then, this fingerprint is used for ACL creation, which is then deployed into mitigation device. Another way is to block the DDoS attack of specific characteristics through static rules. These static rules can be pre-configured on the mitigation device, even before the attack starts.

Case 1: If there are incoming packets with source port below 1024 and destination port 80, then such traffic could be blocked. Since, ports from 0-1023 are well-known ports, used for specific purpose. Any traffic with source port below 1024 targeting to a destination server on port 80 should be blocked.

Mellanox# seq-number 11 deny tcp any 1.2.3.4 mask 255.255.255.255 eq-source range 0-1023 eq-destination 80 log.

Case 2: This case is applicable, when an attacker is trying to access destination server on port 22. Since, administrator wants to restrict the direct access of the device, he can pre-configure the rule for blocking public access. Administrator can configure an ACL to block the traffic:

Mellanox# seq-number 12 deny tcp any 1.2.3.4 mask 255.255.255.255 eq-destination 22 log.

Case 3: This case is applicable, when there is invalid combination of TCP flags. TCP flags are used to indicate a specific connection state and most commonly used TCP flags are SYN, ACK and FIN. TCP connection is established with SYN, SYN ACK, and ACK flags. There are also other flags such as URG PSH and RST, which is used for specific purpose. Packets with no flags are considered as invalid and indicates some malicious activity in the network. Also, If packet contains SYN and

FIN then it's an invalid combination, we can block such packets with below rule:

Mellanox# seq-number 13 deny tcp 211.144.72.153 mask 255.255.255.255 1.2.3.4 mask 255.255.255.255 eq-destination 80 syn 1 fin 1 log.

Since, these static rules are considered for known behaviors, blocking them at first instance will minimize the resource consumption and leads to faster mitigation. These pre-configured rules for suspicious traffic can reduce the number of ACL entries and with single ACL we can block the attack.

4.7 Concluding Remarks

The goal of this chapter was to answer RQ3 (How to convert summary of DDoS attack into mitigation rules automatically?). To answer RQ3, we spilt this chapter into six sections. In section 4.1, we discussed about summary of an DDoS attack using DDoS Dissector. Before generating ACL, we have one intermediate phase to use collected data for generating summary of DDoS attack. This summary of attack characteristics is called DDoS fingerprint. We have already collected enough attack data, but this attack data involves repeated source IP address and unorganized information. Therefore, we use a tool that generates the fingerprint for collected data. The fingerprint of a DDoS attack gives more information about DDoS attack, which includes protocol, total source IP address, source ports, destination ports and number of packets etc.

Then, in section 4.2 we discussed about rule converter and its requirements for generating ACL's. Access-control list deployment at faster rate will increase the effectiveness of the mitigation system. The rule converter automatically creates the ACL's then deploy in signature-based device. The main requirements for rule converter is: collecting 'enough' data after an anomaly is detected (phase 1), then convert collected data to a fingerprint using DDoS Dissector (phase 2), finally rule converter uses fingerprint as a input and generates ACL's for mitigating DDoS attack (phase 3).

Then, in section 4.3 we discussed rule generation process, which shows the ACL generation procedure. We can create ACL with Individual mapping i.e every source IP address will be an ACL entry. However, this Individual mapping will contribute to many ACL's and memory consumption will be high, therefore we propose an algorithm to generate an access-control list with Individual mapping and subnet-based approach. This can reduce the count of number of ACL entries. We group the IP addresses, which are in similar range for obtaining a smaller subnet mask. The smaller subnet mask will have less legitimate IP addresses in the final ACL. The rule converter tool first tries to group IP addresses in subnets between /24 and /31. Next,

it groups remaining IP addresses between /16 and /23. Finally, individual mapping approach (/32 subnet) is considered for remaining IP addresses.

Then, in section 4.4 we explained the experimental setup used in mitigation of DDoS attacks. We evaluated our solution in a live environment at NBIP(Nationale beheersorganisatie internet providers). The experimental setup mainly consists of Mellanox switches, traffic generator, traffic monitors, DDoS Dissector tool and rule converter tool. The traffic generator was used to initiate the attack traffic with tcpreplay and traffic monitors captures traffic through tcpdump utility. DDoS Dissector tool was used to obtain fingerprint of a DDoS attack and rule converter tool was used to generate the ACL's. These ACL's were deployed on signature-based Mellanox device.

Then, in section 4.5 we discussed about the results. The reduction of source IP addresses was considered as a metric and evaluated the tool performance. A key observation during the analysis was the overlap of source IP addresses among fingerprint and rule converter. Since, source IP address are not contiguous and few IP addresses were not group under a segment. We also saw surprising result during the analysis, for few attacks the source IP addresses could not be reduced because they were widely distributed. There were attacks whose total source IP address count was less than 9000 and limitation of hardware was still suitable to accept all 9000 IP addresses without reduction. However, there was large difference among fingerprint and rule converter source IP addresses for attacks having more than 15000 source IP address. There were attacks that reduction was still not sufficient to fit in hardware capacity. This reduction was considering legitimate IP addresses for ACL creation.

Then, in section 4.6 we highlighted the impact of the ACL which was created using rule converter. During ACL creation, rule converter is considering IP addresses which are not participating in the attack. This addition of legitimate IP addresses will possibly have impact, if normal traffic passes during attack period. This impact of blocking the normal traffic might affect many applications in production network. This drawback needs to be fixed and we will be discussed in future work section, which is in chapter 5. Finally, we were able to generate ACL's automatically for blocking the attack, but addition of legitimate IP addresses makes tool more unreliable and more improvements is required in the area of not blocking legitimate IP addresses.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

This thesis aimed to addressed knowledge transfer problem from anomaly-based detection to signature-based detection that can improve the system efficiency. A logical connection is established from anomaly-based to signature-based detection that helps to mitigate the attack faster. It creates the knowledge from anomaly-based detection and add value to signature-based detection. The goal of this thesis was to create access-control list using summary of the attack and these rules was deployed on Mellanox switch.

To meet our goal, we defined three research questions: In RQ1 (What are the existing solutions that combine signature-base and anomaly-base to detect DDoS attack?) we performed literature review on DDoS attacks, Intrusion Detection systems and its related works. This literature review helped in having strong background and to discover the novelty of my work. At first, we explained about DDoS attacks types and its architecture followed by Intrusion Detection systems. DDoS attack is accomplished by flooding the target with enormous Internet traffic that result to huge damage. There are two categories of DDoS attacks: bandwidth depletion and resource depletion. Bandwidth depletion generates massive traffic from botnets which creates congestion and consumes more bandwidth and resource depletion attack consumes server resources such as firewalls and load balancers, which disrupt the service.

There are two main detection method for detecting DDoS attacks: signature-based detection and anomaly-based detection. Signature-based detection is more specific and effective in detecting known attacks. Anomaly-based detection is more generic and trained to detect new attacks. Both signature-based and anomaly-based detection have advantages and disadvantages. The main advantage of signature-based detection is detecting the known attacks faster and the disadvantages is not detecting unknown attacks. The main advantage of anomaly-based detection is de-

tecting the unknown attacks and the disadvantages is, it generates more false positive alerts. There is another method that combines signature-based and anomaly-based detection known as hybrid-based detection. This system is capable in detecting known and unknown attacks. There are various works that showed effective mitigation of DDoS attacks using hybrid-based detection. The background of DDoS and Intrusion Detection systems helped in getting better understanding.

Next, in RQ2 (How long the traffic needs to be collected for generating the fingerprint of an attack once the anomaly is been detected?) the main goal is to discover the time required to capture attack traffic from anomaly-based detection for fingerprint generation. For this analysis we used nearly 200 attack traces and evaluated separately. Measurement of 'enough' time is very important in collect sufficient amount of traffic to generate the fingerprint. This 'enough' time provides the maximum source IP address in the attack data. This time also contributes in mitigating the attack as early and helps in less data storage while capturing. We discovered that 300 seconds needed to capture all source IP addresses involved in DDoS attacks based on our dataset. we capture this sample attack data in the PCAP format and pass it to DDoS Dissector tool, which generates summary of the attack. This summary of attack known as fingerprint will be used in RQ3 for creating access control list.

Finally, in RQ3 (How to convert summary of DDoS attack into mitigation rules automatically?) the goal is to create access-control list (ACL) which can mitigate DDoS attacks on signature-based device. We assessed the complete execution flow by including anomaly detection, traffic collection, fingerprint conversion, and signature-based mitigation. During phase 1 and 2 we collected enough traffic and fingerprint was generated. Then, fingerprint was used as an input in rule converter and ACL's were generated. These ACL's was then deployed on signature-based Mellanox device to assess the DDoS mitigation. During analysis we observed an overlap of source IP addresses among fingerprint and rule converter. We also discovered that source IP address are not contiguous and few attacks the source IP addresses could not be reduced because they were widely distributed. For some attacks there was large difference among fingerprint and rule converter source IP addresses. Also, for few attacks the reduction did not help in deploying the ACL to fit in hardware capacity. On one hand, individual mapping approach creates specific ACL for every IP address that is participating in the attack, which contribute to large number of rules. On the other hand, subnet-based ACL aims to reduce the number of rules while adding more legitimate IP addresses. Although, the attack was mitigated with ACL's on signature-based Mellanox device, it included legitimate IP addresses during ACL creation.

5.2 Future Work

The rule converter attempts to generates ACL automatically, however it includes many legitimate IP address that is not participating in the attack. Also, the current algorithm needs to be enhanced for producing ACL entries matching device capacity. These gaps need to be addressed for making the tool more reliable and administrator can make use of the tool.

Since, rule converter is making use of subnet-based approach, there are high probabilities of adding more IP address which are not involved in attack. One way to avoid this problem is by analyzing the impact of the ACL before deployment. If there is no overlap of legitimate IP addresses, then the administrator can choose to deploy the rule. An administrator should test against the normal traffic and determine the impact, if it is actually blocking the legitimate traffic. Another point is, an administrator can input the limit specifying maximum number of rules of that device can support. Then, the tool should attempt to recursively check and generate ACL's for specified number. It can group similar set of IP address and reduce the subnet mask for achieving this goal.

Bibliography

[1] 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*, vol. 15, no. 4, pp. 2046–2069, Fourth 2013.

[2] “Netscout,” https://www.netscout.com/blog/asert/netscout-arbor-confirms-17-tbps-ddos-attack-terabit-attack-era/, accessed: 2019-03-20.

[3] K. Kumar, “Intrusion detection and prevention system in enhancing security of cloud environment,” vol. 6, pp. 2278–1323, 08 2017.

[4] M. Gupta, “Hybrid intrusion detection system: Technology and development,” 2015.

[5] V. Bukac, V. Stavova, L. Nemec, Z. Riha, and V. Matyas, “Service in denial – clouds going with the winds,” in *Network and System Security*, M. Qiu, S. Xu, M. Yung, and H. Zhang, Eds. Cham: Springer International Publishing, 2015, pp. 130–143.

[6] J. J. Santanna, R. van Rijswijk-Deij, R. Hofstede, A. Sperotto, M. Wierbosch, L. Z. Granville, and A. Pras, “Booters âĂŤ an analysis of ddos-as-a-service attacks,” in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, May 2015, pp. 243–251.

[7] “Ddos dissector,” https://github.com/ddos-clearing-house/ddos_dissector/, accessed: 2019-09-21.

[8] “Corero,” https://www.corero.com/blog/914-automated-ddos-mitigation-is-essential-.html, accessed: 2019-03-20.

[9] “Arbor,” https://www.netscout.com/blog/consequences-ddos-attacks-are-rising, accessed: 2019-03-20.

[10] S. Rajan and C. Vivek, “Systematic review of sar reduction techniques for minimizing mobile phone radiation,” *JOURNAL OF ADVANCES IN*

CHEMISTRY, vol. 12, no. 9, pp. 4341–4348, Nov. 2016. [Online]. Available: https://rajpub.com/index.php/jac/article/view/4102jac

[11] S. Arukonda and S. Sinha, "The innocent perpetrators: Reflectors and reflection attacks," 2015.

[12] K. S. V. Lovepreet Kaur Somal, "Classification of distributed denial of service attacks architecture, taxonomy and tools," *International Journal of Advanced Research in Computer Science and Technology (IJARCST)*, vol. 2, 2014.

[13] V. P. Keyur Chauhan, "Distributed denial of service(ddos) attack techniques and prevention on cloud environment," *International Journal of Innovations Advancement in Computer Science (IJIACS)*, vol. 4, 2015.

[14] "Radware," https://security.radware.com/ddos-knowledge-center/ddospedia/amplification-attack/, accessed: 2019-07-05.

[15] "Cloudflare," https://www.cloudflare.com/learning/ddos/ddos-attack-tools/how-to-ddos/CommonDDoSAttacks, accessed: 2019-07-05.

[16] "imperva," https://www.imperva.com/learn/application-security/ddos-attacks/, accessed: 2019-07-05.

[17] "securelist," https://securelist.com/ddos-report-in-q3-2018/88617/, accessed: 2019-07-05.

[18] B. Shanmugam and N. B. Idris, "Improved intrusion detection system using fuzzy logic for detecting anamoly and misuse type of attacks," in *2009 International Conference of Soft Computing and Pattern Recognition*, Dec 2009, pp. 212–217.

[19] J. Han, M. Kamber, and J. Pei, "13 - data mining trends and research frontiers," in *Data Mining (Third Edition)*, third edition ed., ser. The Morgan Kaufmann Series in Data Management Systems, J. Han, M. Kamber, and J. Pei, Eds. Boston: Morgan Kaufmann, 2012, pp. 585 – 631. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780123814791000137

[20] V. Paxson, "Bro: a system for detecting network intruders in real-time," *Computer networks*, vol. 31, no. 23-24, pp. 2435–2463, 1999.

[21] M. Roesch, "Snort - lightweight intrusion detection for networks," in *Proceedings of the 13th USENIX Conference on System Administration*, ser. LISA '99. Berkeley, CA, USA: USENIX Association, 1999, pp. 229–238. [Online]. Available: http://dl.acm.org/citation.cfm?id=1039834.1039864

[22] M. Khamruddin and C. Rupa, “A rule based ddos detection and mitigation technique,” in *2012 Nirma University International Conference on Engineering (NUiCONE)*, Dec 2012, pp. 1–5.

[23] B. Agarwal and N. Mittal, “Hybrid approach for detection of anomaly network traffic using data mining techniques,” *Procedia Technology*, vol. 6, pp. 996 – 1003, 2012, 2nd International Conference on Communication, Computing amp; Security [ICCCS-2012]. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2212017312006664

[24] J. Yang, X. Chen, X. Xiang, and J. Wan, “Hids-dt: An effective hybrid intrusion detection system based on decision tree,” in *2010 International Conference on Communications and Mobile Computing*, vol. 1, April 2010, pp. 70–75.

[25] I. Brahmi, S. Ben Yahia, and P. Poncelet, “Mad-ids: Novel intrusion detection system using mobile agents and data mining approaches,” in *Intelligence and Security Informatics*, vol. 6122, 09 2010.

[26] C. Buragohain, M. Jyoti, S. Singh, and D. K, “Anomaly based ddos attack detection,” *International Journal of Computer Applications*, vol. 123, pp. 35–40, 08 2015.

[27] “Caida,” https://www.caida.org/data/, accessed: 2019-03-14.

[28] J. B. Cabrera, L. Lewis, X. Qin, W. Lee, R. K. Prasanth, B. Ravichandran, and R. K. Mehra, “Proactive detection of distributed denial of service attacks using mib traffic variables-a feasibility study,” in *Integrated network management proceedings, 2001 IEEE/IFIP international symposium on*. IEEE, 2001, pp. 609–622.

[29] K. Hwang, P. Dave, and S. Tanachaiwiwat, “Netshield: Protocol anomaly detection with datamining against ddos attacks,” *6th International Symposium, Pittsburgh*, 2003.

[30] Tanachaiwiwat and K. Hwang, “Adaptive packet filtering against ddos attacks,” *Dept. of EE-Systems, University of Southern California (USC)*, 2003.

[31] C. Thomas, V. Sharma, and N. Balakrishnan, “Usefulness of darpa dataset for intrusion detection system evaluation,” 03 2008.

[32] Y. Ding and M. X. and, “Research and implementation on snort-based hybrid intrusion detection system,” in *2009 International Conference on Machine Learning and Cybernetics*, vol. 3, July 2009, pp. 1414–1418.

[33] V. Katkar and S. Bhirud, “Novel dos/ddos attack detection and signature generation,” *International Journal of Computer Applications*, vol. 47, pp. 18–24, 06 2012.

[34] G. Nadiammai and M. Hemalatha, “Effective approach toward intrusion detection system using data mining techniques,” *Egyptian Informatics Journal*, vol. 15, no. 1, pp. 37 – 50, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1110866513000418

[35] O. Cepheli, S. Buyukcorak, and G. Karabulut Kurt, “Hybrid intrusion detection system for ddos attacks,” *Journal of Electrical and Computer Engineering*, vol. 2016, pp. 1–8, 01 2016.

[36] P. Bhakti and P. Kalvadekar, “Hybrid intrusion detection system using anomalous internet episodes rules with weighted signature generation,” 2015.

[37] A. Tesfahun and L. Bhaskari, “Effective hybrid intrusion detection system: A layered approach,” *International Journal of Computer Network and Information Security*, vol. 7, pp. 35–41, 02 2015.

[38] 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*, July 2009, pp. 1–6.

[39] “Tcprewrite,” https://linux.die.net/man/1/tcprewrite/, accessed: 2019-09-21.

[40] “Tcpreplay,” https://tcpreplay.appneta.com/wiki/overview.html/, accessed: 2019-09-21.