DDoS SYN Flooding; Mitigation and Prevention

Jamaluddin. M, Touqir Anwar. M, K. Saira, M.Y. Wani

Abstract—the purpose of both Denial of Service and Distributed Denial of Service attack is to make the network resources unavailable to the users. A typical DDoS attack is an attempt to disrupt the access of a legitimate user. SYN flooding is one of the most basic DDoS attacks. In a typical SYN flooding attack, an attacker floods the network with SYN packets. The attackers exploit the vulnerabilities of a large number of computers and set up their own army called Botnets. Once the attackers manage to set up a widespread Botnet, they can easily initiate the attack in coordinated fashion. DDoS attacks are statistically considered to be one of the leading threats to the internet. A common way to launch a DDoS attack is to send malicious traffic on the victim's computer. There are various different techniques in practice to defend against the DDoS attacks but the fact remains that these attacks still remain one of the most elusive security attacks. The main reason being that the attacking machines are very large in number and use many different tactics. Intrusion detection systems are aimed at identifying and anticipating the DDoS attacks in advance. However, in order to develop such an effective and comprehensive solution, a thorough understanding of the mechanism is needed.

Index Terms—DDoS Attack, SYN flooding attack, UDP flooding, botnet, zombies, defense architecture, mitigation.

1 INTRODUCTION

Distributed Denial of Service (DDoS) attack is a sophisticated form of the traditional Denial of Service (DoS) attack. Both these attacks are aimed to render the resources unavailable to the users [1]. The network performance is measured along the following three metrics:

- Integrity
- Confidentiality
- Availability

DDoS and DoS attacks are aimed at the availability aspect of the network. The attacker tries to, either make a server busy in processing bogus requests, making the resources unavailable to legitimate clients or the attacker floods the network with such huge amount of bogus network traffic that the legitimate users feel that the network bandwidth is used up. A traditional DoS attack is carried out by a single attacker whereas in DDoS, numerous agents (zombies) carry out the attack.

Researchers have developed several detection and prevention methods for DDoS and DoS, but the fact remains that there can be no universal solution for protection against this attack.

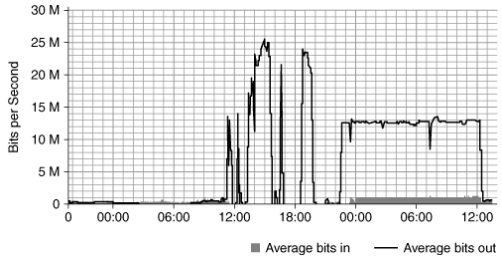

Figure 1: A full scale DDoS attack from victim's perspective

The reason being that the attackers never rely on a single mode of attack nor do they capitalize on a single particular vulnerability [2]. As a result, DoS attacks still remain one of the major threats to networks all around the world. Detection and mitigation techniques against DDoS attacks can be placed at three main location. These are:

1. At locations closer to the source of attack
2. At locations closer to the victim
3. Throughout the intermediate network

Closer to the source of attack approach of detection and mitigation is very complex due to the fact that it can cause a lot of collateral damage, as a lot of hosts or nodes in the network can be targeted which are not actually the source of the attack.

Previously, the internet was so designed that the main focus was the best effort forwarding of any sort of data packet in the least possible time. The early designers of internet were not concerned with the fact the network packets might contain malicious data. This initial internet architecture resulted in pathways that were not regulated at all, which were used by the attackers later on. The DoS attacks made use of this fact to launch attacks by making the Web services unavailable to the users. The intention behind this type of attack was to deny the users of their requested resources. It is now possible to counter this traditional DoS attack and even identify and shut down the attacker. But, with the growth and development of internet, these attacks have also grown in sophistication. The growth

- *Mir Jamal uddin is currently pursuing MS degree program in Computer Science from University of Lahore, Pakistan, E-mail: mir.jamal@uolisb.edu.pk*
- *Malik Touqir Anwar is currently pursuing MS degree program in Computer Science from University of Lahore, Pakistan, E-mail: malik.touqir.anwar910@gmail.com*
- *Saira Kishwer is currently pursuing MS degree program in Computer Science from University of Lahore, Pakistan, E-mail: saira.abbasi11@gmail.com*
- *M. Yaqoob Wani is HoD SIT in University of Lahore, Pakistan, E-mail: yaqoobwani@gmail.com*

of internet has also increased the number of vulnerable systems (zombies) all around the world which can be used to launch a very sophisticated Distributed Denial of Service (DDoS) attack. Now it has become quite possible for the attackers to recruit a large amount of host systems (zombies), with or without their knowledge, and launch a large scale attack on a target in any place of the world [3]. The distributed nature of the attacks makes them more effective by increasing the strength and makes them difficult to detect. A DDoS attack is coordinated attack strategy which is aimed at the denying the services to a legitimate users.

DDoS attacks in recent times exploited the UDP traffic. UDP protocol works at the transport layer. This protocol does not provide any form of reliability mechanism, making it less time consuming for end to end traffic flow. The UDP protocol provides direct access to IP layer. Unlike TCP, UDP is a connection-less protocol. Due to this reason, UDP protocol has to face the issues of reliable delivery of the data packets, congestion/collision avoidance, flow control and much more. This unreliable nature of UDP prohibits the users from sending important data over this protocol. Instead, it is mostly used for ping messages, checksum data/results and multiplexing the various ports.

The UDP flooding DDoS/DoS attacks use quite different approach in comparison to TCP and ICMP attacks. These two types of attacks are most common and most of the organizations have mechanism installed which protects against TCP and ICMP attacks. Since the existing DoS/DDoS prevention mechanism is to protect against the TCP and ICMP attacks, it makes UDP flooding more deadly. UDP flooding attacks against DNS servers, routers and switches can cripple a whole network.

Today it has become a trivial job to launch a DDoS attack against an organization. On the other hand, it is quite difficult to detect and then respond accordingly to such an attack[4]. Overtime, there have been numerous solutions which were specific only to applications that generate bogus UDP traffic. However there is a dire need of a generic solution for UDP based DDoS attack which can both detect and prevent the UDP flooding attacks. To achieve this objective, it is important to develop a thorough understanding of the UDP based network traffic which would help in distinguishing valid and harmless traffic from the network traffic which is part of a flooding attack.

2 DDoS Attack Architecture

There are two basic architectures of DDoS attacks:

2.1 Agent-handler architecture

The basic components of agent handler architecture are:

- Clients
- Handlers
- Agents

The client system is used by the attacker to communicate with the rest of the attack network. Handlers are the complete software packages which are used by the client for communicating with the agents. The compromised systems which have become 'zombies' or 'bots' contain agent software. These agent software carry out the DDoS attacks, without the knowledge of the users of the system.

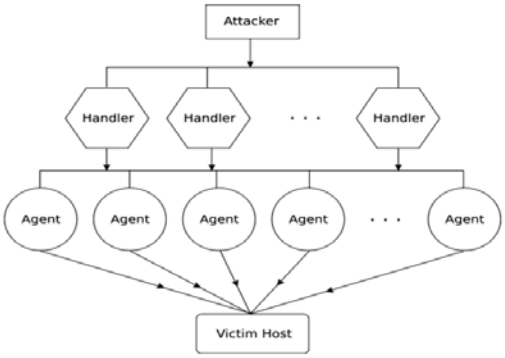

Figure 2: Agent-Handler architecture

2.2 IRC based architecture

This narchitecture relies on an IRC communication channel for establishing a connection with the clients and agents. The attackers use the IRC communication ports for transmitting instructions to the agents. Embedding the commands within the normal communication channels makes the DDoS attacks deadlier and virtually impossible to be traced. As the IRC channels have huge amounts of data traffic, this makes it easier for the attacker to hide within these channels.

A notorious hacktivist group 'Anonymous' released an IRC based DDoS attack tool called LOIC (Low Orbit Ion Cannon). This tool provides three modes of attack i.e. TCP, UDP and HTTP [5]. It even makes it possible for the clients to establish a remote connection and become part of a 'Botnet'. Size of a botnet has a directly proportional relation to strength of a DDoS attack. The greater the number of botnets, greater would be the strength of the attack.

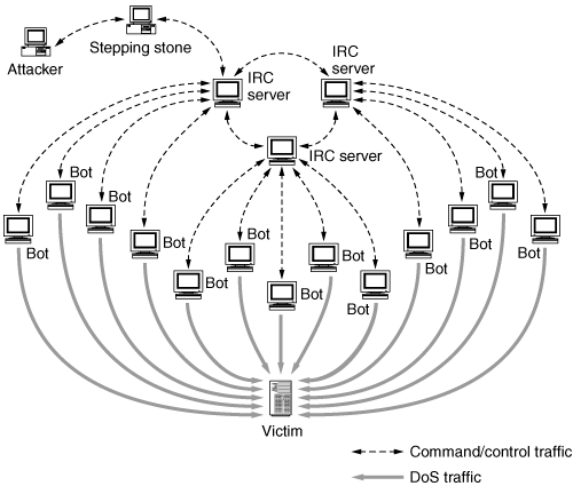

Figure 3: IRC based architecture

With the sophistication of DDoS attacks and attack tools, researches have proposed certain defense mechanisms which have also grown in sophistication.

3 DEFENSE ARCHITECTURE

The DDoS defense mechanism fall under three categories in terms of deployment:

1. Victim end mechanism
2. Source end mechanism
3. In-network mechanism

In practice, the source end defense mechanism is found to be very successful but the problem faced in this approach is that legitimate users may be denied resources at certain times [6]. This makes the source end defense against DDoS to be quite difficult. The DDoS defense mechanism can be either supervised or unsupervised.

3.1 Victim-end defense mechanism

The victim-end defense mechanism is mostly installed at the routers of the victim's network which provide important web services like DNS. The below given figure shows a general architecture of victim-end defense mechanism.

The detection engine is the most important component of this architecture. It is responsible for detecting online/offline intrusion. The detection engine uses one of the two techniques for detecting intrusion. These are a) misuse-based detection b) anomaly-based detection. The database must contain information regarding all the well-known intrusion signatures as well as profile of the normal network behavior. The database information needs to be constantly updated.

Security manager is responsible to keep the database updated, as well as keep track of false alarms. DDoS attack can be detected very easily in a victim router as the resources are suddenly consumed at very high rate. Although this approach can be easily implemented but it has a few disadvantages. In case of a DDoS attack, the bandwidth of the entire network can be consumed by the DDoS attack traffic and the routers fail to stop such traffic at their boundaries [7]. Another major problem with this technique is that an attack is detected when the victim starts facings its symptoms and is unable to access the resources. Such a defense mechanism which detects an attack when the victim is already deprived of the resources is of no use at all.

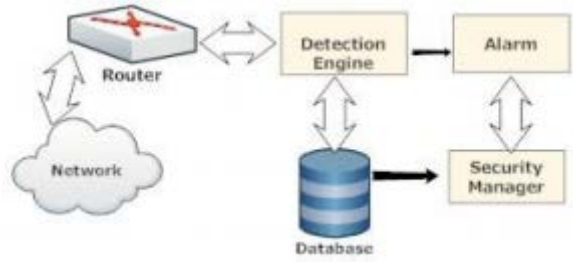

Figure 4: Victim end defense mechanism architecture

3.2 Source-end defense mechanism

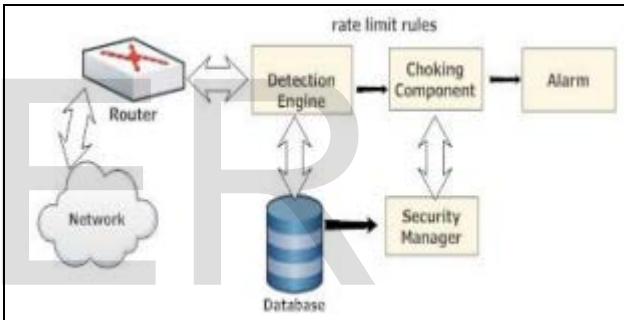

Figure 5: Victim end defense mechanism architecture

The above figure shows a common source-end DDoS defense mechanism. As can be seen in the figure, the source-end defense mechanism is quite similar to victim-end defense mechanism. The only difference is that of a choking component [5]. The purpose of choking component is to enforce a limitation on rate of established connections. An important thing to be noted here is that in this architecture, the traffic statistics of incoming and outgoing architecture are compared against certain pre-defined threshold values. This increases the overall intelligence of the system. It has been found that source-end defense mechanism is very efficient and yields good results. This approach protects the victim as well as all of the intermediate from being flooding. The disadvantage of this mechanism is that due to the distributed nature of the attack, it is very difficult to prevent as the sources are geographically dispersed all over the world [8]. Even if every potential single source is observed, it would appear to be working normally. The inbound or outbound statistics will not show any irregular spikes.

3.3 Intermediate network defense mechanism

The two main problems in source-end and victim-end defense are that of accurately detecting the source and preventing high percentage of bandwidth consumption by the attack. To overcome this problem, the intermediate network defense mechanism optimizes both of these problem features and makes careful tradeoffs.

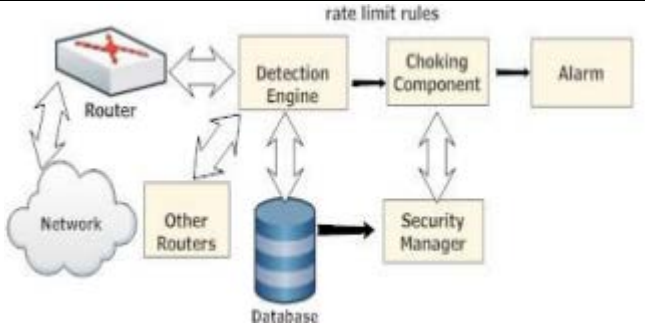

Figure 6: Intermediate network defense mechanism architecture

The above figure illustrates a general intermediate network defense mechanism architecture. This defense mechanism is based on the collaborative nature of routers. The routers are made to share their knowledge with their neighbors. This defense mechanism makes use of the promising feature of source-end defense mechanism by limiting the transfer rate in comparison to certain threshold values. This rate limiting is performed at the routers which monitor incoming and outgoing connections. By using this technique, it becomes very easy to first detect and then perform a 'traceroute' to the attack source, as all the routers share their own knowledge. It is possible for routers to make a mesh of their own for sharing their information [8]. The problem with this technique comes with its deployment. Attacks can be detected with maximum accuracy only when all the routers are using this detection mechanism. This is important because if some of the intermediate routers are not using this scheme, they will not keep any records. Consequently, they will not be able to compare the statistics to any threshold value and as a result they will not be able to participate in the trace-back process. As a result, this fact makes the intermediate defense mechanism impossible to implement.

4 DDoS ATTACK STRATEGY

Launching a DDoS attack is a systematic process and follows the steps as shown in the figure given below

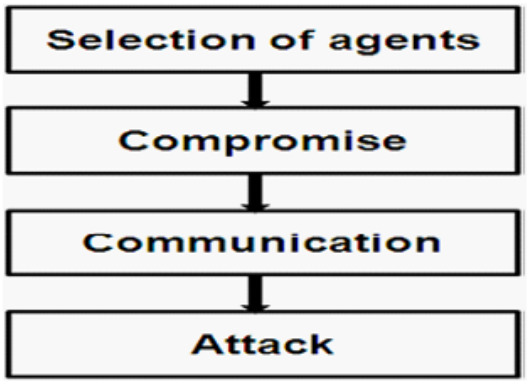

Figure 7: steps to perform a DDoS attack

4.1DDoS attack elements

4.1.1 Selection of agents

As indicated by the name, DDoS is a distributed effort. The first thing an attacker does is to recruit agents that will perform the attack [9]. The agents are chosen by exploiting the vulnerabilities in their systems. Normally such systems are chosen to act as agents for attackers which have more resources than are actually used by their legitimate users. As the users do not the resources fully, most often than not, they fail to detect the unauthorized usage.

4.1.2 Compromise

Once the agents are selected, the attackers exploit the loop holes in the security of such systems and insert their own personalized code into the agents which performs the DDoS attack. It is very important for the attackers to hide their own identity. They take special precautionary measure to hide their footprint in the code. As a result, the zombies become an accessory to the attack on the victim, without them being aware of this fact [3]. It is very difficult for an agent to realize that its security has been compromised unless very sophisticated tools are being used. There are several automated tools present which can perform the DDoS attack. These tools take extra measures to use least possible amount of memory and bandwidth of the agent/zombie to avoid detection.

4.1.3 Communication

It is very important for the attacker to communicate with a large number of handlers for identification of functional agents to schedule the attack or to upgrade the attack code on the agents. This communication is not limited to any particular protocol.

4.1.4 Attack

After all the preliminary preparation is done, the attacker now begins the attack. It is possible to customize the features of the attack; for example type of attack to launch, length of the data packets, TTL (time to live), port numbers etc. It is important for any successful attack to keep these feature adjustable, thereby making it more difficult to detect the attack.

5 DDoS ATTACK: SCOPE AND CLASSIFICATION

As discussed above, the distributed nature of the DDoS attacks makes them difficult to counter and the source of the attack to be traced. Using a spoofed IP address for launching a DDoS attack is very easy [10]. Internet has grown exponentially over the last decade or so. Such unregulated growth of internet has also given birth to countless vulnerabilities which can be exploited. What happens is that by the time a victim realizes that it is under attack, nothing much can be done to protect the victim, except for manually disconnecting from the network.

There are two major types of DDoS attacks based on the protocol that is targeted. These are discussed below[9].

5.1 Network/transport-layer DDoS flooding attacks

This attack is launched by using the protocols which work on network/transport layer. These protocols used are TCP, UDP, ICMP and DNS. On the basis of these protocols, there can be five types of attacks. These are:

5.1.1 Flooding attacks

The network bandwidth of the victim is flooded by bogus data which consumes all of the bandwidth.

5.1.2 Protocol exploitation flooding attacks

Attackers look to manipulate some specific features of the protocol which use up the network bandwidth, e.g. SYN-flood.

5.1.3 Reflection-based flooding attacks

Instead of legitimate requests, the attackers flood the victim with spoofed requests. The victim responds normally to these requests and as a result, the bandwidth is consumed.

5.1.4 Application-level DDoS flooding attacks

These attacks are aimed at consuming the server resources so that these are not available to the legitimate users. It has been observed that application level DDoS attacks use comparatively less bandwidth, thereby making them more difficult to be detected. Application level attacks target the following protocols: HTTP, DNS, SIP.

5.1.5 Reflection/amplification based flooding attacks

Reflection/amplification attacks have been proved very difficult to detect. The attackers send countless DNS request to the server with spoofed IPs. Since a DNS response is bigger in size than a DNS request, this creates huge amount of unnecessary network bandwidth. The network bandwidth is consumed by a large amount of DNS request and DNS response traffic and the resources are exhausted [11].

5.2 HTTP flooding attacks

A large number of zombie systems make request for sessions to the server. Consequently, the number of requests from zombies increase in number as compared to legitimate requests. As a result, the server's resources are consumed by zombies and valid users are denied of their required services [2]. Consequently the server is flooded by the request from zombies.

5.2.1 SYN Flooding Attack

A SYN flooding attacks exploits the vulnerabilities of the TCP protocol design. TCP is a connection-oriented protocol. It uses the 3-way handshake mechanism, as shown in the figure given below, to establish connection. The attacker sends SYN packets to the victim with spoofed IP addresses. Server has no knowledge that it is under attack. It considers all the packets to be legitimate [4]. The server stores the state information of these connections containing source IP address, source port number, destination port number, destination IP address. In response to each SYN packet, the server responds with SYN-ACK addressed to each spoofed IP address. This traffic of SYN and SYN-ACK packets consume the bandwidth almost completely. Also, the resources of serves are wasted by storing state information of bogus connections. As a result, the server gets busy with processing the request of the attacker and the legitimate users are denied of the server resources.

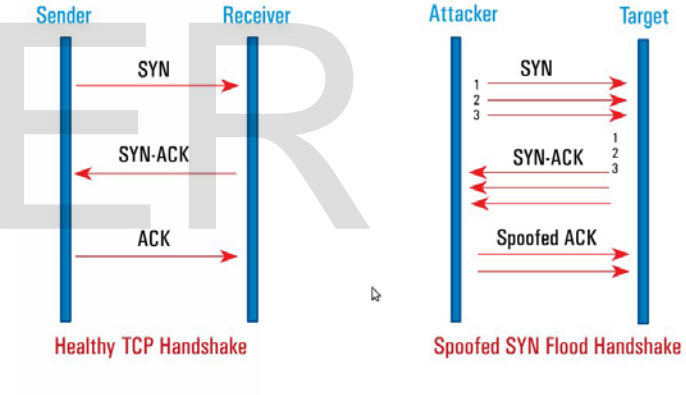

Figure 8: TCP 3-way handshake mechanism and SYN attack

5.2.2 Theory of Operation

SYN flooding attacks exploit the vulnerabilities of the TCP implementation. A SYN flood attack optimistically assumes that the for each SYN packet, the victim server reserves a state and there exist a specific limit for the number of these states. The feature that is exploited by the attacker is that every TCP connections takes up some amount of memory in the server. If a large number of connections are created, no space will be left for legitimate users [12].

Initially the TCP port of a server is in listen state. When a port receives SYN packet, the state of the port is changed from LISTEN to SYN-RECEIVED and the state information that came along with the SYN packet is stored. The overhead of saving the information in case of every SYN packet is in the range of 280 bytes-1300 bytes, depending upon the nature of TCP implementation.

It is important to note that a SYN flooding attack tries to consume the resources of the server by increasing the overhead created by half open connections. This attack does not exhaust the network resources. The main objective of a SYN attack is to transmit a large number of SYN packets, preferably from spoofed IP addresses, and generate an equally large number of SYN-ACKs in their response. The SYN packets establish half open connections and as a result, valid requests by legitimate clients of the server are denied. The three most important characteristics of a SYN flood attack are discussed below [9].

5.2.2.1 Barrage Size

In an ideal scenario, the barrage size of a SYN flood attack must be equal to the backlog limit of a server. A SYN flood attack will only be successful only if the size of the barrage is nearly equal to the backlog limit. Barrage size larger than the limit would unnecessarily consume the network bandwidth, making it prone to detection. On the other hand a barrage size smaller than the backlog limit would not produce desired results as some amount of users will be able to establish connection with the server [1].

5.2.2.2 Barrage Frequency

An important feature of TCP is that it implements a lifetime limit on the half opened connections. After a certain time, the half opened connections are converted to fully opened connections and the server takes back the allocated memory. This feature of the protocol manages to limit the number of half opened connections very effectively. A timer can be initialized while sending the very first SYN-ACK packet. Upon expiring of this timer, if the connection is still in half open state, the memory can be taken back. Although this time limit is not a default feature of TCP, yet some operating systems implement this independently.

To overcome this feature, a SYN flood attack must need to send the barrages at regular time intervals, as soon as the resources are claimed back by the server. If the frequency is kept higher than what is actually required, it would make the attack more visible. On the other hand, if the frequency is kept low, it would allow valid users to establish connection to the server [13].

5.2.2.3 IP Address Selection

Another important factor of a SYN flood attack is that the spoofed IP addresses used must not respond to the SYN-ACK packets. If valid IP addresses are used instead of spoofed IP addresses, the agents will send a packet in response to SYN-ACK and then free the established connection. An effective approach is to use a whole list of unresponsive IP addresses that will keep the connections in half open state [5].

6 Conclusion

Some researchers consider TCP SYN flooding to be a bug in TCP protocol suite whereas others believe that this is more of a feature of the original protocol design. The logic behind such connection-oriented approach of TCP was to personalize the internet and make it more user-friendly. This idea worked for several years. Some of the TCP improvements include increasing the queue length and reducing the lifetime of a half opened connection. However the length of the queues cannot be increased indefinitely as each lengthening each queue requires significant amount of memory to be reserved. Decreasing the time limit for opened connection also has a limit. If the lifetime is increasing beyond a specific limit, remote users having slow internet connections will never be able to get connected to a server [11].

The best possible solution to SYN flood attack is to bring such changes in the implementation of TCP which store less amount of information with each connection. Another solution could be of performing a tracerout/traceback of new connections. If the route is a different one than the route of received packets, such connections should be dropped. There is dire need to revamp the TCP/IP protocol suite. If the researchers come up with ways of lengthening the queues without taking up large amount of memory, the only problem with queue size would be solved [6].

It can be very accurately stated that if a server or a computer is connected to the internet, it is very much susceptible to any kind of DDoS attack, unless proper security measures are taken. There are a number of different solutions for a number of different attacks.

References

[1] A.M. Lonea, D. P. (2012). Detecting DDoS Attacks in Cloud Computing Environment. International Journal of Computer Communication , 70-78.

[2] Dileep Kumar G, C. G. (2013). A Survey on Defense Mechanisms countering DDoS Attacks in the Network. International Journal of Advanced Research in Computer and Communication Engineering , 2599-2606.

[3] Farrow, R. (2004). TCP SYN Flooding Attacks and Remedies. Retrieved from Network Computing: http://www.networkcomputing.com/unixworld/security/004/004.txt.html

[4] Fu-Yuan Lee, S. S.-T.-H. (2005). A Source-End Defense System Against DDoS Attacks. In Computer Security in the 21st Century (pp. 147-168). Springer US.

[5] Haining Wang, D. Z. (2004). Change-point monitoring for the detection of DoS attacks. IEEE Transactions on Dependable and Secure Computing , 193 - 208.

[6] Hakem Beitollahi, G. D. (2012). Analyzing well-known countermeasures against distributed denial of service attacks. Computer Communications , 1312-1332.

[7] Karre, S. K. (2013). Distributed Detection of DDoS Attack. International Journal of Future Computer and Communication , 628-632.

[8] Laxmi Bala, A. V. (2012). Quality based Bottom-up-Detection and Prevention Techniques for DDOS in MANET. International Journal of Computer Applications , 12-19.

[9] Mohan K Mali, P. A. (2013). Review of DDoS and Flooding Attacks in MANET. International Journal of Emerging Technology and Advanced Engineering , 480-485.

[10] Saman Taghavi Zargar, J. J. (n.d.). A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks. IEEE COMMUNICATIONS SURVEYS & TUTORIALS .

[11] Saurabh Ratnaparikhi, A. B. (2012). DDOS Attacks on Network; Anomaly Detection using Statistical Algorithm. International Journal of Advanced Research in Computer Science and Software Engineering , 321-326.

[12] Yu Chen, K. H.-S. (n.d.). Collaborative Detection of DDoS Attacks over Multiple Network Domains . IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS .

[13] Zhengmin Xia, S. L. (2005). Enhancing DDoS Flood Attack Detection via Intelligent Fuzzy Logic. Informatica , 497-507.