Denial of Service attack using Slowloris

Krishna Kant Nath Tiwari[1]
Galgotias University, U.P

Mr. S Rakesh Kumar[2]
Assistant Professor, Galgotias University

---***---

Abstract- Attacks on the Denial of Service broadcasts remain among the most dangerous and visible attacks on the Internet. Contrary to previous attacks, most recent DDoS attacks did not occur over the network layer, but over the system layer. The main difference is that ultimately, an attacker can look for a specific server system, while leaving the remaining systems in place , thus producing less saturation and more difficult to find. Such attacks are possible by exploiting the terms of the application layer used. Denial of Service Attack is any type of attack on a network structure to disable the server being used by its customers. The attack of sending millions of requests to the server in an effort to slow it down, degrading the server with large packets of inactive data, sending the request with an IP address. In this paper we introduce the implementation and analysis of Slowloris.

Keywords- HTTP request and response, bandwidth, DoS, Slowloris, Partial HTTP request

1. INTRODUCTION

DoS attacks simply indicate denial of service attacks. This source can be of any kind, for example imagine that your mother hides your cellphone when preparing your exams to help you study without interruption. Your mother's purpose without care and attention; you are denied the calling service and other functions provided by your phone. In terms of computer network and computers, a denial of service could be in the form of –"Hijacking web server, port overloading or denying any services that are available on the internet".

This attack can be performed on a single machine, a single machine attack is easy to perform and monitored, as it is easily detected. Although DoS attacks do not always lead to theft or loss of important information or other assets, it can cost a victim a lot of time and money to handle. To solve this problem, this attack can be performed from multiple devices distributed over a wide area, in this way it is difficult to stop the attack and it is very difficult to pinpoint the cause of the attack, such attacks are called "Distributed Denial of Service" (DDoS Attack).

A. Ways a DDoS attack be performed

DoS attack can be done in a several ways. The basic types of DoS attack in

- Flooding the network to prevent legitimate network traffic.
- Disrupting the connections between two machines, thus preventing access to a service.
- Preventing a particular individual from accessing a service.
- Disrupting a service to a specific system or individual.
- Disrupting the state of information, such resetting of TCP sessions.

B. Problems caused by DDoS attack

DDoS attacks can cause the following problems:

- Ineffective services
- Inaccessible services
- Interruption of network traffic
- Connection interference.

2. ATTACK TECHNIQUES

Most attack techniques can be used for DoS purposes as long as they can disable the service, or limit the functionality of the service by terminating the services of the service provider. Although it is not possible to read all of the attack techniques, we describe many attacks involving broadcasters in this section to show the goals of the attack..

A. Host based Attacks

These types of attacks usually work in a particular type i.e. they use a certain algorithm, the law of authentication, and the application. We use IDS-based attacks. IDS-based hostels are included in hostels so that they can monitor the traffic flowing to and from that home only. Location-based IDSs can be used for check or monitor user login procedures that apply to the user system, data integrity. These IDS can also tell you whether or not the attack is successful. We know that an attacker uses the same traffic to launch this attack, so it was easy for us to determine who the attacker was. Attackers can also launch DoS attacks by exploiting the vulnerability that exists in the target system. Traffic to a derived owner is not as high as a network due to a feature in the application that can interrupt an application or use multiple program resources [3].

B. Network based Attacks

Network-based attacks are threatened or malicious code presented and managed by one or more tools. Denial of service (DoS) or Distributed denial of service (DDOS) are examples of network attacks. In this type of attack, the attacker sends a large number of requests to the server that can handle other requests to that server. We use fire extinguishers or IDSs to detect this type of attack. Network-based IDSs are positioned on the network and therefore are able to detect any attacks on that network owner [3]. We install IDS inside and outside of our network so we can get all the information that goes through that network.

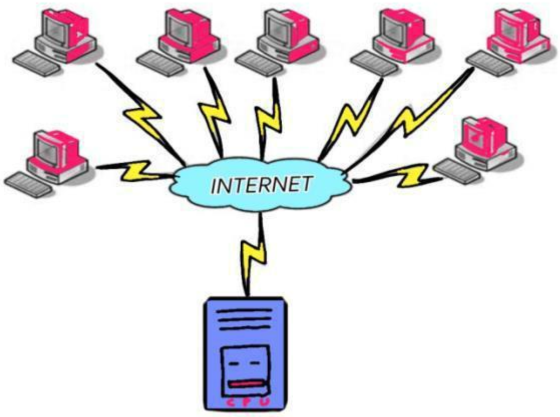


Figure 1 : Network based attack (DoS)

a) DDoS attack:

In a DoS attack, one machine is used to perform the attack. While in DDoS attacks many machines are used, every machine sends requests to a server or network that will eventually use all server or network resources. Denial of Service is basically a cyber attack on a particular server or network with the purpose of interrupting the normal operation of the networks and server. DDoS attacks do this by flooding the deployed server or network with a general flood of traffic such as fraudulent requests that terminate the system, cause interruptions or rejection of services on network server.

The question arises “how does the attacker get other computers to get involved in a DDoS attack?

The simple answer to this question is that, by using malicious software, an attacker can develop a malware program and distribute it online to things like websites and email attachments. Computers are affected by malware that interacts with other computers called “Botnets”. These botnets are not limited to hundreds or thousands of computers, it can be scattered across the world.

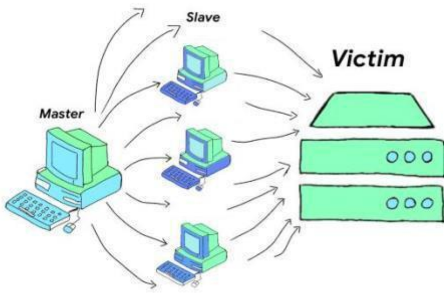


Figure 2: Classification of DoS Attacks

COMMON DDOS ATTACKS TYPES

- *SYN Flood:*

A SYN flood DDoS attack is known as weakness in TCP connection sequence ("handshake method)" initially SYN request goes then SYN-ACK response comes back and then ACK request is sent. It is a denial of service when an attacker sends a sequence of SYN requests to the target system in an attempt to utilize sufficient server resources to make the system unresponsive to legitimate capacity [4]. SYN Flood attacks can result in all of the damage associated with DDoS attacks, financial loss, loss of customer loyalty, Hardware and software damage, financial information theft.

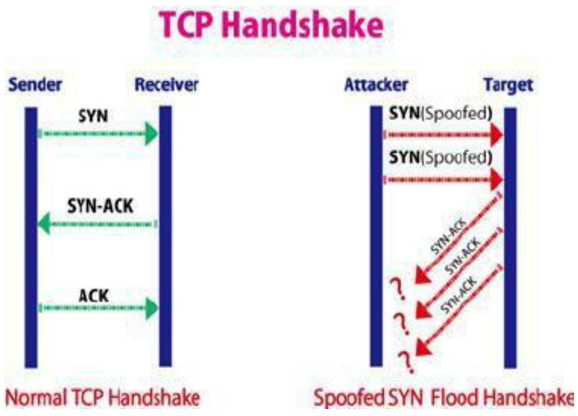
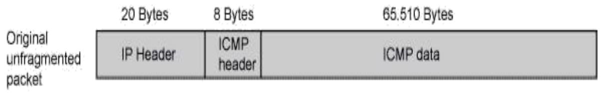


Figure 3 : Handshake in SYN Flood

- *Ping of Death:*

It uses the fact that in the TCP TCP protocol the maximum packet size is 65,535 bytes. The ping of Death Attack uses this fact [4]. In this attack the attacker sends packets that are more than the size of a large package when collecting packets. This can override the memory provided by the packet and the computer usually does not know what to do with such packets and keeps the packets or sometimes a complete crash causes a denial of the service of the official packets.



The size of this packet is 65.538 bytes. It exceedes the size limit prescribed by RFC 791, *Internet Protocol*, which is 65.535 bytes. As the packet is transmitted, it becomes broken into numberous fragments. The reassembly process might cause the receiving system to crash

Figure 4: Fragmenting of Packets [6]

- *UDP Flood:*

A UDP flood, is a DDoS attack that attacks User Data Protocol (UDP) packets. The User Data Protocol (UDP) is offline and the session protocol is network communication. The UDP flood does not need to be handled as TCP (Transfer Control Protocol) [4]. If there is no handshake, to establish a valid connection, we send a large amount of data to the UDP channel to any host or server. This means that the UDP flood not only works but it can work with the help of a few resources. The purpose of this attack is to fill random ports on the remote keeper. This allows the custodian to check if there are any programs that respond to requests for that particular position. If there are no plans to receive packets in that port, the server responds by ICMP "Inaccessible" (ping) to notify the sender that the inaccessible location is not working.

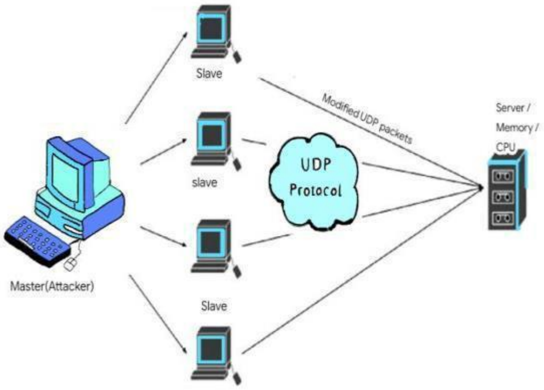


Figure 5: UDP flooding

- *HTTP Flood:*

An HTTP flood is a type of DDoS attack. In this attack, the attacker uses a valid HTTP GET or POST request to attack the server or request. HTTP Flood Attack is a volumetric attack and uses "Internet" botnets. Computers are infected with malicious software, often with malware such as Trojan Horses

[4]. HTTP Flood is layer 7 DDoS Attack. The GET request is used to access general content, such as images while using POST requests to access enabled resources.

Instead of using unplanned packets, host or display modes, HTTP flooding requires less bandwidth to attack targeted websites and servers. This attack is most effective when we try to force a server or programs to allocate more than one application. This attack is difficult to detect because you are using a valid HTTP request. These requests use all server resources and cause the server to decline.

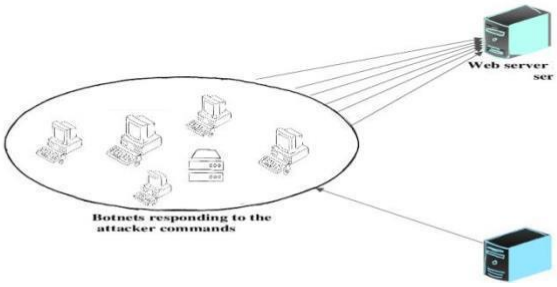


Figure 6: HTTP Flood Attack

3. SLOWLORIS

It is a protocol attack means it is a layer 7 application attack. It is developed in such a way that the server wait for its requests. In this case the request is so slow that the server drops. In other words if we send a request and then the server waits for 10 to 20 seconds and then it is released, that is an HTTP request to take requests from another user. But when the server is about to clear a hole we remind the server that we are still coming and send any small bytes to the server and we will do this thing over and over again. In doing so, we log on to the server for more requests. We try to crack or defeat a web server or computer by offering an equivalent request (bandwidth) that we can instantly send to a server or computer crash. Knowing that each server has a specific bandwidth (requests) to use at the same time, our focus is to send requests or use more bandwidth than the server can handle. So in denying service or slowloris, we offer more requests than the server can handle and the server goes down. If we deploy that many programs it will not affect any web server. Slowloris is very difficult to identify or detect by a firewall because it sends HTTP valid requests but in a slow way. It's probably because we have a slow internet connection. Not applicable to all servers (Red Hat web server, IBM web server, Oracle web server). It mainly affects the Apache web server. Now a days, there are about 45 - 50% of web servers that use Apache. At Apache, when they designed it they thought it would be a good idea to start a new thread to take advantage of all the connections, so when a connection comes with an HTTP request, they create a new thread that will handle that request once the thread runs when the request is completed.

So we use this retreat to open the connection for longer than we think, when our communication limit is reached. Suppose that the connection limit of an Apache web server has 200 objects in common. Therefore, slowloris first consumes all open connections, and then waits for other connections to be free. As a new connection is released, from someone using the website instead of using it then it goes on until all communications have been completed.

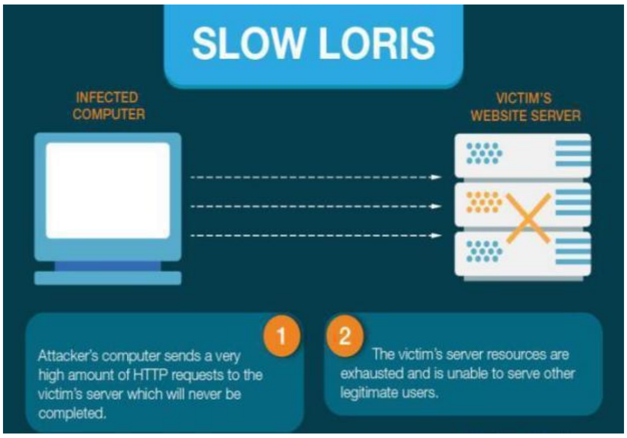

SLOW LORIS
INFECTED COMPUTER
VICTIM'S WEBSITE SERVER
1
Attacker's computer sends a very high amount of HTTP requests to the victim's server which will never be completed.
2
The victim's server resources are exhausted and is unable to serve other legitimate users.

Figure 7: Slowloris [5]

The idea behind this tool is that it only affects the HTTP-managed service, without affecting other servers running the server. The name "SLOWLORIS" is a good fit for the tool, due to the simple fact, that it can directly remove a web server by slowly removing it from all server connections. When we make an HTTP request to the server, it responds to an HTTP response. Slowloris creates a partial HTTP connection to the host by requesting only from the server when the server requires user input but causes the server to wait for its next install. At a time, when the server thinks that the user is now offline and wants to disconnect, when the user transmits a small amount of data to make the server feel like it is running on that server. The Apache server has a timeout machine. The apache server will wait approximately. 300 seconds to complete the program. We can change the timeout limit. The opposing session is reset every time a user makes any request to the server for a calculation of the period of time i.e. will start from 1 [7]. When slowloris uses all communication by requesting a fixed request from the server then continues to maintain the connection by sending request data and resetting the shutdown timer.

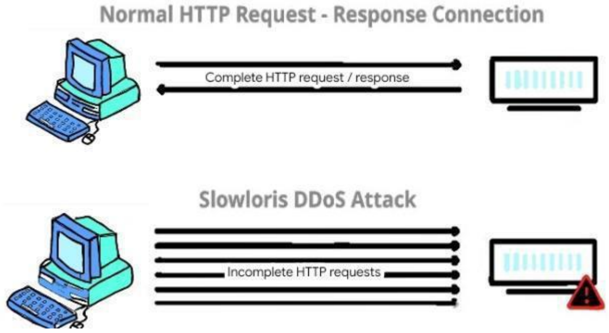

Normal HTTP Request - Response Connection
Complete HTTP request / response
Slowloris DDoS Attack
Incomplete HTTP requests

Figure 8: Slow HTTP attack (Slowloris)

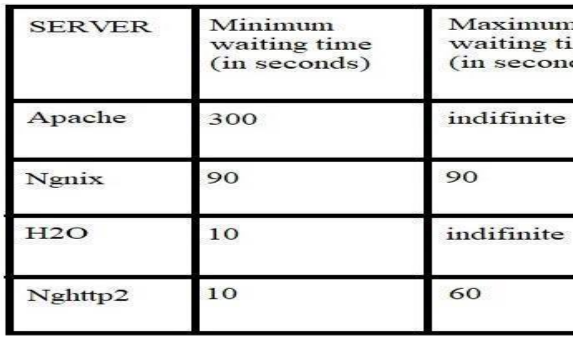
4. PROPOSED METHOD

Attack:

Slowloris HTTP DoS attacks are performed using incomplete or POST GET requests. A malicious client or attacker launches this attack using GET / POST Header. To launch this attack using a GET request, the attacker sends the headers framework END_HEADER and END_STREAM to reset and reset respectively. Headers frame is used to handle a blocker, if the blocker is large enough to fit into a single HEADERS frame using the frame CONTINUTION. The END_HEADER flag, when reset, indicates that the HEADER frame must be followed by one or more validation frameworks. Continuity Framework exists since a single value for example a cookie may exceed 16KiB (kilobyte). It shows that the value cannot be equal to one frame. It has been determined that the tendency of the smallest in the continuation frame is that it requires that all header information comes independently from the back, making it easier to manage and write buffer. The last frame for continuation must contain the en headers flag. When a server receives such a headers framework, the server thinks it has received an incomplete header and must wait to receive the complete block. We use this opportunity to keep the server busy and prevent the server from releasing the port. This way, we try to find all ports so that the server cannot accept any new GET requests. To launch this attack using POST requests, the attacker sends a HEADER framework containing a POST request and has both END_HEADER and END_STREAM to reset the flag. So, there are two flavors of this attack depending on the HEADERS frame type.

Behavior of Web Servers Against Attack:

Before closing a connection, Apache and Nginx wait for 300 and 90 seconds respectively after receiving a HEADERS frame having END_STREAM flag set and END_HEADERS flag reset. However, if malicious client keeps sending continuation frames having END_HEADERS flag reset at regular intervals, Apache waits for an indefinite amount of time by resetting its expiry timer again and again. However, Nginx do not reset the expiry timer and after the waiting time expires, Nginx closes the connection by sending GOAWAY frame with PROTOCOL_ERROR code. Both H2O and Nghttp2 just wait for 10 seconds after receiving such HEADERS frame. After timeout, H2O and Nghttp2 close the connection by sending GOAWAY frame with NO_ERROR and HEADER_TIMEOUT code respectively. However, if malicious client keeps sending CONTINUATION frames having END_HEADERS flag reset at regular intervals, H2O waits for an indefinite amount of time while Nghttp2 waits for 60 seconds. After this timeout, Nghttp2 closes the connection by sending RST_STREAM frame with INTERNAL_ERROR code. Table I show minimum and maximum time for which different server implementations wait before closing the connection if payloads corresponding to attack are sent [1].



SERVER	Minimum waiting time (in seconds)	Maximum waiting ti (in secon
Apache	300	indifinite
Ngnix	90	90
H2O	10	indifinite
Nghttp2	10	60

Figure 9: Connection Waiting Time at Server if Slow HTTP Attack Payload is sent

c. Proposed Attack Technique over TLS:

In order to test the effectiveness of proposed attacks using encrypted HTTP/2 requests, we repeated the above experiments by configuring web servers to use TLSv1.2 while serving the requests. We used a python script on the malicious client. The python script was automatically executed multiple times and so that malicious client could establish multiple concurrent connections with the server and send an encrypted HTTP/2 request from each of those connections. Once enough number of connections was established, genuine client was used to check the availability of server. From this experiment, we observed that malicious client required same number of connections to create DoS scenario using either clear text or encrypted HTTP/2 requests.

5. CONCLUSION

There were reviewed and analyzed known slowloris attack in this paper. The further development can be done on focusing on the type of DDOS mode because they are harder to detect and the attacker can reduce the number of connections from each machine or bots to reduce the chance of detention or disconnecting by the server. In this way, even that attacker can assimilate the flooding attack, if he or she uses a lot of bots. The development can also be done by making the changes in packets that can be its size, generation and its efficiency.

REFERENCES

Research Paper

[1] Tripathi, Nikhil & Hubballi, Neminath. (2017). Slow Rate Denial of Service Attacks Against HTTP/2 and Detection. Computers & Security. 72. 10.1016/j.cose.2017.09.009.

[2] T. Dierks and E.Rescorla. (RFC 5246) The Transport Layer Security (TLS) Protocol Version 1.2.2008 Online Links

[3] Online 15/03/2019 https://securitywing.com/host-based-ids-vs-network-based-ids/

[4] Online 12/03/2019 https://www.incapsula.com/ddos/ddos-attacks.html

[5] Online 13/03/2019 https://www.globaldots.com/ddos-distributed-denial-service-explained/attacks_w_signatures-04/

[6] Online 13/03/2019 https://www.juniper.net/documentation/en_US/junos/topics/topi c-map/security-os-specific-dos-attack.html

[7] Published by Computerphile 17/03/2019 https://www.youtube.com/watch?v=XiFkyR35v2Y