International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 8, August - 2013

A Survey on Different Approaches to Mitigate DDoS Attacks

Treesa Nice P. A.
Dept. of Computer Science and Engineering
Rajagiri School of Engineering and Technology

***Abstract*—This paper contains a survey on different types of Distributed Denial of Service (DDoS) Attacks and the different methods to mitigate these types of attacks. There are bandwidth based attacks, application level attacks, protocol based attacks etc. Examples for these attacks are described in this paper. Besides that, the methods for preventing from these attacks also described in this paper.**

Keywords-dos, ddos, attacks, mitigation.

I. INTRODUCTION

Internet is growing day by day. Attack also increases by the same scale. There are different types of attacks. One of these types of attacks is denial-of-service (DoS) attacks. It is an effort to make a network resource or service unavailable to the genuine customer. The attack may come from a solitary system or a set of compromise systems. If the attack is from a group of compromised systems, it is known as distributed-denial-of-service (DDoS) attack.

DDoS attacks can be of different types. It can be bandwidth based or volumetric attacks, protocol based attacks, application level attacks etc. Volumetric type attacks are the most common type of DDoS attacks. It means, overflow the network with pointless data. A large amount of requests will be sent to the server. If this is larger than the capacity that the server can handle, then the server cannot process all these requests. Then the legal users will not get the response. The server may be crashed. In this type, the bandwidth will be high. So considering the bandwidth, one can understand whether any denial of service attack is happening or not. But in application level attacks, the applications will be directly attacked. In this case, the bandwidth will be less. So by considering the bandwidth, one cannot understand whether any DoS attack takes place or not.

There will be different methods for preventing from DDoS attacks. If the attack is from a particular system, the upstream flow from this system can be analyzed to find out the attack. If the attack is not from a single challenger, then it will be difficult to detect and prevent. Filters can be used to each of the routers. Then the quantity of filters needed may be large. For application layer based attacks, port-hopping method can be used to mitigate DoS and DDoS attacks.

The paper is organized as follows. In Section II, the different types of DDoS attacks are described. Then in Section III, the different methods to mitigate DDoS attacks are specified. Section IV concludes our discussion.

II. TYPES OF DDOS ATTACKS

DDoS attacks can be divided into three types. They are Bandwidth Based Attacks or Volume Based Attacks, Protocol Attacks, and Application Level Attacks.

Bandwidth Based Attacks deluge the bandwidth of the attacked site. It consists of UDP floods, ICMP floods, other spoofed packet floods etc. It is measured in bits per second.

Protocol attacks uses actual server resources or those of the intermediate communication resources such as firewalls, load balancers etc. Examples are SYN floods, Ping of Death, fragmented packet attacks, Smurf DDoS etc. It is measured in Packets per second.

Application Level Attacks attack the applications directly. It will concentrate only on a single application and will not affect any other applications. It is used to crash the server. Slowloris, Zero-Day DDoS attacks etc. are examples of application layer attacks.

A. UDP Flood

UDP Flood attack maneuvers the User Datagram Protocol (UDP). It is a session less networking protocol. This type of attack deluges arbitrary ports on a remote host with frequent UDP packets. This increases the work of host. The host needs to continually check for the application listening at that port, and when no application is found, respond with an ICMP Destination Unreachable packet. This practice fades host resources, and can eventually lead to inaccessibility.

B. ICMP Flood

It is similar in principle to the UDP flood attacks; an ICMP flood overcomes the target resource with ICMP Echo Request (ping) packets. The victim will send packets as fast as possible without waiting for any responses. Here the attack will consume a large bandwidth such as those for outgoing and incoming. The server of the victim will attempt to respond with ICMP Echo Reply packets. This will cause an overall system slowdown.

C. SYN Flood

A SYN flood DDoS attack takes the advantage of the weakness of the three-way handshake in the TCP connection sequence. First a SYN request should be sent to initiate a TCP connection with the host. Then this request should be replied by a SYN-ACK response from that host. This should be authenticated by an ACK response from the requester. In a SYN flood situation, two things may happen. First one is

the requester sends multiple SYN requests, but does not respond to the host's SYN-ACK response. The second one is, sends the SYN requests from a spoofed IP address. In both these cases, the host system waits endlessly for acknowledgement for each of the requests, binding resources until no new associations can be made, and in due course resulting in denial of service.

D. Ping of Death

In a ping of death attack, the attacker sends multiple deformed or malicious pings to a computer. The maximum size of the ping packet is 65,535 bytes. A ping packet of length larger than 65,535 bytes may be sent by the attacker. This is one case. In an Ethernet network, the Data Link Layer habitually has limits to a maximum frame size of 1500 bytes. A large IP packet will be fragmented to multiple IP packets and send to the receiver. The receiver will reassemble these IP packets into the complete packet. But in Ping of Death attack, the fragment contents are maneuvered maliciously. So the reassembled IP packet will have a size larger than 65,535. This will overflow memory buffers allocated for the packet, causing denial of service for legal packets.

E. Slowloris

It enables one web server to take down another server, without upsetting other services or ports on the target network. Slowloris first creates as many connections as possible to the target server. Then it will send only partial requests to the server. The targeted server will keep all these false connections open. This will overflow the maximum coexisting connection pool. Then it will deny to grand additional connections from legitimate clients.

F. Zero-Day DDoS

Zero-Day attacks are unknown or new attacks make use of vulnerabilities for which no patch has yet been released.

III. METHODS TO MITIGATE DDOS ATTACKS

If the communication ports in a network based application is opened for a long time, then it will be a strong cause for distributed or simply Denial of Service attacks. Port-hopping can be used a solution for this problem. The processes in communication exchange acknowledgments between each other. But if the acknowledgements are lost, the communication port will be opened for a longer time. This makes the system a target for a directed attack. So in [1], the ports will be changed dynamically. It will not depend on acknowledgements. For that purpose, an algorithm known as HOPERAA is used. In this algorithm, the execution interval will be calculated using some equations. Here the logic is that, there will be a pseudo random function, which will be known to only server and client. Server will send a seed to the client. This seed also will be known only to server and client. Using this seed and the pseudo random function, the client will calculate the next port to which data is sent. So it will be difficult for the attacker to find out the port and start a direct attack. Besides that, in this solution, port-hopping [1] is extended to support cooperative applications. For this function, BIGWHEEL algorithm is used. In this, every application server communicates with more than one client in a port-hopping style. Group management is not desirable in this case. The clock rates of both the server and clients will be different. So the clock of the server is kept as the standard and for clients, clock drifts will be calculated and using this, new execution intervals will be calculated.

In another solution, Active Internet Traffic Filtering (AITF) [2], a method for jamming extremely distributed denial-of-service attacks is used. This paper illustrates the grounds why no efficient DDoS cleaning means has been deployed yet. It explains that the present Internet's routers have enough filtering resources to spoil such attacks, with the condition that attack traffic be blocked near to its sources; AITF uses this finding. AITF can obstruct a million stream attacks within seconds, while it requires only tens of thousands of wire-speed filters per involving router.

The next solution presents a filter-based DoS defense system (StopIt) [3]. Fundamental to the StopIt design is a novel closed-control, open-service architecture. Any recipient can use StopIt to obstruct the undesired traffic it receives. The design is robust to various tactical attacks from millions of bots. This can be used to reduce various attacks such as bandwidth flooding attacks and filter exhaustion attacks that aim to interrupt the timely setting up of filters. StopIt can block the attack traffic from a few millions of adversaries within tens of minutes with restricted router memory.

There are reactive attacks which allow the attack to hppen first, then do the steps to recover from that. The attacks can be prevented proactively also, meaning that, the attack can be prevented before happening it. Secure Overlay Services (SOS) [4] is an architecture that proactively prevents DoS attacks. It focuses on supporting Emergency Services or similar types of communication. In this architecture, a mixture of secure overlay tunneling, routing via consistent hashing, and filtering is used. Then lessens the probability of successful attacks by performing thorough filtering close to the protected network boundaries, pushing the attack point border into the center of the network, where high-speed routers can handle the volume of attack traffic, and bringing in randomness and anonymity into the architecture, making it difficult for an adversary to aim nodes along the path to an exact SOS-protected destination.

Mayday [5] is a structural design that unites overlay networks with lightweight packet filtering to guard against denial of service attacks. The overlay nodes carry out client validation and protocol verification, and then pass on the requests to a protected server. The server is made secure from outside attack by easy packet filtering rules that can be effectively set up even in backbone routers. Mayday generalizes earlier effort on Secure Overlay Services. Mayday separates the overlay routing and filtering. It provides a more commanding set of selections for each.

General Internet Signaling Transport (GIST) Overlay Networking Extension or GONE [6] is the new solution for this problem. In this, GONE creates a half-permanent overlay mesh containing GONE enabled edge routers. This utilizes capability based DoS avoidance and forwards end-to-end user traffic using the GIST messaging relations. GONE's use of GIST on the above of SCTP permits multi-homing, multi-streaming and partial consistency. But only a narrow overhead for upholding the messaging association is established. In this, hosts are recognized by their sole host identities independent of their topologies positions, and simply need (de-)multiplexing in spite of the conventional connection management and other compound functionality in the transport layer. Because of this, a number of advantages for upper layer end-to-end applications, together with intrinsic provisioning of resilience and DoS avoidance in a dynamic and roaming environment are offered.

Another solution for counteracting DDoS attacks is Indirection-based overlay networks (IONs) [7]. In this method, an assumption is made such that adversaries will attack a predetermined and bounded set of overlay nodes causing service disturbance to a small portion of the clients. Adversaries cannot eavesdrop on relations inside the network or otherwise get information that can assist them focus their attacks on overlay nodes that are serious for particular communication flows. A logical model considers both simple and advanced attackers are developed in this. The impact of these attacks on IONs can rigorously disturb communications. Between every pair of edge nodes, a stateless spread spectrum pattern is used to generate per packet path multiplicity using a modified ION access protocol. It defends end-to-end communications from DoS attacks without giving up strong client validation or allowing an attacker with partial connectivity information to continually disturb communications.

In next solution, before sending something to someone, nodes must first gain authorization to send from the destination [8]. To those senders whose traffic it agrees to accept, the recipient will provide tokens or capabilities. Then the senders have to include these tokens in the next sending packets. To check the traffic, verification points will be scattered around the network. By using the capabilities, the verification points will check the legitimacy of the traffic between both end points. If the traffic is not legitimate, then it will be discarded. This method reduces much of the restrictions of the currently accepted methods to DoS based on anomaly detection, trace-back, and pushback.

Recipient of a packet cannot stop the flows of data before that uses the resources of recipient if that flow founds malicious. This is the one drawback of the Internet. Current mechanisms require a lot of information such as state of each flow at routers, ISP association, or the use of an overlay infrastructure to protect against these actions. SIFF, [9] a Stateless Internet Flow Filter, is a new approach which permits an end-host to selectively end individual flows from reaching its network, without any of the above information. Here, all network traffic is divided into two classes. They are privileged and unprivileged classes. Privileged means prioritized packets subject to recipient control and unprivileged means legacy traffic. Privileged channels are recognized using a capability exchange handshake. Capabilities are verified by the routers in the network, and can be cancelled by satisfying update messages to an aberrant host. SIFF is obvious to legacy hosts. But the benefits of this will get only to the updated hosts.

Next approach is TVA, [10] a network architecture that limits the shock of Denial of Service. This also depends on the capabilities which are received from the recipient before sending any data. This can be function at gigabit speed with product hardware.

Another DoS opposing architecture is NetFence [11]. NetFence uses secure congestion policing feedback, to allow strong congestion policing inside the network. Blockage routers update the feedback in packet headers to signal blocking, and access routers use it to police senders' traffic. Secure congestion policing feedback can be used as the capability tokens in DoS victims to reduce the unwanted traffic. NetFence gives the legal sender, its considerable share of network resources when compromised senders and receivers arrange into couple to congest a network link. For that, no information needed about state of each host.

A divide and conquer method for tracking attack source is a data dissemination architecture [12]. The main aim of this proposal is to take on the three aspects, such as attack tree construction, attack path frequency detection, and packet to path association. They use recurrence relations to state their own execution.

Another solution [13] in a high-speed network surroundings demands lightweight mechanisms for distinguishing between valid traffic and the attacker's packets. The protocol uses only available capable packet filtering mechanisms based on addresses and number of ports. This protocol changes the ports means it performs pseudo random port hopping. It defines measures for the capabilities of the attacker and for the hit rate of the protocol. Using these, it offers a novel thorough analysis of the impact of DoS on an end-to-end protocol.

Next approach [14] concentrates on the solution to make sure that genuine traffic is given a satisfactory level of quality of service. This addresses a new procedure, called port hopping where the UDP or TCP port number used by the server changes as a function of time and a shared secret. This function and secret will be only known to the server and the client. The main power of the mechanism lies in the simplification of both the detection and filtering of harmful adversary packets and that it does not require any changes to existing protocols. This port hopping technique is well matched with the UDP and TCP protocols and can be put into practice using the socket communications for the UDP protocol, and for setting up TCP communications.

An efficient solution [15] to guard against DoS attacks is to filter DoS attack requests at the earliest point as possible

say, at the web site's firewall, before consuming much of the server's resources. The primary goal of most of the defense systems will be to filter out unauthenticated clients from consuming much of the server's resources and services. Client authentication using techniques like IPSec or SSL may often necessitate alterations to the client-side software and in addition, it may have need of super user privileges at the client for deployment. Additionally, using digital signatures as in SSL, makes authentication very luxurious, thereby making the authentication process itself a viable DoS target for the attacker or adversary. This approach proposes a light weight client transparent method or technique to guard against DoS attacks with two unique features. First one is, this method can be implemented completely using JavaScript support made available by a standard client-side browser like Mozilla FireFox or Microsoft Internet Explorer. Client transparency follows from the fact that no changes to client side software are necessary, then no client-side super user privileges are required, and the last one is clients such as human beings or automated clients can browse a DoS protected website in the same way as that they browse other websites. Although operates using the client-side browser as in HTTP layer, this technique allows fast IP level packet filtering at the server's firewall and necessitates no changes to the applications hosted by the web server.

IV. CONCLUSION

In this paper, the main thing described is that about what is distributed denial of service attack. Besides that, the different types of distributed denial of service attacks are also described. Finally, different methods for mitigating distributed denial of service are also discussed here.

ACKNOWLEDGMENT

First of all thanks to God Almighty. I am very much thankful to my family members. I am thankful to our Principal, HOD, and my guide. I am thankful to my friends for helping me in doing this paper. Last but not least, I am thankful to all the people who have helped me in doing such a survey paper.

REFERENCES

[1] Zhang Fu, Marina Papatriantafilou, and Philippas Tsigas, "Mitigating Disatributed Denial of Service Attacks in Multiparty Applications in the Presence of Clock Drifts," IEEE Transactions on Dependable and Secure Computing, Vol. 9, No. 3, May/June 2012

[2] Katerina Argyraki, David R. Cheriton, "Active Internet Traffic Filtering: Real Time Response to Denial-of-Service Attacks", Distributed Systems Group, Stanford University.

[3] Xin Liu, Xiaowei Yang, Yanbin Lu, "To Filter or to Authorize: Network Layer DoS Defence Against Multimillion-node Botnets", SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.

[4] Angelos D. Keromytis, Vishal Misra, Dan Rubenstein, "SOS: Secure Overlay Services", SIGCOMM'02, August 19-23, 2002 Pittsburgh, Pennsylvania,USA.

[5] David G. Andersen, "Mayday: Distributed Filtering for Internet Services", DARPA and the Space and Naval Warfare Systems Center, San Diego, N66001-00-1-8933.

[6] Xiaoming Fu, Jon Crowcroft, "GONE: An Infrastructure Overlay for Resilient, DoS-Limiting Networking," NOSSDAV'06 Newport, Rhode Island USA.

[7] Angelos Stavrou, Angelos D. Keromytis, "Countering DoS Attacks with Stateless Multipath Overlays", CCS'05, November 7-11, 2005, Alexandria, Virginia, USA.

[8] Tom Anderson, Timothy Roscoe, David Wetherall, "Preventing Internet Denial-of-Service with Capabilities", Intel research Berkeley, IRB-TR-03-047, November, 2003.

[9] Abraham Yaar, Adrian Perrig, Dawn Song, "SIFF: A Stateless Internet Flow Filter to Mitigate DDoS Flooding Attacks", DAAD 19-02-1-0389, Carnegie Mellon.

[10] Xiaowei Yang, David Wetherall, Thomas Anderson, "A DoS-limiting Network Architecture", SIGCOMM'05, August 21-26, 2005, Philadelphia, Pennsylvania, USA.

[11] Xin Liu, Xiaowei Yang, Yong Xia, "NetFence: Preventing Internet Denial of Service from Inside Out", SIGCOMM'10, August 30-September 3,2010, New Delhi, India.

[12] M. Muthuprasanna, G. Manimaran, "Distributed divide-and-conquer techniques for effective DDoS attack defences", Iowa State University.

[13] Gal Badishi, Amir herzberg, and Idit Keidar, "Keeping Denial-of-Service Attackers in the Dark", IEEE Transactions on Dependable and Secure Computing, Vol. 4, No. 3, July-September 2007.

[14] Henry C. J. Lee, Vrizlynn L. L. Thing, "Port Hopping for Resilient Networks", IEEE 2004.

[15] Mudhakar Srivatsa, Arun Iyengar, Jian Yin, and Ling Liu, "A Client-Transparent Approach to Defend Against Denial of Service Attacks", 25th IEEE Symposium on Reliable Distributed Systems (SRDS'06).