DISTRIBUTED DENIAL OF SERVICE ATTACKS

Hardik Prajapati[1]
[1]Asst. Prof, Department of Electronics and Communication Engineering.
Gujarat, India.

Abstract: ***Distributed Denial of Service (DDoS) attacks have become a large problem for users of computer systems connected to the Internet. DDoS attackers hijack secondary victim systems using them to wage a coordinated large-scale attack against primary victim systems. As new countermeasures are developed to prevent or mitigate DDoS attacks, attackers are constantly developing new methods to circumvent these new countermeasures. In this paper, we describe DDoS attack models and propose taxonomies to characterize the scope of DDoS attacks, the characteristics of the software attack tools used, and the countermeasures available. These taxonomies illustrate similarities and patterns in different DDoS attacks and tools, to assist in the development of more generalized solutions to countering DDoS attacks, including new derivative attacks.***

Keywords: DDOS, distributed denial of service, DDOS Attack.

I. INTRODUCTION

A Denial of Service (DoS) attack is an attack with the purpose of preventing legitimate users from using a specified network resource such as a website, web service, or computer system [1]. A Distributed Denial of Service (DDoS) attack is a coordinated attack on the availability of services of a given target system or network that is launched indirectly through many compromised computing systems. The services under attack are those of the "primary victim", while the compromised systems used to launch the attack are often called the "secondary victims". The use of secondary victims in a DDoS attack provides the attacker with the ability to wage a much larger and more disruptive attack while remaining anonymous since the secondary victims actually perform the attack making it more difficult for network forensics to track down the real attacker.

II. DDoS ATTACK ARCHITECTURE

Two types of DDoS attack networks have emerged :

1. The Agent-Handler model and
2. The Internet Relay Chat (IRC) - based model

1. **The Agent-Handler model :**
The Agent-Handler model of a DDoS attack consists of clients, handlers, and agents (see Figure 1). The client is where the attacker communicates with the rest of the DDoS attack system. The handlers are software packages located throughout the Internet that the attacker’s client uses to communicate with the agents. The agent software exists in compromised systems that will eventually carry out the attack. The attacker communicates with any number of handlers to identify which agents are up and running, when to schedule attacks, or when to upgrade agents.

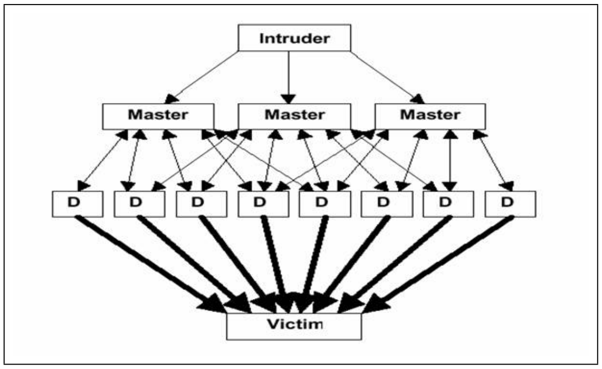

Figure 1: Agent-Handler Model

Usually, attackers will try to place the handler software on a compromised router or network server that handles large volumes of traffic. This makes it harder to identify messages between the client and handler and between the handler and agents. In descriptions of DDoS tools, the terms "handler" and "agents" are sometimes replaced with "master" and "daemons", respectively.

2. **The Internet Relay Chat (IRC) - based model :**
The IRC-based DDoS attack architecture is similar to the Agent-Handler model except that instead of using a handler program installed on a network server, an IRC (Internet Relay Chat) communication channel is used to connect the client to the agents. An IRC channel provides an attacker with additional benefits such as the use of "legitimate" IRC ports for sending commands to the agents [4]. This makes tracking the DDoS command packets more difficult. Additionally, IRC servers tend to have large volumes of traffic making it easier for the attacker to hide his presence. Another advantage is that the attacker does not need to maintain a list of the agents, since he can log on to the IRC server and see a list of all available agents [4].

In an IRC-based DDoS attack architecture, the agents are often referred to as "Zombie Bots" or "Bots". In both IRC-based and Agent-Handler DDoS attack models, we refer to the agents as "secondary victims" or "zombies".

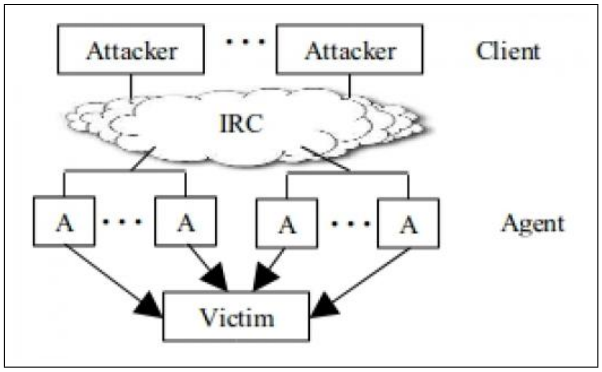

Figure 2: IRC Based DDOS Architecture

III. CHARACTERISTICS OF DISTRIBUTED DENIAL OF SERVICE ATTACK

A denial of service attack is characterized by an explicit attempt by an attacker to prevent legitimate users of a service from using the desired resources. Examples of denial of service attacks include [6].

a) Attempts to "flood" a network, thereby preventing legitimate network traffic.

b) Attempts to disrupt connections between two machines, thereby preventing access to a service.

c) Attempts to prevent a particular individual from accessing a service.

d) Attempts to disrupt service to a specific system or person.

The distributed format adds the "many to one" dimension that makes these attacks more difficult to prevent A distributed denial of service attack is composed of four elements, as shown in Figure 1 [4]. First, it involves a victim, i.e., the target host that has been chosen to receive the brunt of the attack. Second, it involves the presence of the attack daemon agents. These are agent programs that actually conduct the attack on the target victim. Attack daemons are usually deployed in host computers. These daemons affect both the target and the host computers. The task of deploying these attack daemons requires the attacker to gain access and infiltrate the host computers. The third component of a distributed denial of service attack is the control master program. Its task is to coordinate the attack. Finally, there is the real attacker, the mastermind behind the attack. By using a control master program, the real attacker can stay behind the scenes of the attack. The following steps take place during a distributed attack.

The real attacker sends an execute message to the control master program. The control master program receives the execute message and propagates the command to the attack daemons under its control. Upon receiving the attack command, the attack daemons begin the attack on the victim.

IV. DDoS ATTACK TAXONOMY

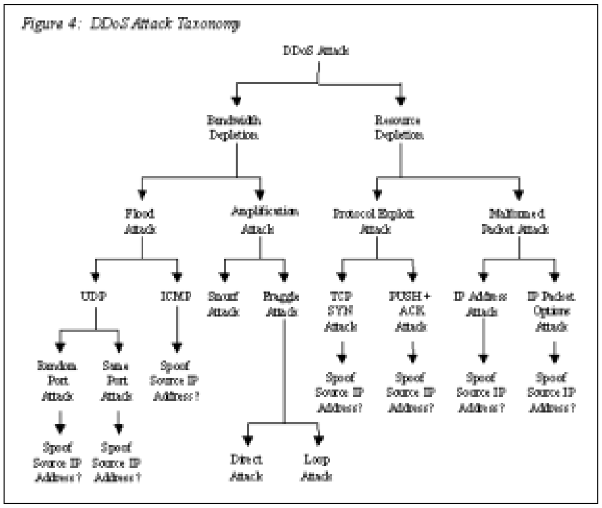

Figure 3: DDoS ATTACK TAXONOMY

There are two main classes of DDoS attacks :

1. Bandwidth Depletion and
2. Resource Depletion

Bandwidth Depletion :

Designed to flood the victim network with unwanted traffic that prevents legitimate traffic from reaching the primary victim.

Resource Depletion :

An attack that is designed to tie up the resources of a victim system making the victim unable to process legitimate requests for service.

A. Bandwidth Depletion Attacks

Bandwidth depletion attacks can be characterized as flood attacks and amplification attacks.

a) **Flood Attacks**
A flood attack involves zombies sending large volumes of traffic to a victim system, to congest the victim systems network bandwidth with IP traffic. The victim system slows down, crashes, or suffers from saturated network bandwidth, preventing access by legitimate users. Flood attacks have been launched using both UDP (User Datagram Protocol) and ICMP (Internet Control Message Protocol) packets.

In a UDP Flood attack, a large number of UDP packets are sent to either random or specified ports on the victim system. The victim system tries to process the incoming data to determine which applications have requested data.

b) **Amplification Attacks**
An amplification attack involves the attacker or the zombies sending messages to a broadcast IP address, using this to cause all systems in the subnet reached by the broadcast address to send a reply to the victim system. The broadcast IP address feature is found on most routers; when a sending system specifies a broadcast IP address

as the destination address, the routers replicate the packet and send it to all the IP addresses within the broadcast address range. In this attack, the broadcast IP address is used to amplify and reflect the attack traffic, and thus reduce the victim system's bandwidth.

B. Resource Depletion Attacks

DDoS resource depletion attacks involve the attacker sending packets that misuse network protocol communications or are malformed. Network resources are tied up so that none are left for legitimate users

a) **Protocol Exploit Attacks**

We give two examples, one misusing the TCP SYN (Transfer Control Protocol Synchronize) protocol, and the other misusing the PUSH+ACK protocol.

In a DDoS TCP SYN attack, the attacker instructs the zombies to send bogus TCP SYN requests to a victim server in order to tie up the server's processor resources, and hence prevent the server from responding to legitimate requests. The TCP SYN attack exploits the three-way handshake between the sending system and the receiving system by sending large volumes of TCP SYN packets to the victim system with spoofed source IP addresses, so the victim system responds to a non- requesting system with the ACK + SYN. When a large volume of SYN requests are being processed by a server and none of the ACK + SYN responses are returned, the server eventually runs out of processor and memory resources, and is unable to respond to legitimate users.

V. DDoS ATTACK TOOLS

DDoS attack tools include a number of common software characteristics

a) **LOIC (Low Orbit Ion Canon)**
This tool performs a DOS attack by sending UDP, TCP, or HTTP requests to the victim server.

b) **HOIC: High Orbit Ion Canon HOIC**

1. High Orbit Ion Canon HOIC is Anonymous DDOS Tool. HOIC is an Windows executable file High-speed multi-threaded HTTP Flood.
2. Simultaenously flood up to 256 websites at once.
3. Built in scripting system to allow the deployment of 'boosters', scripts.
4. Designed to thwart DDoS counter measures and increase DoS output.
5. Easy to use interface.
6. Ability to select the number of threads in an ongoing attack.
7. Ability to throttle attacks individually with three settings: LOW, MEDIUM ,and HIGH.

c) **XOIC**
XOIC is another nice DOS attacking tool. It performs a DOS attack an any server with an IP address, a user-selected port, and a user-selected protocol.

d) **TorHammer**
Tor's Hammer is a slow post dos testing tool written in Python. It can also be run through the Tor network to be anonymized. If you are going to run it with Tor it assumes you are running Tor on 127.0.0.1:9050. Kills most unprotected web servers running Apache and IIS via a single instance. Kills Apache 1.X and older IIS with 128 threads, newer IIS and Apache 2.X with 256 threads.

e) **PyLoris**
PyLoris is a scriptable tool for testing a server's vulnerability to connection exhaustion denial of service (DoS) attacks. PyLoris can utilize SOCKS proxies and SSL connections, and can target protocols such as HTTP, FTP, SMTP, IMAP, and Telnet.

VI. CONCLUSION

DDoS attacks make a networked system or service unavailable to legitimate users. These attacks are an annoyance at a minimum, or can be seriously damaging if a critical system is the primary victim. Loss of network resources causes economic loss, work delays, and loss of communication between network users. Solutions must be developed to prevent these DDoS attacks

References

[1] Mukherjee A., Datta J., Jorapur R., Singhvi R., Haloi S., Akram W. Shared disk big data analytics with apache hadoop. *High Performance Computing (HiPC) 19th International Conference*, 2012.

[2] Garlasu D., Sandulescu V., Halcu I., Neculoiu G. A big data implementation based on grid computing. *11th Roedunet International Conference (RoEduNet)*, 2013.

[3] Sagiroglu S., Sinanc D. Big data: A review. *Collaboration Technologies and Systems (CTS) International Conference*, 2013.

[4] Zhang Du. Inconsistencies in big data. *Cognitive Informatics andCognitive Computing (ICCI*CC) 12th IEEE International Conference*, 2013.

[5] `http://www-01.ibm.com/software/in/data/bigdata/`.

[6] `http://www.cloudcomputingpath.com/challenges-and-opportunities-with-big-data/`.

[7] Grosso P., de Laat C., Membrey P. Addressing big data issues in scientific data infrastructure. *Collaboration Technologies and Systems (CTS) International Conference*, 2013.

[8] Aditya B. Patel, Manashvi Birla, Ushma Nair. Addressing big data problem using hadoop and map reduce. *Engineering (NUiCONE) Nirma University International Conference*, 2012.

[9] Szczuka Marcin. Ifsa world congress and nafips annual meeting (ifsa/nafips). *Data and Knowledge Engineering 53*, 2013.

[10] Tien J.M. Big data: Unleashing information. *Service Systems and Service Management (ICSSSM) 10th International Conference*, 2013.

[11] `http://www.intel.in/content/dam/www/public/us/en/documents/white-papers/big-data-visualization-turning-big-data-into-big-insights.pdf`.

[12] `http://blogs.computerworld.com/business-intelligenceanalytics/23159/data-visualization-picture-worth-billion-bytes`.

[13] `http://smallbusiness.yahoo.com/advisor/applying-big-data-visualization-data-mining-054555947.html`.