Volume : 2 | Issue : 7 | July 2013 • ISSN No 2277 - 8160

Research Paper | Computer Science

Cyber Crime : How Distributed Denial of Service Works

Santosh Mishra Research Student of Pacific University, Under the guidance of Dr. Ashok Gaekwad

ABSTRACT

A denial of service is characterized by an explicit attempt by an attacker to prevent legitimate users from using resources. An attacker may attempt to: "flood" a network and thus reduce a legitimate user's bandwidth, prevent access to a service, or disrupt service to a specific system or a user. We describe methods and techniques used in denial of service attacks, and we list possible defences. In Recently many prominent web sites face so called Distributed Denial of Service Attacks (DDoS). While former security threats could be faced by a tight security policy and active measures like using firewalls, vendor patches etc. these DDoS are new in such way that there is no completely satisfying protection. We find that under persistent denial of service attacks, class based queuing algorithms can guarantee bandwidth for certain classes of input flows.

KEYWORDS : DOS, DDOS

INTRODUCTION:

A typical DDoS attack is an malicious behaviour to make a computer resource unavailable to its intended users. Perpetrator of DDoS attacks typically focus on sites or services on high-profile web servers such as banks and credit card payment. In a DDoS attack, the perpetrator uses a network of compromised computers, known as "zombies", to send tremendous data to the target(s) of the attack. "Bots" is a kind of software that invisibly infiltrates a computer without the owner's awareness. "Zombie" are the computers that have been contaminated by "bots", who can take over "zombie" computers secretly. Therefore, the owners of "zombies" have no idea that their computers have already become minions of the moon for a malicious force. The only indication of "zombies" for the owner might be just that the running speed is a little slower than usual, but for a common user, not a master of computer science, it's difficult to smell some big serious problem based on this kind of tiny difference.

2. CLASSIFICATION OF DOS AND DDOS ATTACKS

2.1 Attack scenario :

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 :

- attempts to "flood" a network, thereby preventing legitimate network traffic.
- attempts to disrupt connections between two machines, thereby preventing access to a service
- attempts to prevent a particular individual from accessing a service.
- attempts to disrupt service to a specific system or person.

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 :

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

The attacker must study the target's network topology and search for bottlenecks and vulnerabilities that can be exploited during the attack. Because of the use of attack daemons and control master programs, the real attacker is not directly involved during the attack, which makes it difficult to trace who spawned the attack.

2.1.1 System attacked:

Our first category is the system under attack. We can identify a number of attack points that would render the web server useless to normal clients.

A first reasonable attack point for a cracker wanting to launch a DoS is the router that connects the site hosting the web server to its ISP. This would effectively cut off all access to the website.

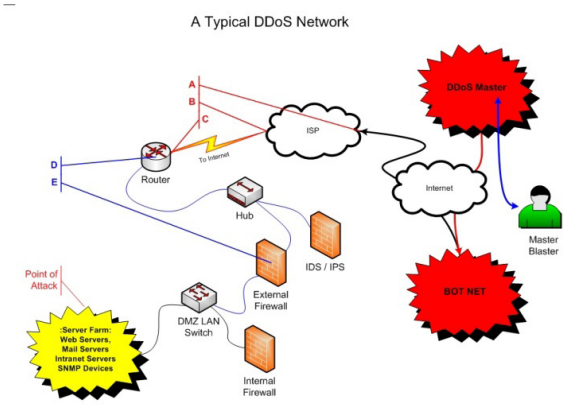
Another possible target is the firewall system, although firewalls usually tend to be (or at least should be) quite immune to direct attacks. On the other hand, the firewall is usually a bottleneck as all in and outbound traffic needs to pass through it. An attack form with a very high load might stop the firewall from working properly. As an example some misconfigured SYN flood attack overload the firewall and blocked most in- and outgoing traffic.

There may be supporting services like database servers etc. As foreign access to these servers should be blocked at the firewall, the chance of a direct DoS attack on one of this systems is fairly small.

2.1.2 Parts of systems attacked:

Some attacks may affect the hardware of a given system although such incidents are very rare. Theoretically, a network card or CPU could fail to work due to some data in a network packet it receives or processes. On the other hand, a simple overload of a 10 MBit/s Ethernet

link is an attack based on the limitation of the hardware. Attacks targeting the operating system or the TCP/IP stack of a host or router are more likely. Many attacks of this type are known, some are bugs that can be fixed, some are fundamental limitations of a protocol specification etc. On the application level there are web servers, database servers, CGI scripts etc. These could be attacked as well.



2.1.3 Bug or Overload:

In general one has to distinguish whether a DoS is a cause of a specific bug or just an overload of components that function according to their specification. Although bugs are often more severe in their effects, most of the time the vendors quickly provide fixes. All the administrators have to do is to apply them to their system in order to avoid fur-

ther attacks. Attacks that are based on an overload are typically harder to cope with. Of course you can buy new hardware, but as long as an attacker finds enough resources to use as relays in the Internet he will always bring your system to a halt. Changing the specification or protocols in order to fix the hole that allows the DoS is nearly impossible as this would often mean changing the software in millions of computers worldwide.

2.1.3 Examples :
The following examples give a little insight in different DoS attack forms. They show that our categorization is suitable for distinguishing common DoS attacks.

Cisco 7xxx routers with IOS/700 Software Version 4.1(1) or 4.1(2) have a bug where a long password given at a telnet login session leads to a buffer overflow and a subsequent reboot. According to our criteria the attacked system is a router, the part of the system under attack is the operating system and it is a bug which has been fixed in subsequent releases.

Jolt2 is an attack targeting most Microsoft Windows systems (Win 98, NT4 SP5 and SP6, Win 2000). Jolt sends a continuous stream of ICMP ECHO REPLY fragments with specially tuned fragmentation parameters to the attacked host. The exact cause of the following action is not known as the code of Microsoft's Operating Systems isn't freely available. What can be observed is that the consumption of CPU and memory resources raises to 100% which renders the system is unusable. This may be used as an attack on web servers, load balancers or even firewalls, if they run one of the named operating systems. The attacks aim at the network stack of the operating system and the reaction is caused by a bug. Fixes are available from Microsoft.

Microsoft's Internet Information Server versions 4.0 and 5.0 suffered from a so called server URL parsing bug. The decoding of escape sequences in URL strings was implemented very inefficiently. Submitting long strings with large amounts of escape characters effectively stopped the web server from working for a significant time. This attack is an attack against the web server application. It is based on an implementation bug. A fix is available from Microsoft.

2.2Methods of Denial of Service Attacks
Some widely known basic denial of service attack methods that are employed by the attack daemons.

2.2.1 Smurf attack: Smurf attacks use so-called amplifier sites in order to multiply the amount of traffic that hits the destination. These attacks send ICMP ECHO REQUEST packets with a spoofed sender address to one or several subnet broadcast addresses. The packets are received and replied by as many stations as are connected to the subnet. The replies are sent directly to the spoofed address of the attack destination. Normally this leads to congestion in the target's local network connection. Often even the lines of the ISP to which the target is connected become overloaded. This attack class strikes all kinds of targets, no matter if routers or hosts. Most of the time it is used against web servers. The effects of the attack typically put a huge stress on the network and the system (both software and hardware) that has to parse and discard the incoming packets. It is not based on a bug, although many people today think that subnets that allow a broadcast ping are a faulty configuration.

2.2.2 SYN Flood attack: It is also known as the Transmission Control Protocol (TCP) SYN attack, and is based on exploiting the standard TCP three–way handshake. The TCP three-way handshake requires a three-packet exchange to be performed before a client can officially use the service. A server, upon receiving an initial SYN (synchronize/start) request from a client, sends back a SYN/ACK (synchronize/acknowledge) packet and waits for the client to send the final ACK (acknowledge). However, it is possible to send a barrage of initial SYN's without sending the corresponding ACK's, essentially leaving the server waiting for the non-existent ACK's. Considering that the server only has a limited buffer queue for new connections, SYN Flood results in the server being unable to process other incoming connections as the queue gets overloaded.

The main principle of SYN flood attacks is to generate many half-open TCP connections by sending SYN packets to the target without replying to the following SYN-ACK packet. Most of the time the SYN packets also have spoofed source addresses of non-existent or currently inactive hosts. As today's hosts can typically handle only a limited number of half-open connections per port and discard new connection requests as long as the so called backlog queue is full, this effectively stops ordinary clients from connecting to the (web-) server. This attack form falls in the category of attacks to web servers or load-balancers which have a weakness in their TCP implementations. Part of the problem is the unlucky connection establishment procedure of the TCP protocol so it is not totally clear, weather it is an implementation bug or design.

2.2.3 UDP Flood attack: It is based on UDP echo and character generator services provided by most computers on a network. The attacker uses forged UDP packets to connect the echo service on one machine to the character generator (chargen) service on another machine. The result is that the two services consume all available network bandwidth between the machines as they exchange characters between themselves. A variation of this attack called ICMP Flood, floods a machine with ICMP packets instead of UDP packets.

2.2.4 Apache MIME flooding: There had been a number of attacks on apache web servers or Apache Sioux Attack With specially formatted HTTP requests it was possible to make the web server use up huge amounts of memory. As soon as the system started swapping out server processes or used up the whole system memory, the server was effectively dead. These attacks both target a specific web server software on the web server systems. They exploit implementation bugs.

2.3From DoS to DDoS
Major Internet websites like amazon or Yahoo tend to have Internet connections with very large bandwidth and server farms with lots of components. Furthermore they are typically protected by firewall systems that block the known attacks that are based on malformed packets like jolt2 does. The architecture of these tools is very similar and in fact some tools are modifications of others. The actual attack is carried out by so called daemons. A number of daemons is controlled by a handler and finally these handlers are activated by the attacker using client tools. The intrusion into computers onto which handlers and daemons are to be installed usually follows a simple pattern:

- A stolen account is set up as a repository for precompiled versions of scanning tools, attack (i.e. buffer overrun exploit) tools, root kits and sniffers, DDoS handler and daemon programs, lists of vulnerable and previously compromised hosts, etc.
- A scan is performed on large ranges of network blocks to identify potential targets. Targets would include systems running various services known to have remotely exploitable buffer overflow security bugs, such as wu-ftpd, RPC services for "cmsd", "statd", "ttdbserverd", "amd", etc.
- A list of vulnerable systems is then used to create a script that performs the exploit, sets up a command shell running under the root account that listens on a TCP port and connects to this port to confirm the success of the exploit.
- From this list of compromised systems, subsets with the desired architecture are chosen for the Trinoo network. Precompiled binaries of the DDoS daemons and handlers programs are created and stored on a stolen account somewhere on the Internet.
- A script is then run which takes this list of "owned" systems and produces yet another script to automate the installation process, running each installation in the background for maximum multitasking. The result of this automation is the ability for attackers to set up the denial of service network in a very short time frame and on widely dispersed systems whose true owners often don't even realize the attack.
- Optionally, a "root kit" is installed on the system to hide the presence of programs, files, and network connections. This is more important on the handler system, since these systems are the keys to the DDoS attack network.

2.3.1 Trinoo: It uses TCP to communicate between the attacker and the control master program. The master program communicates with the attack daemons using UDP packets. Trinoo's attack daemons implement UDP Flood attacks against the target victim.

2.3.2 Tribe Flood Network (TFN): It uses a command line interface to communicate between the attacker and the control master program.

Communication between the control master and attack daemons is done via ICMP echo reply packets. TFN's attack daemons implement Smurf, SYN Flood, UDP Flood, and ICMP Flood attacks.

Stacheldraht: It is based on the TFN attack. However, unlike TFN, Stacheldraht uses an encrypted TCP connection for communication between the attacker and master control program. Communication between the master control program and attack daemons is conducted using TCP and ICMP, and involves an automatic update technique for the attack daemons. The attack daemons for Stacheldraht implement Smurf, SYN Flood, UDP Flood, and ICMP Flood attacks.

Shaft: It is modelled after Trinoo. Communication between the control master program and attack daemons is achieved using UDP packets. The control master program and the attacker communicate via a simple TCP telnet connection. A distinctive feature of Shaft is the ability to switch control master servers and ports in real time, hence making detection by intrusion detection tools difficult.

3. DEFENCES AGAINST ATTACKS

Many observers have stated that there are currently no successful defences against a fully distributed denial of service attack. This may be true. Nevertheless, there are numerous safety measures that a host or network can perform to make the network and neighbouring networks more secure.

To provide protection from DoS or DDoS attacks, basic security measures are mandatory. If a running system is hacked into, no more network attacks are necessary, since local attacks (like processes consuming lots of memory or CPU time, or simply shutting down the system) are far more effective. A set of firewalls should be used to separate the interior net from the Internet.

These measures include:

3.1 Filtering Routers: Filtering all packets entering and leaving the network protects the network from attacks conducted from neighbouring networks, and prevents the network itself from being an unaware attacker. This measure requires installing ingress and egress packet filters on all routers.

3.2 Disabling IP Broadcasts: By disabling IP broadcasts, host computers can no longer be used as amplifiers in ICMP Flood and Smurf attacks. However, to defend against this attack, all neighbouring networks need to disable IP broadcasts.

3.3 Applying Security Patches: To guard against denial of service attacks, host computers must be updated with the latest security patches and techniques. For example, in the case of the SYN Flood attack, there are three steps that the host computers can take to guard themselves from attacks:

- increase the size of the connection queue,
- decrease the time-out waiting for the three-way handshake,
- and employ vendor software patches to detect and circumvent the problem.

3.4 Disabling Unused Services: If UDP echo or chargen services are not required, disabling them will help to defend against the attack. In general, if network services are unneeded or unused, the services should be disabled to prevent tampering and attacks.

3.5 Performing Intrusion Detection: By performing intrusion detection, a host computer and network are guarded against being a source for an attack, as while as being a victim of an attack. Network monitoring is a very good pre-emptive way of guarding against denial of service attacks. By monitoring traffic patterns, a network can determine when it is under attack, and can take the required steps to defend itself. By inspecting host systems, a host can also prevent it from hosting an attack on another network.

It should be used to notify the system administrators of unusual activities. The firewall rules should include some sanity checks for source and destination addresses: Packets arriving from the Internet must not have a source address originating from the interior net, and vice versa. By rejecting packets from the interior net with a non-local source address, packet spoofing becomes impossible. This technique is known as ingress and egress filtering. Even if a host is invaded by a hacker, these rules make it impossible to use that host as a platform for further attacks requiring spoofed packets (e. g. SYN-Floods, Smurf-Attacks, etc.).

3.6 Use of balance loader: It is rather difficult to defend against DoS or DDoS attacks which overload the systems network connection or local resources. These attacks usually put a heavy load on the target by making regular requests very rapidly.

It is hard to distinguish if a web server is stormed by thousands of clients, or if there is a DoS attack in progress. A simple way to force the problem of heavy load is to use a server farm together with a load balancer. This will help against small attacks, but not against a DDoS started from several hundred hosts. Furthermore, increasing the number of servers is rather expensive.

Many experts think that the only durable solution to this problem is to globally improve the security on all hosts in the Internet to take attackers the possibility to use other hosts for running daemons. This includes securing the local operating system, applying ingress- and egress filters and protecting sites with firewall systems.

3.7 First packet rejection: A simple response to SYN floods is the "first packet reject" method: The first packet sent by every host is discarded. This renders the common SYN flood attack ineffective, but all normal connection requests are delayed to the first retransmit of the SYN packet. Thus, all requests will be considerably slower. On the other hand, the SYN flood attack may be simply adapted by sending each SYN packet twice.

3.8 Moving target defence: Another technique is the so called moving target defence. Here the host under attack changes its IP address to avoid being attacked. The problem here is that the legitimate users of the system need to be informed that the IP address has changed which usually is done by updating the DNS system. Due to caching it can take up to a number of days until all clients are informed of the update. This is clearly unacceptable as the effects are as severe as any DoS attack can be. Furthermore attackers only need to incorporate DNS lookups into their tools in order to evade this protection.

REFERENCES

1. K. Hafner, M.Lyon. Where Wizards Stay Up Late.Simon & Schuster, New York, 1996. | 2. E.H. Spa_ord. The internet worm program: Ananalysis. Purdue Technical Report CSD-TR-823,Deoartment of Computer Sciences Purdue University,West Lafayette, IN. 1988. | 3. D. Seeley. A tour of the worm. Department of Computer Science, University of Utah, 1988. | 4. M. Eichin, J. Rochlis. With microscope and tweezers: An analysis of the internet virus of november 1988. Massachusetts Institute of Technology, 1988. | 5. http://www.nwfusion.com/news/2000/0209attack.html | 6. L. Stein. The world wide web security faq, version 2.0.1. http://www.w3.org/Security/Faq/ | 7. S.M Bellovin, W.R. Cheswick. Firewalls and Internet Security. Addison Wesley Longman, 1994. | 8. Dr. J.D. Howard. An analysis of security incidents on the internet 1989 - 1995. Carnegie Mellon University, Carnegie Institute of Technology,