A
Seminar Report on
Denial of Service Attack

Submission Date: October 18, 2011

Prepared by:
Ram Chandra Bhushan
M.Tech (ICT)
10IT61B07
IIT Kharagpur
Attack:

Is anything which imposes the harm on the system. It can be of two kinds in general.

Active Attack: An Attack that may change the data or harm the System.

Passive Attack: A type of Attack in which the Attacker’s goal is to obtain information. The Attack doesn’t modify or harm the system.

The Three Goals of Security:

I. Confidentiality: hiding information from unauthorized access
   - Information is exchanged over the un-trusted network.
   - Information, while exchange, should remain secret.
   - Confidentiality related to both the storage as well as transmission of information.

II. Integrity: preventing information from unauthorized modification
   - Information is always changing.
   - But the change should be made by authorized users
   - Modification: change made by unauthorized users.
   - Need techniques to ensure the integrity of data:
- preventing the modification
- detect any modification made

III. Availability: should be easily available to authorized users
- Confidentiality and Integrity should not hinder the availability of data.
- Data must be available to authorized users.
- Cryptographic mechanisms should have a small overhead.

Attacks broadly categorized in two distinct types:

**Cryptanalytic Attacks:** Applies mathematical techniques to obtain the key better than a brute force search (try all possibilities).

**Non-cryptanalytic Attacks:**

**Denial of Service Attack:** an attack on a computer or network that prevents legitimate use of its resources.
A denial-of-service attack (DoS attack) is an attempt to make a computer resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the concerted efforts of a person, or multiple people to prevent an Internet site or service from functioning efficiently or at all, temporarily or indefinitely. Perpetrators of DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root name servers. The term is generally used relating to computer networks, but is not limited to this field; for example, it is also used in reference to CPU resource management.

One common method of attack involves saturating the target machine with external communications requests, such that it cannot respond to legitimate traffic, or responds so slowly as to be rendered effectively unavailable. Such attacks usually lead to a server overload. In general terms, DoS attacks are implemented by either forcing the targeted computer(s) to reset, or consuming its resources so that it can no longer provide its intended service or obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately.

Traditional DOS

- One attacker
Distributed DOS
- Countless attackers

DoS Attacks Affect:
- Software Systems
- Network Routers/Equipments
- Servers and End-User PCs

Symptoms and manifestations
The United States Computer Emergency Readiness Team (US-CERT) defines symptoms of denial-of-service attacks to include:
- Unusually slow network performance (opening files or accessing web sites)
- Unavailability of a particular web site
- Inability to access any web site
- Dramatic increase in the number of spam emails received—(this type of DoS attack is considered an e-mail bomb)

Denial-of-service attacks can also lead to problems in the network 'branches' around the actual computer being attacked. For example, the bandwidth of a router between the Internet and a LAN may be consumed by an attack, compromising not only the intended computer, but also the entire network.

If the attack is conducted on a sufficiently large scale, entire geographical regions of Internet connectivity can be compromised.
without the attacker's knowledge or intent by incorrectly configured or flimsy network infrastructure

**Classification of DoS Attacks**

A "denial-of-service" attack is characterized by an explicit attempt by attackers to prevent legitimate users of a service from using that service. There are two general forms of DoS attacks: those that crash services and those that flood services.

A DoS attack can be perpetrated in a number of ways. The five basic types of attack are:

1. Consumption of computational resources, such as bandwidth, disk space, or processor time.
2. Disruption of configuration information, such as routing information.
3. Disruption of state information, such as unsolicited resetting of TCP sessions.
4. Disruption of physical network components.
5. Obstructing the communication media between the intended users and the victim so that they can no longer communicate adequately.

**ICMP flood**

A smurf attack is one particular variant of a flooding DoS attack on the public Internet. It relies on misconfigured network devices that allow packets to be sent to all computer hosts on a particular network via the broadcast address of the network, rather than a specific machine. The
network then serves as a smurf amplifier. In such an attack, the perpetrators will send large numbers of IP packets with the source address faked to appear to be the address of the victim. The network's bandwidth is quickly used up, preventing legitimate packets from getting through to their destination. To combat Denial of Service attacks on the Internet, services like the Amplifier Registry have given network service providers the ability to identify misconfigured networks and to take appropriate action such as filtering.

Ping flood is based on sending the victim an overwhelming number of ping packets, usually using the "ping" command from unix-like hosts (the -t flag on Windows systems has a far less malignant function). It is very simple to launch, the primary requirement being access to greater bandwidth than the victim.

Ping of death is based on sending the victim a malformed ping packet, which might lead to a system crash.

**SYN flood**

A SYN flood occurs when a host sends a flood of TCP/SYN packets, often with a forged sender address. Each of these packets is handled like a connection request, causing the server to spawn a half-open connection, by sending back a TCP/SYN-ACK packet, and waiting for a packet in response from the sender address. However, because the sender address is forged, the response never comes. These half-open connections saturate the number of available connections the server is
able to make, keeping it from responding to legitimate requests until after the attack ends.

**Teardrop attacks**

A Teardrop attack involves sending mangled IP fragments with overlapping, over-sized payloads to the target machine. This can crash various operating systems due to a bug in their TCP/IP fragmentation re-assembly code. Windows 3.1x, Windows 95 and Windows NT operating systems, as well as versions of Linux prior to versions 2.0.32 and 2.1.63 are vulnerable to this attack.

Around September 2009, a vulnerability in Windows Vista was referred to as a "teardrop attack", but the attack targeted SMB2 which is a higher layer than the TCP packets that teardrop used.

**Asymmetry of resource utilization in starvation attacks**

An attack which is successful in consuming resources on the victim computer must be either:

- carried out by an attacker with great resources, by either:
  - controlling a computer with great computation power or, more commonly, large network bandwidth
  - Controlling a large number of computers and directing them to attack as a group. A DDOS attack is the primary example of this.
- Taking advantage of a property of the operating system or applications on the victim system which enables an attack consuming vastly more of the victim's resources than the attacker's (an
asymmetric attack). Smurf attack, SYN flood, Sockstress and NAPTHA are all asymmetric attacks.

An attack may utilize a combination of these methods in order to magnify its power.

**Countermeasures for DoS Attacks:**

<table>
<thead>
<tr>
<th>Attack Level</th>
<th>Countermeasure Options</th>
<th>Example Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Level</td>
<td>Software patches, packet filtering</td>
<td>Ingress and Egress</td>
<td>Software upgrades can fix known bugs and packet filtering can prevent attacking traffic from entering a network.</td>
</tr>
<tr>
<td>Device</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OS Level</td>
<td>SYN Cookies, drop backlog connections, shorten timeout time</td>
<td>SYN Cookies</td>
<td>Shortening the backlog time and dropping backlog connections will free up resources. SYN cookies proactively prevent attacks.</td>
</tr>
<tr>
<td>Application</td>
<td>Intrusion Detection System</td>
<td>GuardDog, other vendors.</td>
<td>Software used to detect illicit activity.</td>
</tr>
<tr>
<td>Attacks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Flood</td>
<td>Replication and Load Balancing</td>
<td>Akami/Digital Island</td>
<td>Extend the volume of content under attack makes it more complicated and harder for attackers to identify services to attack</td>
</tr>
<tr>
<td>(Amplification, Oscillation)</td>
<td></td>
<td>provide content</td>
<td></td>
</tr>
</tbody>
</table>
n, Simple Flooding) and accomplish complete attacks.

| Protocol Feature | Extend protocols to support security | ITEF standard for itrace, DNSSEC | Trace source/destination packets by a means other than the IP address (blocks against IP address spoofing). DNSSEC would provide authorization and authentication on DNS information. |

**DoS Shortfalls:**

- New distributed server architecture makes it harder for one DoS to take down an entire site.
- New software protections neutralize existing DoS attacks quickly.
- Service Providers know how to prevent these attacks from affecting their networks.
- Most current IDS’s detect the current generation of tools.

**Distributed Denial of Service Attacks:**

As Defined by the World Wide Web Security FAQ: A Distributed Denial of Service (DDoS) attack uses many computers to launch a coordinated DoS attack against one or more targets. Using client/server
technology, the perpetrator is able to multiply the effectiveness of the Denial of Service significantly by harnessing the resources of multiple unwitting accomplice computers which serve as attack platforms. Typically a DDoS master program is installed on one computer using a stolen account. The master program, at a designated time, then communicates to any number of "agent" programs, installed on computers anywhere on the internet. The agents, when they receive the command, initiate the attack. Using client/server technology, the master program can initiate hundreds or even thousands of agent programs within seconds.

A distributed denial of service attack (DDoS) occurs when multiple systems flood the bandwidth or resources of a targeted system, usually one or more web servers. These systems are compromised by attackers using a variety of methods.

A system may also be compromised with a trojan, allowing the attacker to download a zombie agent (or the trojan may contain one). Attackers can also break into systems using automated tools that exploit flaws in programs that listen for connections from remote hosts. This scenario primarily concerns systems acting as servers on the web.

These collections of systems compromisers are known as botnets. DDoS tools like Stacheldraht still use classic DoS attack methods centered on IP spoofing and amplification like smurf attacks and fraggle attacks (these are also known as bandwidth consumption attacks). SYN floods (also known as resource starvation attacks) may also be used. Newer tools can use DNS servers for DoS purposes. See next section.

It is important to note the difference between a DDoS and DoS attack. If an attacker mounts an attack from a single host it would be classified as a DoS attack. In fact, any attack against availability would be classed as a Denial of Service attack. On the other hand, if an attacker uses many
systems to simultaneously launch attacks against a remote host, this would be classified as a DDoS attack.

The major advantages to an attacker of using a distributed denial-of-service attack are that multiple machines can generate more attack traffic than one machine, multiple attack machines are harder to turn off than one attack machine, and that the behavior of each attack machine can be stealthier, making it harder to track down and shut down. These attacker advantages cause challenges for defense mechanisms. For example, merely purchasing more incoming bandwidth than the current volume of the attack might not help, because the attacker might be able to simply add more attack machines.

**Architecture of DDoS:**

![Diagram of DDoS architecture]

**Widely Used DDoS Programs**

- Trinoo
- attacker uses TCP; masters and daemons use UDP; password authentication

• **Tribe Flood Network**
  - attacker uses shell to invoke master; masters and daemons use ICMP ECHOREPLY

• **TFN2K**

• **stacheldraht (barbed wire)**
  - Attacker uses encrypted TCP connection to master; masters and daemons use TCP and ICMP ECHO REPLY; rcp used for auto-update.

**Trinoo:**

• Discovered in August 1999
• First DDoS Tool widely available.
• Daemons found on Solaris 2.x systems
• Attack a system in University of Minnesota
• Victim unusable for 2 days
• Uses UDP flooding attack strategy.
• TCP connectivity between master and hosts.
• UDP connectivity between master and agents.
• Trinoo is famous for allowing attackers to leave a message in a folder called cry_baby. The file is self replicating and is modified on a regular basis as long as port 80 is active.
How it's done?

Using a compromised host, attacker compiles a list of machines that can be compromised.

- As soon as the list of machines that can be compromised has been compiled, scripts are run to compromise them and convert them into the Trinoo Masters or Daemons.
- A new script is written to automatically install the trinoo daemon on the selected systems. Some systems will fail to install, but all successful installations create the attacking network.
- One Master can control multiple Daemons.
- The Daemons are the compromised hosts that launch the actual UDP floods against the victim machine.
- The DDoS attack is launched when the attacker issues a command on the Master hosts. The Masters instruct every Daemon to start a DoS attack against the IP address specified in the command.
- Remote control to the master is set up via TCP port 27665. The master system can communicate with the agents via UDP on port 27444 and the agents send responses to the master on UDP port 31335.
- Master and Agents are password protected.
- Commands are three bit letters in binary won’t show up as strings
Password protection

- Password used to prevent administrators or other hackers to take control
- Encrypted password compiled into master and daemon using crypt()
- Default passwords
  - “l44adsl” – trinoo daemon password
  - “gOrave” – trinoo master server startup
  - “betaalmostdone” – trinoo master remote interface password
  - “killme” – trinoo master password to control “mdie” command

Login to master

- Telnet to port 27665 of the host with master
- Enter password “betaalmostdone”
- Warn if others try to connect the master
- Commands and execution:
[root@r2 root]# telnet r1 27665
Trying 192.168.249.201...
Connected to r1.router (192.168.249.201).
Escape character is '^[']'.
betaalmostdone
trinoo v1.07d2+f3+c..[rpm8d/cb4Sx/]

trinoo>

**Master and daemon**
- Communicate by UDP packets
- Command line format
  - arg1 password arg2
- Default password is “l44adsl”
- When daemon starts, it sends “HELLO” to master
- Master maintains list of daemon

**Master commands**
- **dos IP**
  - DoS the IP address specified
  - “aaa l44adsl IP” sent to each daemon
- **mdos <ip1:ip2:ip3>**
  - DoS the IPs simultaneously
- **mtimer N**
  - Set attack period to N seconds
• **bcast**
  – List all daemons’ IP

• **mdie password**
  – Shutdown all daemons

• **killdead**
  – Invite all daemons to send “HELLO” to master
  – Delete all dead daemons from the list

**Daemon commands**

• Not directly used; only used by master to send commands to daemons

• Consist of 3 letters
  – Avoid exposing the commands by using Unix command “strings” on the binary

• **aaa password IP**
  – DoS specified IP

• **bbb password N**
  – Set attack period to N seconds

• **rsz password N**
  – Set attack packet size to N bytes

**Prevention and response**

**Firewalls**

Firewalls have simple rules such as to allow or deny protocols, ports or IP addresses. Some DoS attacks are too complex for today's firewalls,
e.g. if there is an attack on port 80 (web service), firewalls cannot prevent that attack because they cannot distinguish good traffic from DoS attack traffic. Additionally, firewalls are too deep in the network hierarchy. Routers may be affected even before the firewall gets the traffic. Nonetheless, firewalls can effectively prevent users from launching simple flooding type attacks from machines behind the firewall.

Some stateful firewalls, like OpenBSD's pf(4) packet filter, can act as a proxy for connections: the handshake is validated (with the client) instead of simply forwarding the packet to the destination. It is available for other BSDs as well. In that context, it is called "synproxy".

**Switches**

Most switches have some rate-limiting and ACL capability. Some switches provide automatic and/or system-wide rate limiting, traffic shaping, delayed binding (TCP splicing), inspection and Bogon filtering (bogus IP filtering) to detect and remediate denial of service attacks through automatic rate filtering and WAN Link failover and balancing.

These schemes will work as long as the DoS attacks are something that can be prevented by using them. For example SYN flood can be prevented using delayed binding or TCP splicing. Similarly content based DoS can be prevented using deep packet inspection. Attacks originating from dark addresses or going to dark addresses can be prevented using Bogon filtering. Automatic rate filtering can work as long as you have set rate-thresholds correctly and granularly. Wan-link failover will work as long as both links have DoS/DDoS prevention mechanism.

**Routers**

Similar to switches, routers have some rate-limiting and ACL capability. They, too, are manually set. Most routers can be easily overwhelmed under DoS attack. If you add rules to take flow statistics out of the router
during the DoS attacks, they further slow down and complicate the matter. Cisco IOS has features that prevent flooding, i.e. example settings.

**Application front end hardware**

Application front end hardware is intelligent hardware placed on the network before traffic reaches the servers. It can be used on networks in conjunction with routers and switches. Application front end hardware analyzes data packets as they enter the system, and then identifies them as priority, regular, or dangerous. There are more than 25 bandwidth management vendors. Hardware acceleration is key to bandwidth management.

**IPS based prevention**

Intrusion-prevention systems (IPS) are effective if the attacks have signatures associated with them. However, the trend among the attacks is to have legitimate content but bad intent. Intrusion-prevention systems which work on content recognition cannot block behavior-based DoS attacks.

An ASIC based IPS can detect and block denial of service attacks because they have the processing power and the granularity to analyze the attacks and act like a circuit breaker in an automated way.

A rate-based IPS (RBIPS) must analyze traffic granularly and continuously monitor the traffic pattern and determine if there is traffic anomaly. It must let the legitimate traffic flow while blocking the DoS attack traffic.

**DDS based defense**

More focused on the problem than IPS, a DoS Defense System (DDS) is able to block connection-based DoS attacks and those with legitimate content but bad intent. A DDS can also address both protocol attacks
(such as Teardrop and Ping of death) and rate-based attacks (such as ICMP floods and SYN floods).

Like IPS, a purpose-built system, such as the well-known Top Layer IPS products, can detect and block denial of service attacks at much nearer line speed than a software based system.

**Blackholing and sinkholing**

With blackholing, all the traffic to the attacked DNS or IP address is sent to a "black hole" (null interface, non-existent server, ...). To be more efficient and avoid affecting your network connectivity, it can be managed by the ISP.

Sinkholing routes to a valid IP address which analyzes traffic and rejects bad ones. Sinkholing is not efficient for most severe attacks.

**Clean pipes**

All traffic is passed through a "cleaning center" via a proxy, which separates "bad" traffic (DDoS and also other common internet attacks) and only sends good traffic beyond to the server. The provider needs central connectivity to the Internet to manage this kind of service.

**Three lines of defense:**

**Attack prevention**

- before the attack
- Protect hosts from installation of masters and agents by attackers
- Scan hosts for symptoms of agents being installed
- Monitor network traffic for known message exchanges among attackers, masters, agents
• Inadequate and hard to deploy

**Attack detection and filtering**

• **Detection**
  – Identify DDoS attack and attack packets

• **Filtering**
  – Classify normal and attack packets
  – Drop attack packets

• Can be done in 4 places
  – Victim’s network
  – Victim’s ISP network
  – Further upstream ISP network

**Attack source traceback**

Identify actual origin of packet

• Without relying on source IP of packet

• 2 approaches
  – Routers record info of packets
  – Routers send additional info of packets to destination

• Source traceback cannot stop ongoing DDoS attack
  – Cannot trace origins behind firewalls, NAT (network address translators)
- More to do for reflector attack (attack packets from legitimate sources)

**Conclusion**

- DDoS attacks are complex and serious problem
  - affecting not only a victim but the victim’s legitimate clients
- DDoS defense approaches are numerous
  - need to learn how to combine the approaches to completely solve the problem
- Internet community must cooperate to counter threat
  - global deployment of defense mechanisms

**References**


2. Kargl, Frank, Joern Maier, and Michael Weber. Protecting Web Servers from Distributed Denial of Service Attacks. WWW10, May 1-5 Hong Kong. ACM 1-58113-348-0/01/0005.

http://staff.washington.edu/dittrich/misc/trinoo.analysis.txt – visited on October 1, 2002

http://staff.washington.edu/dittrich/misc/trinoo.analysis.txt – visited on October 1, 2002

http://staff.washington.edu/dittrich/misc/stacheldraht.analysis.txt – visited on October 1, 2002

http://grc.com/dos/grcdos.htm
