**🡺 DoS Attack**

A DoS attack is an attack on a computer or network that reduces, restricts, or prevents access to system resources for legitimate users. In a DoS attack, attackers flood a victim’s system with nonlegitimate service requests or traffic to overload its resources and bring down the system, leading to the unavailability of the victim’s website or at least significantly reducing the victim’s system or network performance. The goal of a DoS attack is to keep legitimate users from using the system, rather than to gain unauthorized access to a system or to corrupt data.

DoS attacks target network bandwidth or connectivity. Bandwidth attacks overflow the network with a high volume of traffic by using existing network resources, thereby depriving legitimate users of these resources. Connectivity attacks overflow a system with a large number of connection requests, consuming all available OS resources to prevent the system from processing legitimate user requests.

**🡺 DDoS Attack**

A DDoS attack is a large-scale, coordinated attack on the availability of services on a victim’s system or network resources, and it is launched indirectly through many compromised computers (botnets) on the Internet. 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.

The services under attack belong to the “primary victim,” whereas the compromised systems used to launch the attack are called “secondary victims.” The use of secondary victims in performing a DDoS attack enables the attacker to mount a large and disruptive attack while making it difficult to track down the original attacker.

🡺 **Categories of DoS/DDoS Attack Vectors**

**Volumetric Attacks:**

These attacks exhaust the bandwidth either within the target network/service or between the target network/service and the rest of the Internet to cause traffic blockage, preventing access to legitimate users. The attack magnitude is measured in bits per second (bps). Volumetric DDoS attacks generally target protocols such as the Network Time Protocol (NTP), Domain Name System (DNS), and Simple Service Discovery Protocol (SSDP), which are stateless and do not have built-in congestion avoidance features. The generation of a large number of packets can cause the consumption of the entire bandwidth on the network.

There are two types of **bandwidth depletion attacks**:

* **Flood attack:** zombies send large volumes of traffic to the victim’s systems to exhaust the bandwidth of these systems.
* **Amplification attack:** the attacker or zombies transfer messages to a broadcast IP address. This method amplifies malicious traffic that consumes the bandwidth of the victim’s systems.

The following are examples for **volumetric attack techniques**:

* **UDP Flood Attack:** The flooding of UDP packets causes the server to check repeatedly for non-existent applications at the ports. Consequently, legitimate applications become inaccessible by the system, and any attempts to access them return an error reply with an ICMP “Destination Unreachable” packet.
* **ICMP Flood Attack:** Attackers send large volumes of ICMP echo request packets to a victim’s system directly or through reflection networks. These packets signal the victim’s system to reply, and the large traffic saturates the bandwidth of the victim’s network connection, causing it to be overwhelmed and subsequently stop responding to legitimate TCP/IP requests.
* **Ping of Death Attack:** Attacker attempts to crash, destabilize, or freeze the target system or service by sending malformed or oversized packets using a simple ping command.
* **Smurf Attack:** Attacker spoofs the source IP address with the victim’s IP address and sends a large number of ICMP ECHO request packets to an IP broadcast network. This causes all the hosts on the broadcast network to respond to the received ICMP ECHO requests.
* **Pulse Wave DDoS Attack:** In pulse wave DDoS attacks, the attack pattern is periodic, and the attack is huge, consuming the entire bandwidth of target networks. Attackers send a highly repetitive train of packets as pulses to the target victim every 10 min, and the attack session lasts for approximately an hour or some days. A single pulse (300 Gbps or more) is more than enough to crowd a network pipe.
* **Zero Day Attack:** Zero-day DDoS attacks are attacks in which DDoS vulnerabilities do not have patches or effective defensive mechanisms. Until the victim identifies the threat actor’s attack strategy and deploys a patch for the exploited DDoS vulnerability, the attacker actively blocks all the victim’s resources and steals the victim’s data.

**Protocol Attacks:**

Protocol DDoS attacks exhaust resources available on the target or on a specific device between the target and the Internet. These attacks consume the connection state tables present in network infrastructure devices such as load balancers, firewalls, and application servers. No new connections will be allowed, because the device will be waiting for existing connections to close or expire. In this case, the attack magnitude is measured in packets per second (pps) or connections per second (cps).

The following are examples for **protocol attack techniques**:

* **Syn Flood Attack:** In a SYN attack, the attacker sends a large number of SYN requests to the target server (victim) with fake source IP addresses. The attack creates incomplete TCP connections that use up network resources. Normally, when a client wants to begin a TCP connection to a server, the client and server exchange the following series of messages:
* A TCP SYN request packet is sent to a server.
* The server sends a SYN/ACK (acknowledgement) in response to the request.
* The client sends a response ACK to the server to complete the session setup.

This method is a “three-way handshake.”In a SYN attack, the attacker exploits the three-way handshake method. First, the attacker sends a fake TCP SYN request to the target server. After the server sends a SYN/ACK in response to the client’s (attacker’s) request, the client never sends an ACK response. This leaves the server waiting to complete the connection.

* **SYN-ACK Attack:** This type of attack is similar to the SYN flood attack, except that in this type of flood attack, the attacker exploits the second stage of a three-way handshake by sending a large number of SYN-ACK packets to the target machine to exhaust its resources.
* **ACK & PUSH ACK Flood Attack:** During an active TCP session, ACK and PUSH ACK are the flags used to transfer information to and from the server and client machines till the session ends. Attackers send a large amount of spoofed ACK and PUSH ACK packets to the target machine, making it non-functional.
* **Fragmentation:** Attacker sends a large number of fragmented (1500+ byte) packets to a target web server with a relatively small packet rate. Since the protocol allows fragmentation, these packets are usually uninspected as they pass through network equipment such as routers, firewalls, and the intrusion detection system (IDS)/intrusion prevention system (IPS). The reassembly and inspection of these large fragmented packets consume excessive resources. Moreover, the content in the packet fragments is randomized by the attacker, which makes the reassembly and inspection consume more resources and, in turn, causes the system to crash.
* **Spoofed Session Flood Attack:** attackers create fake or spoofed TCP sessions by carrying multiple SYN, ACK, and RST or FIN packets. Attackers employ this attack to bypass firewalls and perform DDoS attacks against target networks, exhausting their network resources.
* **Multiple SYN**-ACK Spoofed Session Flood Attack: Attackers create a fake session with multiple SYN and multiple ACK packets, along with one or more RST or FIN packets.
* **Multiple ACK Spoofed Session Flood Attack:** Attackers create a fake session by completely skipping SYN packets and using only multiple ACK packets along with one or more RST or FIN packets.

**Application Layer Attacks:**

In these attacks, the attacker attempts to exploit vulnerabilities in the application layer protocol or in the application itself to prevent legitimate users from accessing the application. Attacks on unpatched, vulnerable systems do not require as much bandwidth as protocol or volumetric DDoS attacks for succeeding. In application DDoS attacks, the application layer or application resources are consumed by opening connections and leaving them open until no new connections can be made. These attacks destroy a specific aspect of an application or service and can be effective with one or a few attacking machines that produce a low traffic rate.

The following are examples for application layer attack techniques:

* **HTTP GET/POST Attack:** In an HTTP GET attack, the attacker uses a time-delayed HTTP header to hold on to an HTTP connection and exhaust web-server resources. The attacker never sends the full request to the target server. Consequently, the server retains the HTTP connection and waits, making it inaccessible for legitimate users. an HTTP POST attack, the attacker sends HTTP requests with complete headers but an incomplete message body to the target web server or application. An HTTP GET/POST attack is a sophisticated layer-7 attack that does not use malformed packets, spoofing, or reflection techniques.
* **Slowloris:** It is a DDoS attack tool used to perform layer-7 DDoS attacks to take down web infrastructure. Attacker sends partial HTTP requests to the target web server or application. Upon receiving the partial requests, the target server opens multiple connections and waits for the requests to complete. However, these requests remain incomplete, causing the target server’s maximum concurrent connection pool to be filled up and additional connection attempts to be denied.
* **UDP Application Layer Flood Attack:** Though UDP flood attacks are known for their volumetric attack nature, some application layer protocols that rely on UDP can be employed by attackers to perform flood attacks on target networks.
* UDP-based application layer protocols that attackers can employ for flooding target networks: Character Generator Protocol (CHARGEN)Simple Network Management Protocol Version 2 (SNMPv2)Quote of the Day (QOTD)
* Remote procedure call (RPC)
* SSDP  Connection-less Lightweight Directory Access Protocol (CLDAP)Trivial File Transfer Protocol (TFTP)
* Network Basic Input/Output System (NetBIOS)NTP
* Quake Network Protocol
* Steam Protocol
* Voice over Internet Protocol (VoIP)
* **Multi-vector Attack:** In multi-vector DDoS attacks, the attacker uses combinations of volumetric, protocol, and application layer attacks to take down the target system or service. The attacker quickly changes from one form of DDoS attack (e.g., SYN packets) to another (layer 7). These attacks are either launched through one vector at a time or through multiple vectors in parallel to confuse a company’s IT department, making them spend all their resources and maliciously diverting their focus.
* **Peer-to-peer Attack:** Attackers exploit flaws found in networks that use the Direct Connect (DC++) protocol, which allows the exchange of files between instant-messaging clients. This kind of attack does not use botnets. Unlike a botnet-based attack, a peer-to-peer attack eliminates the need for attackers to communicate with the clients they subvert. Here, the attacker instructs clients of large peer-to-peer file sharing hubs to disconnect from their peer-to-peer network and instead connect to the victim’s website. Consequently, several thousand computers may aggressively attempt to connect to a target website, causing a drop in the performance of the target website.
* **Permanent DoS Attack:** Permanent DoS (PDoS) attacks, also known as phlashing, purely target hardware and cause irreversible damage to the hardware. Unlike other types of DoS attacks, it sabotages the system hardware, requiring the victim to replace or reinstall the hardware. The PDoS attack exploits security flaws in a device to allow remote administration on the management interfaces of the victim’s hardware, such as printers, routers, and other networking devices.
* **Distributed Reflection DoS Attack:** A distributed reflection DoS (DRDoS) attack, also known as a “spoofed” attack, involves the use of multiple intermediary and secondary machines that contribute to a DDoS attack against a target machine or application. A DRDoS attack exploits the TCP three-way handshake vulnerability. This attack involves an attacker machine, intermediary victims (zombies), secondary victims (reflectors), and a target machine. The attacker launches this attack by sending requests to the intermediary hosts, which in turn reflect the attack traffic to the target.

**🡺 Botnet**

Bots are used for benign data collection or data mining activities, such as “web spidering,” as well as to coordinate DoS attacks. The main purpose of a bot is to collect data. There are different types of bots, such as Internet bots, IRC bots, and chatter bots. Examples for IRC bots are Supybot, Sopel, EnergyMech, and Eggdrop.A botnet (a contraction of “roBOT NETwork”) is a group of computers “infected” by bots; however, botnets can be used for both positive and negative purposes. As a hacking tool, a botnet is composed of a huge network of compromised systems. A relatively small botnet of 1,000 bots has a combined bandwidth larger than the bandwidth of most corporate systems.

**Attackers can use botnets to perform the following Attacks:**

* **DDoS attacks:** Botnets can generate DDoS attacks, which consume the bandwidth of the victim’s computers. Botnets can also overload a system, wasting valuable host system resources and destroying network connectivity.
* **Spamming:** Attackers use a SOCKS proxy for spamming. They harvest email addresses from web pages or other sources.
* **Sniffing Traffic:** A packet sniffer observes the data traffic entering a compromised machine. The sniffer allows to collect information also allows an attacker to steal information from one botnet and use it against another botnet. In other words, botnets can rob one another.
* **Keylogging:** It is a method of recording the keys typed on a keyboard, and it provides sensitive information such as system passwords.
* **Spreading new malware:** Botnets can be used to spread new bots.
* **Installing advertisement add-ons:** Botnets can be used to perpetrate a “click fraud” by automating clicks.
* **Google AdSense abuse:** Some companies permit showing Google AdSense ads on their websites for economic benefits. Botnets allow an intruder to automate clicks on an ad, producing a percentage increase in the click queue.
* **Attacks on IRC chat networks:** Also called clone attacks, these attacks are similar to a DDoS attack.
* **Manipulating online polls and games**: Every botnet has a unique address, enabling it to manipulate online polls and games.
* **Mass identity theft:** Botnets can send a large number of emails while impersonating a reputable organization such as eBay

**Scanning Methods for Finding Vulnerable Machines**

* **Random Scanning:** In this technique, the infected machine (an attacker’s machine or a zombie) probes IP addresses randomly in the target network’s IP range and checks their vulnerability. On finding a vulnerable machine, it hacks and attempts to infect the vulnerable machine by installing the same malicious code installed on it.
* **Hit-list Scanning:** Through scanning, an attacker first collects a list of potentially vulnerable machines and then creates a zombie army. Subsequently, the attacker scans the list to find a vulnerable machine. On finding one, the attacker installs malicious code on it and divides the list in half. The attacker continues to scan one half, whereas the other half is scanned by the newly compromised machine.
* **Topological Scanning:** This technique uses the information obtained from an infected machine to find new vulnerable machines. An infected host checks for URLs in the hard drive of a machine that it wants to infect.
* **Local Subnet Scanning:** In this technique, an infected machine searches for new vulnerable machines in its local network, behind a firewall, by using the information hidden in the local addresses.
* **Permutation Scanning:** In this technique, attackers share a common pseudorandom permutation list of IP addresses of all machines. The list is created using a block cipher of 32 bits and a preselected key. If a compromised host is infected during either hit-list scanning or local subnet scanning, the list is scanned from immediately after the point of the compromised host to identify new targets. If a compromised host is infected during permutation scanning, scanning restarts from a random point. If an already infected machine is encountered, scanning restarts from a new random start point in the permutation list.

**DoS Attack on GitHub**

This attack was an amplification attack using a Memcached-based approach that peaked at 1.35 Tbps. The attack originated from over a thousand different autonomous system numbers (ASNs) across tens of thousands of unique endpoints. The attack worked by abusing Memcached instances that were inadvertently accessible on the public Internet with UDP support enabled. The vulnerability due to this misconfiguration caused an amplification factor of up to 51,000, implying that up to 51 KB was sent toward the target for each byte sent by the attacker. This large amplification factor caused the devastating inflow of 1.3 Tbps data towards GitHub, interrupting its normal operations.

**DoS DDoS Attack Tools**

* **High Orbit Ion Cannon (HOIC):** It is a network stress and DoS/DDoS attack application written in BASIC language. It is designed to attack up to 256 target URLs simultaneously. It sends HTTP POST and GET requests to a computer that uses lulz-inspired GUIs.
* **Low Orbit Ion Cannon (LOIC):** it is a network stress testing and DoS attack application. LOIC attacks can be called application-based DOS attacks because they primarily target web applications. LOIC can be used on a target site to flood the server with TCP packets, UDP packets, or HTTP requests with the intention of disrupting the service.

**Prevent Potential Attacks**

* **Egress Filtering:** Egress filtering scans the headers of IP packets leaving a network. If the packets meet specifications, they can be routed out of the sub-network from which they originated. On the other hand, the packets do not reach the targeted address if they fail to meet the necessary specifications. Egress filtering ensures that unauthorized or malicious traffic never leaves the internal network.
* **Ingress Filtering:** Ingress filtering is a packet filtering technique used by many Internet Service Providers (ISPs) to prevent the source address spoofing of Internet traffic. Thus, ingress filtering can indirectly combat several types of net abuse by making Internet traffic traceable to its true source.
* **TCP Intercept:** TCP intercept is a traffic-filtering feature in routers to protect TCP servers from a TCP SYN-flooding attack, which is a kind of DoS attack. n the TCP intercept mode, a router intercepts the SYN packets sent by clients to a server and matches them with an extended access list. If a match is obtained, then on behalf of the destination server, the intercept software establishes a connection with the client. Similarly, the intercept software also establishes a connection with the destination server on behalf of the client. Once the two half connections are established, the intercept software combines them transparently. Thus, the TCP ntercept software prevents fake connection attempts from reaching the server by acting as a mediator between the server and client throughout the connection.
* **Rate Limitation:** Rate limiting is a technique used to control the rate of outbound or inbound traffic of a network interface controller. This technique effectively reduces the high volume of inbound traffic that causes a DDoS attack.

**Mitigate Attack**

* **Load Balancing:** Bandwidth providers can increase bandwidth on critical connections in case of a DDoS attack to prevent their servers from shutting down. Using a replicated server model provides additional failsafe protection. Replicated servers help in better load management by balancing loads on each server in a multiple-server architecture.
* **Throttling:** “Min-max fair server-centric router” throttles (minimum and maximum throughput controls) help users prevent their servers from shutting down. Throttling helps in preventing damage to servers by controlling the DoS traffic. This method helps routers manage heavy incoming traffic so that the server can handle it. It also filters legitimate user traffic from fake DDoS attack traffic and can be extended to throttle DDoS attack traffic while allowing legitimate user traffic for better results.
* **Drop Request:** Another method is to drop packets when the load increases. Usually, the router or server performs this task. However, before continuing with a request, the system duces the requester to drop the request by making them solve a difficult puzzle that requires a lot of memory or computing power.

**Post Attack Forensics**

* **Traffic Pattern Analysis:** During a DDoS attack, the traffic pattern tool stores post-attack data, which users analyze to identify characteristics unique to the attacking traffic. These data are helpful in updating load balancing and throttling countermeasures to enhance their efficiency and protection ability.
* **Zombie Zapper Tool:** When a company is unable to ensure the security of its servers and a DDoS attack starts, the network IDS notices the high volume of traffic, which indicates a potential problem. The targeted victim can run Zombie Zapper to stop packets from flooding the system. This tool acts as a defense mechanism against Trinoo, Tribe Flood Network (TFN), Shaft, and Stacheldraht.
* **Packet Traceback:** Packet traceback refers to tracing back attack traffic. It is similar to reverse engineering. In this method, the targeted victim works backward by tracing the packet to its source. Once the victim identifies the true source, they can take steps to block further attacks from that source by developing the necessary preventive techniques.
* **Event Log Analysis:** DDoS event logs assist in forensic investigation and the enforcement of laws, which are helpful when an attacker causes severe financial damage. Providers can use honeypots and other network security mechanisms such as firewalls, packet sniffers, and server logs to store all the events that occurred during the setup and execution of the attack.

**Defend Botnets**

* **RFC 3704 Filtering:** RFC 3704 is a basic access-control list (ACL) filter, which limits the impact of DDoS attacks by blocking traffic with spoofed addresses. This filter requires packets sourced from valid, allocated address space that is consistent with the topology and space allocation. A “bogon list” consists of all unused or reserved IP addresses that should not come from the Internet. If a packet is sourced from any of the IP addresses from the bogon list, then the packet is from a spoofed source IP, and the filter should drop it.
* **Cisco IPS Source IP Reputation Filtering:** Reputation services help in determining whether an IP or service is a source of threat. Cisco Global Correlation, a new security capability of Cisco IPS 7.0, uses immense security intelligence. The Cisco SensorBase Network contains information about all known threats on the Internet, such as botnets, malware outbreaks, dark nets, and botnet harvesters. The Cisco IPS makes use of this network to filter DoS traffic before it damages critical assets.
* **Black Hole Filtering:** Black-hole filtering is a common technique to defend against botnets and, thus, to prevent DoS attacks. Black holes refer to network nodes wherein incoming traffic is discarded or dropped without informing the source that the data did not reach the intended recipient. Undesirable traffic can be dropped before it enters a protected network with a technique called remotely triggered black-hole (RTBH) filtering. As this is a remotely triggered process, this filtering must be performed in conjunction with the ISP. It uses Border Gateway Protocol (BGP) host routes to route traffic to the victim’s servers to a “null0” next hop.
* **DDoS Prevention Offerings from IPS or DDoS Service:** This method is effective in preventing IP spoofing at the ISP level. Here, the ISP scrubs/cleans traffic before allowing it to enter a user’s Internet link. Because this service runs in the cloud, DDoS attacks do not saturate the Internet links.

**DoS/DDoS Protection at ISP Level**

One of the best ways to defend against DoS attacks is to block them at the gateway. This task is performed by the contracted ISP. ISPs offer a “clean pipes” service-level agreement that provides an assured bandwidth of genuine traffic, rather than the total bandwidth of all traffic. Most ISPs simply block all requests during a DDoS attack, denying even legitimate traffic from accessing the service.

ISPs offer in-the-cloud DDoS protection for Internet links to avoid saturation due to an attack. This type of protection redirects attack traffic to the ISP during an attack. Administrators can request ISPs to block the original affected IP and move their site to another IP after performing DNS propagation.

**Enabling TCP Intercept on Cisco IOS Software**

TCP intercept can be enabled by executing the commands given in the below table in the global configuration mode.

* **access-list access-list-number {deny | permit} tcp any destination destination-wildcard**

Defines an IP extended access list

* **ip tcp intercept list access-list-number**

Enables TCP intercept

An access list achieves three purposes:

* Interception of all requests
* Interception of only requests originating from specific networks
* Interception of only requests destined for specific servers

In the active intercept mode, the Cisco IOS software actively intercepts all inbound connection requests (SYN) and replies with a SYN-ACK on behalf of the server, following which it waits for an acknowledge (ACK) from the client. On receiving the ACK from the client, the server sends the original SYN, and the software makes a three-way handshake with the server. Once the three-way handshake is complete, the two half connections are linked.

In the passive watch mode, the user sends connection requests that pass through the server, but they need to wait until the connection is established. If connection requests fail to establish within 30 s, the software sends a reset request to the server to clear its state.

Below command to set the TCP intercept mode in the global configuration mode.

**ip tcp intercept mode {intercept | watch}**