🧠 Lecture 21 – Defense Against Vulnerabilities and DoS/DDoS Attacks

---

🔹 1. What Happens After Exploiting a Vulnerability

In the previous lecture, we learned how to use **Metasploit** to exploit vulnerabilities and gain access to a target machine.
Now, in this lecture, we'll learn **how to protect systems** from such attacks.

When a vulnerability exists in a system:

- Attackers can use exploits (like Metasploit modules) to take control of that machine.
- To **prevent** this, we must **secure the system**, **harden configurations**, and **reduce attack surface**.

🔹 2. How to Protect a System from Exploitation

1️⃣ Change Default Ports

- Attackers often scan for **well-known ports** (like 21, 22, 80, 445, 3389).
- Changing default ports can **reduce attack probability** since many automated exploits target default ports.

📌 **Example:**

- Change SSH from **port 22 → 2222**
- Change RDP from **3389 → 3390**

➡️ It doesn't make you completely safe but adds a **layer of security (obscurity).**

2️⃣ Keep Operating System and Software Updated

- Many exploits (like **EternalBlue**) target outdated systems such as **Windows 7 or 8**.
- Regularly install **security patches** to fix vulnerabilities.
- Use **automatic updates** to avoid missing critical fixes.

📌 Example:
Windows Update → Security → Check for updates.

🔹 3. Defensive Strategies Against Metasploit-like Attacks

To protect against advanced frameworks such as **Metasploit**, we need a **multi-layered defense strategy**:

🧱 1. Regular Vulnerability Assessment

- Perform periodic **scans** of your systems using tools like **Nessus**, **OpenVAS**, or **Qualys**.
- Identify and fix weaknesses **before attackers do**.

🧩 2. Patch Management

- Maintain an updated list of all software and systems.
- When a **vendor releases a patch**, test it and **apply immediately**.
- Patch management tools: **ManageEngine**, **SolarWinds**, **Microsoft WSUS**.

🌐 3. Network Segmentation

- Divide the network into smaller segments (e.g., HR, Finance, R&D).
- Apply **access control** between segments using **firewalls or VLANs**.
- Even if one network is compromised, the attacker can't reach others easily.

🛡️ 4. IDS and IPS (Intrusion Detection & Prevention Systems)

- **IDS (Detection):** Monitors network traffic and detects suspicious activities.
- **IPS (Prevention):** Blocks malicious traffic in real time.
- Keep your IDS/IPS **rules updated** regularly.

📌 Examples:

- **Snort**, **Suricata**, **Cisco Firepower**, **Palo Alto Threat Prevention**.

🧠 5. Behavioral Analysis

- Modern attacks evolve, so use **AI-based monitoring** systems that analyze behavior.
- If a system suddenly sends too many packets or connections, it's flagged as **anomalous**.
- Tools: **Darktrace**, **CrowdStrike Falcon**, **Microsoft Defender 365**.

🔹 4. DoS and DDoS Attacks

⚔️ What is a DoS Attack (Denial of Service)?

- A **DoS attack** is when one system sends a flood of requests or packets to another system to **overload** it.
- This makes the target **slow or completely unavailable** for legitimate users.

📌 Example:
A single attacker floods a website with millions of HTTP requests.

⚔️ What is a DDoS Attack (Distributed Denial of Service)?

- Similar to DoS, but the attack comes from **multiple machines** (called **bots or zombies**).
- These compromised systems form a **botnet** controlled by an attacker.

- They send huge volumes of traffic to the target at the same time.

📌 Example:
Thousands of infected computers attack a web server, making it unreachable for normal users.

🧠 **How DDoS Works**

1. Attacker infects many systems with **malware or bots**.
2. These bots connect to a **command-and-control server**.
3. When commanded, all bots send traffic (HTTP requests, pings, etc.) to the target.
4. The target server becomes **overwhelmed** and stops responding.

🔹 **5. Performing a DoS/DDoS Simulation (in Lab)**

⚠️ **Note:** These commands should be used only in a **controlled lab environment**, not on real networks.

Tool: hping3

- **hping3** is a network tool used to **craft and send packets** (TCP, UDP, ICMP) to a target.
- It's used to simulate **DoS/DDoS attacks** for testing and learning.

🧩 **Basic Commands and Explanation**

Command	Description
hping3 -1 -c 1 <target IP>	Sends 1 ICMP (ping) packet to the target.
hping3 -1 -c 6 -i 5 <target IP>	Sends 6 ICMP packets, one every 5 seconds.
hping3 -1 --fast <target IP>	Sends packets continuously at a fast rate.
hping3 -1 --faster <target IP>	Sends packets faster than normal rate.

Command	Description
hping3 -1 --flood <target IP>	Floods the target with continuous packets (used for DoS).
hping3 -1 -a 192.168.1.60 -c 3 <target IP>	**IP spoofing** – sends fake packets pretending to be from another IP.
hping3 -1 --rand-source -c 3 <target IP>	Sends packets with random fake IP addresses (spoofing).
hping3 -S -c 3 -p 80 <target IP>	Sends TCP SYN packets to port 80 (HTTP) instead of ICMP.

🧠 Key Concepts Behind These Commands

- -1 → ICMP protocol (ping packets).
- -S → TCP SYN flag (used for TCP handshake attacks).
- -c → Number of packets to send.
- -i → Interval between each packet.
- -p → Port number to target.
- --flood → Sends infinite packets, maximum speed flood.
- -a → Spoof source IP.
- --rand-source → Random spoofing of source IPs.

🔬 Observation Using Wireshark

- Open Wireshark on the target machine.
- You will see incoming packets (ICMP or TCP SYN).
- You can confirm the flood or spoofed requests visually.

🔹 6. Blocking ICMP (Ping) Packets

- Since **ICMP** packets are used in ping-based DoS attacks, you can **block ICMP** traffic in the firewall.
- But note: **You cannot easily block TCP SYN packets**, since they are part of legitimate communication.

📌 In Windows Firewall or Linux iptables, you can create a rule:

iptables -A INPUT -p icmp --icmp-type echo-request -j DROP

This drops incoming ping requests.

🔹 7. DDoS Prevention and Mitigation Techniques

Since it's almost impossible to stop all DDoS attacks completely, we **reduce impact** through mitigation strategies.

🧱 1. Strong Firewall Rules

- Configure firewalls to **detect and block spoofed IPs**.
- If packets are **incoming but no response is returning**, it may be a **spoofed request** → block them.

⚖️ 2. Load Balancing

- Distribute traffic among **multiple servers**.
- Even if one server is attacked, others can continue serving users.

☁️ 3. Use Cloud-based DDoS Protection

- Services like **Cloudflare**, **AWS Shield**, or **Akamai** absorb huge amounts of attack traffic.
- These systems use **content delivery networks (CDNs)** to route and filter malicious packets.

🕸️ 4. Rate Limiting

- Limit the number of requests a single IP can send per second.
- This helps reduce traffic floods from bots.

⚙️ 5. IDS/IPS and Behavioral Detection

- Detect unusual traffic spikes or patterns (like same source sending thousands of requests).
- Automatically **block or throttle** suspicious IPs.

💡 6. Network Redundancy

- Use **multiple servers in different locations (multi-region deployment)**.
- If one location is hit by DDoS, other regions can still serve legitimate users.

🚨 7. DDoS Mitigation Tools

Some widely used DDoS protection tools/services:

- **Cloudflare DDoS Protection**
- **Akamai Kona Site Defender**
- **AWS Shield Advanced**
- **Arbor Networks APS**
- **Imperva Incapsula**

🔹 8. Summary Table

Topic	Description
DoS Attack	Single machine attacks another machine by flooding it with traffic.
DDoS Attack	Multiple machines (botnet) attack one target simultaneously.
hping3	Tool used to simulate and send packets (ICMP, TCP, UDP).
ICMP	Ping protocol (can be blocked in firewall).
TCP SYN Flood	Attack using TCP handshake process.
Spoofing	Hiding or faking real IP address during attack.
Defense Tools	Firewall, IDS/IPS, Cloudflare, Load Balancer.
Prevention Techniques	Patch updates, network segmentation, rate limiting, behavioral analysis.

✅ 9. Key Takeaways

1. Always **update your system** and software to fix vulnerabilities.
2. Use **strong firewalls** and **IDS/IPS systems**.
3. Regularly perform **vulnerability assessments** and **patch management**.
4. For DDoS defense:
    - Use **load balancers**, **multiple servers**, and **cloud protection**.
    - **Monitor traffic** for unusual spikes.
    - Implement **rate limiting and spoof detection**.
5. Security is not one-time — it's a **continuous process** of monitoring, patching, and improving.