**Ethical Hacking**

1. **Why Ethical Hacking?** Ethical hacking is the process of testing and securing systems by identifying vulnerabilities before malicious hackers can exploit them. It helps organizations prevent cyberattacks like data breaches, ransomware, or financial fraud. Ethical hackers work with permission and use legal methods to strengthen the organization's security posture. They play a critical role in ensuring data confidentiality, integrity, and availability.
2. **Types of Attacks and Attack Vector Types?** Types of attacks include **DoS (Denial of Service)**, which overwhelms a system to make it unavailable; **DDoS (Distributed DoS)**, where multiple sources attack simultaneously; **SQL Injection**, where attackers manipulate database queries to access sensitive data; **Phishing**, which tricks users into providing confidential information; **Malware**, which includes viruses, worms, and ransomware; **XSS (Cross-Site Scripting)**, which injects malicious scripts into web applications; and **Man-in-the-middle (MITM)**, where attackers intercept communication.  
    **Attack vectors** refer to the methods used to carry out these attacks, such as emails with malicious attachments, compromised websites, infected USB devices, or phishing links sent via text or social media.
3. **What is Cross-Site Scripting (XSS)?** Cross-Site Scripting is a vulnerability in web applications where attackers inject malicious scripts that execute in the user's browser. These scripts can steal sensitive information like cookies, session tokens, or passwords. There are three main types of XSS: **Stored XSS**, where the malicious script is saved on the server; **Reflected XSS**, where the script is temporarily embedded in a URL; and **DOM-based XSS**, which manipulates the Document Object Model to execute the script. Preventing XSS involves input validation, output encoding, and Content Security Policies (CSP).
4. **What is SQL Injection?** SQL Injection occurs when attackers input malicious SQL commands into a vulnerable application, allowing them to manipulate the database. They can retrieve sensitive data, delete or modify records, or even gain administrative access. For example, a login form that doesn’t sanitize inputs may allow an attacker to bypass authentication by inserting commands like ' OR '1'='1. Preventing SQL Injection involves using parameterized queries, stored procedures, and strict input validation.
5. **Hacking Phases?** Hacking is typically divided into five phases:

* **Reconnaissance**: Gathering information about the target system using public or private methods.
* **Scanning**: Actively identifying open ports, services, and vulnerabilities.
* **Gaining Access**: Exploiting vulnerabilities to gain unauthorized entry.
* **Maintaining Access**: Using backdoors or other techniques to ensure persistent access.
* **Covering Tracks**: Deleting logs and hiding activity to avoid detection. These phases are used by both ethical hackers and malicious attackers, but ethical hackers aim to secure systems rather than harm them.

1. **Scanning?** Scanning is the process of actively probing a network or system to identify live hosts, open ports, running services, and potential vulnerabilities. Tools like **Nmap** and **Nessus** are commonly used for scanning. This step helps ethical hackers understand the attack surface and prioritize vulnerabilities based on their criticality. Scanning is typically followed by detailed analysis to plan the next steps in securing the system.
2. **Footprinting?** Footprinting is the process of collecting preliminary information about a target, such as domain names, IP addresses, and email addresses. This is done using public tools like **WHOIS**, search engines, and social media. Footprinting helps hackers map the network and identify weak points for further analysis or attacks.
3. **Footprinting Countermeasures?** To prevent footprinting, organizations should avoid publishing sensitive details online, use firewalls to block unnecessary traffic, implement strong authentication mechanisms, and monitor domain or DNS queries for unusual activity. Disabling unused ports and services is also effective.
4. **Network Scanning?** Network scanning identifies active devices, open ports, and services running in a network. It helps ethical hackers locate potential vulnerabilities like outdated software, default credentials, or weak encryption. Tools such as **Nmap**, **Angry IP Scanner**, and **Zenmap** are used for this purpose.
5. **Reconnaissance?** Reconnaissance is the first phase of hacking, focusing on information gathering. It can be passive (e.g., using search engines and publicly available data) or active (e.g., performing scans to find vulnerabilities). Reconnaissance helps attackers or ethical hackers understand the target’s structure and plan their next moves.
6. **Scanning Networks, Enumeration, and Sniffing?** **Scanning** involves detecting open ports, services, and vulnerabilities.  
    **Enumeration** is extracting detailed information like usernames, shared resources, or running services.  
    **Sniffing** captures data packets on a network to analyze sensitive information like credentials. Tools like **Wireshark** and **tcpdump** are used for sniffing.
7. **Types of Stealth Scans?**

* **SYN Scan**: A half-open scan that sends SYN packets but doesn’t complete the handshake.
* **NULL Scan**: Sends packets with no flags set, often bypassing firewalls.
* **FIN Scan**: Sends packets with the FIN flag, used to evade detection.
* **Xmas Scan**: A scan with all flags set, often used to test firewall rules.

1. **Vulnerability Scanning?** Vulnerability scanning is an automated process to detect security flaws in systems. Tools like **Nessus**, **Qualys**, and **OpenVAS** identify misconfigurations, outdated software, and unpatched vulnerabilities. Regular scanning helps organizations address weaknesses before they are exploited.
2. **Proxy Servers?** Proxy servers act as intermediaries between users and the internet. They mask users’ IP addresses, ensuring anonymity and enhancing security. Proxies can also filter content, control web access, and cache frequently accessed data to improve performance.
3. **Anonymizers?** Anonymizers are tools or services that hide a user’s identity online. They route internet traffic through secure servers, preventing tracking by ISPs or hackers. Common examples include VPNs and Tor.
4. **IP Spoofing and Its Countermeasures?** IP spoofing involves forging the source IP address in packets to impersonate a trusted device. This can bypass firewalls or gain unauthorized access. Countermeasures include implementing packet filtering, secure authentication protocols, and using intrusion detection/prevention systems (IDS/IPS).
5. **Enumeration Techniques?** Enumeration extracts detailed information about systems and networks. Common techniques include **NetBIOS enumeration** for shared resources, **SNMP enumeration** for network management data, **LDAP queries** for directory information, and **DNS enumeration** to gather domain-related data.
6. **Types of Enumeration?** **Active Enumeration** involves direct interaction with systems, such as attempting logins. **Passive Enumeration** gathers data indirectly, like monitoring DNS queries or social media activity.
7. **What Do You Mean by Obfuscation?** Obfuscation makes code or data unreadable to unauthorized users by altering or encrypting it. It is commonly used to protect intellectual property, prevent reverse engineering, or hide malware.
8. **Fragmentation?** Fragmentation splits data packets into smaller parts, making it harder for firewalls or intrusion detection systems to analyze and block the payload. It is often used in advanced stealth attacks.
9. **Worms, Viruses, Trojans?** These are types of malicious software designed to harm systems:

* **Worms**: Self-replicating malware that spreads across networks without requiring human interaction. For example, worms exploit vulnerabilities in network services to propagate.
* **Viruses**: Attach themselves to executable files and require user action to activate. Once activated, they can delete data, corrupt files, or spread to other systems.
* **Trojans**: Malicious programs disguised as legitimate software. For example, a fake antivirus program that installs backdoors into the system. Unlike worms and viruses, Trojans don’t replicate themselves but often enable remote access for attackers.

1. **Types of Worms?**

* **Email Worms**: Spread via infected email attachments or links. They rely on users opening the email or attachment.
* **File-sharing Worms**: Propagate through shared files on peer-to-peer networks or removable media.
* **Network Worms**: Exploit vulnerabilities in network protocols to infect connected systems automatically, without user action.

1. **Preventing Malware Attacks?** Preventing malware attacks involves implementing robust security practices:

* Use updated antivirus and antimalware software to detect and block threats.
* Enable firewalls to block unauthorized access.
* Regularly patch and update systems and applications to fix vulnerabilities.
* Avoid opening unknown email attachments, downloading files from untrusted sources, or clicking suspicious links.
* Implement application whitelisting to control what software can run.

1. **Steganography?** Steganography is the practice of hiding data within other media, such as images, audio files, or videos, to conceal its existence. Unlike encryption, which makes data unreadable, steganography embeds data in a way that appears innocuous. For example, an image file may contain hidden text within its pixel data. Tools like **Steghide** and **OpenStego** are commonly used. It’s often used for covert communication or smuggling sensitive data.
2. **Types of Attacks: DoS/DDoS?**

* **DoS (Denial of Service)**: Overloads a target system with requests, causing it to become unavailable to legitimate users. For example, sending excessive ICMP (ping) requests can crash a server.
* **DDoS (Distributed Denial of Service)**: Similar to DoS but launched from multiple compromised systems (a botnet). This makes it harder to block and more damaging.  
   Prevention measures include rate-limiting, load balancers, and DDoS protection services like Cloudflare.

1. **SQL Injection (Detailed)**

* **What is SQL Injection?** A vulnerability where attackers manipulate SQL queries through user inputs to access or modify a database. For example, inserting ' OR '1'='1 into a login field could bypass authentication.
* **Threats**: Attackers can steal sensitive data, delete or alter records, or even escalate privileges to gain full control of the database.
* **Types**:
  + **Error-based**: Relies on database error messages to extract data.
  + **Union-based**: Combines multiple queries to retrieve additional data.
  + **Blind**: Exploits the application without visible error messages, often using true/false conditions.
  + **Time-based**: Uses time delays in responses to infer information.
* **Prevention**: Use parameterized queries (e.g., prepared statements), sanitize inputs to remove special characters, implement least privilege access for database users, and monitor logs for suspicious queries.

1. **What is Session Hijacking?** Session hijacking occurs when an attacker steals a user's session ID (token) to gain unauthorized access to their account or data. Sessions are used to maintain user authentication across requests. For example, if an attacker intercepts an unencrypted session token from a public Wi-Fi, they can impersonate the user.
2. **Why is Session Hijacking Successful?** Session hijacking succeeds because of weak session management, such as predictable session IDs, unencrypted transmission of tokens, or prolonged session expiration times. Attackers exploit these weaknesses to impersonate users.
3. **Session Hijacking Techniques?**

* **XSS (Cross-Site Scripting)**: Stealing session cookies by injecting malicious scripts into a website.
* **Session Fixation**: Forcing a user to use a known session ID, which the attacker then takes over.
* **MITM (Man-in-the-Middle)**: Intercepting session tokens during communication between a client and a server.

1. **Session Hijacking Process?** The process typically involves:

**Identifying an active session between a client and a server.**  
The attacker begins by locating an active session between a user (client) and a server. This session is typically maintained using a session ID, which is a unique identifier stored in cookies or transmitted via URL parameters. Attackers monitor network traffic or use reconnaissance tools to identify these sessions. Public Wi-Fi networks or unsecured HTTP websites are common targets since session data is transmitted in plain text, making it easier to intercept.

**Stealing session tokens using methods like sniffing or XSS.**After identifying a session, the attacker attempts to steal the session token (or ID) used to authenticate the user. Common techniques include:

* Sniffing: Capturing unencrypted network traffic using tools like Wireshark to extract session tokens. This is particularly effective in environments with weak or no encryption, such as public Wi-Fi.
* XSS (Cross-Site Scripting): Injecting malicious scripts into a vulnerable website to steal session cookies directly from the user's browser. For instance, an XSS payload might send the user's session cookie to the attacker's server.
* Man-in-the-Middle (MITM) Attacks: Intercepting the communication between the client and server to extract session tokens or inject malicious data.

**Using the stolen tokens to impersonate the victim and access their account or data.**

With the session token in hand, the attacker bypasses authentication by presenting the stolen token to the server as if they are the legitimate user. Since the server only checks the session token's validity, it treats the attacker as the rightful account owner.

This allows the attacker to perform actions on behalf of the victim, such as viewing personal data, sending messages, making transactions, or even changing account settings.

1. **Types of Session Hijacking?**

* **Active Hijacking**: Intercepting and modifying the communication between the client and server in real time.
* **Passive Hijacking**: Monitoring the session to gather information without altering it.

1. **Session Hijacking Countermeasures?** To prevent session hijacking:

* Use **HTTPS** for secure communication.
* Implement strong, random session IDs and regenerate them upon login or privilege changes.
* Use secure, HttpOnly cookies to store session tokens.
* Set short session expiration times and enforce idle timeouts.
* Use multi-factor authentication to add an extra layer of security.

1. **Wireless Network Hacking?** Wireless network hacking involves exploiting vulnerabilities in Wi-Fi networks to access sensitive data or connected devices. Common attacks include:

* **WEP/WPA Cracking**: Exploiting weak encryption protocols like WEP or outdated WPA.
* **Rogue Access Points**: Setting up fake Wi-Fi networks to intercept user traffic.
* **Deauthentication Attacks**: Forcing devices off a network to capture handshake packets for cracking.  
   Tools like **Aircrack-ng**, **Kismet**, and **Wireshark** are often used. Securing Wi-Fi requires strong passwords, WPA3 encryption, and disabling WPS.

1. **Cloud Computing Security?** Cloud computing security focuses on protecting data, applications, and services hosted in the cloud. Key components include:

* **Encryption**: Securing data at rest and in transit using algorithms like AES-256.
* **Identity Access Management (IAM)**: Controlling user access with role-based permissions.
* **Monitoring**: Using tools like AWS CloudTrail or Azure Security Center to detect and respond to threats.
* **Regular Backups**: Ensuring data is backed up to recover from breaches or ransomware.

1. **Cryptography?** Cryptography ensures secure communication and data protection by encrypting information into unreadable formats. It involves:

* **Symmetric Encryption**: Uses a single key for encryption and decryption (e.g., AES).
* **Asymmetric Encryption**: Uses a public key for encryption and a private key for decryption (e.g., RSA).
* **Hashing**: Converts data into fixed-length values, ensuring integrity (e.g., SHA-256).  
   Cryptography protects data from unauthorized access, ensures integrity, and authenticates users or systems during transmission or storage.

**36. Penetration Testing (Pen Testing)**  
Penetration Testing, or Pen Testing, is a simulated cyberattack conducted by ethical hackers to assess the security of a system, application, or network. The primary goal is to identify vulnerabilities that attackers could exploit and provide recommendations for mitigation.

**37. ARP Poisoning (ARP Spoofing)**

ARP (Address Resolution Protocol) Poisoning, also known as ARP Spoofing, is a type of cyberattack where a malicious actor sends fake ARP messages onto a local network. These messages associate the attacker's MAC (Media Access Control) address with the IP address of another device (usually the gateway or another target). As a result, network traffic meant for the legitimate device is redirected to the attacker, allowing them to intercept, modify, or even block communications.