Module 6: - Web Application, Windows, and Linux security

6.1 Types of Audits in Windows Environment
6.2 Server Security, Active Directory (Group Policy), Anti-Virus, Mails, Malware
6.3 Endpoint protection, Shadow Passwords, SUDO users, etc.
6.4 Web Application Security: OWASP, Common Issues in Web Apps, what is XSS, SQL injection, CSRF, Password Vulnerabilities, SSL, CAPTCHA, Session Hijacking, Local and Remote File Inclusion, Audit Trails, Web Server Issues, etc.

Q.1 What are different types of audits in the Windows Environment.

There are two main categories of audits in a Windows environment: basic and advanced.

Basic Security Audits in Windows Environment

Audit Category	Subcategory	Description	Example Events
Account Logon Events	Login/Logouts	Monitor user access attempts	Successful and failed logins, network connections
	Account Management	Track changes to user accounts	User creation, deletion, password changes
Directory Service Access	Object Access	Monitor access to Active Directory objects	Access to user accounts, groups, policies
Logon Events	General Login/Logouts	Broader view of login/logouts	Similar to Account Logon Events, with additional details
Object Access	File/Folder/Registry Access	Monitor attempts to access sensitive data	Read, write, modify attempts on files, folders, registry keys
Policy Change	Security Policy Modifications	Track changes to security settings	User rights, audit policies, system-wide policies changes
Privilege Use	Elevated Privileges	Monitor use of specific user privileges	Attempts to change system time, load drivers
Process Tracking	Program Activity	Monitor application activity	Program activation, termination, object/process interaction
System Events	System Health and Security Incidents	Track important system events	Server shutdowns, restarts, security related events

Advanced Security Audits in Windows Environment

Audit Category	Description	Example Events
File Access Tracking	Monitors specific file access attempts, including read, write, and execute operations.	Attempts to open, create, modify, or delete specific files or folders.
Registry Access Tracking	Tracks detailed access to specific registry keys and values.	Attempts to read, write, or modify sensitive registry settings.
Network Access Tracking	Provides detailed information about network connections, traffic flows, and resource access.	Successful and failed connections, port usage, data transfer details.
PowerShell Logging	Monitors PowerShell commands executed on the system, providing insight into administrative activities.	Logs of all PowerShell commands, scripts, and modules used.

Q.2) Write a short note on Server Security

Server security is the practice of protecting servers, the backbone of modern computing, from unauthorized access, misuse, data breaches, and attacks. It encompasses a wide range of measures implemented at various levels to safeguard sensitive information, maintain server integrity, and ensure business continuity.

Key Security Layers in Server Security

Security Layer	Key Measures
Physical Security	Controlled access, surveillance systems, environmental controls, secure hardware disposal
Operating System Hardening	Removal of unnecessary services, secure configuration, strong passwords, regular patching, encryption
Network Security	Firewalls, intrusion detection/prevention systems (IDS/IPS), network segmentation, secure remote access
Application Security	Secure coding practices, regular updates and patching, vulnerability scanning, input validation
Data Security	Access controls, encryption, data loss prevention (DLP), regular backups, secure storage
Monitoring and Logging	Audit trails, intrusion detection systems (IDS), security information and event management (SIEM)
Backups and Recovery	Regular backups, offsite storage, disaster recovery plans

Importance of Server Security:

- Data Protection: Prevent data breaches and unauthorized access to sensitive information, safeguarding privacy and compliance.
- Business Continuity: Ensure uninterrupted operations and prevent costly downtime caused by attacks or disruptions.
- Financial Protection: Avoid financial losses from data breaches, fines, and reputational damage.
- Compliance: Meet regulatory requirements and industry standards for data protection and security.

Q.3) What is Active Directory ? Explain Group Policy in Active Directory

Active Directory (AD):

Centralized Directory Service: A core component of Windows Server that provides centralized management and authentication for users, computers, and other resources within a network. Hierarchical Structure: Organizes objects (users, computers, groups, etc.) into a logical, hierarchical structure, making it easier to manage and control access.

Group Policy:

- Centralized Configuration Management: A powerful feature of Active Directory that allows administrators to define and apply specific configurations and settings to users and computers throughout the domain.
- Enforces Consistency and Compliance: Ensures that systems adhere to organizational policies and standards, maintaining security and efficiency.
- Scope and Targeting: Group Policies can be applied at different levels:
- Domain level: Affects all users and computers in the domain.
- Organizational Unit (OU) level: Targets specific groups of users or computers within the domain.
- Site level: Applies to a specific physical location in a multi-site domain.
- Types of Settings:
    - User settings: Control user environments, such as desktop settings, application restrictions, and logon scripts.
    - Computer settings: Manage computer configurations, such as security settings, software installation, and network access.

Q.4) Explain briefly about Malware protection and anti-virus in server security.

Malware protection and anti-virus solutions play a crucial role in preventing malicious software from infecting and compromising your systems.

Malware Protection:

- Protects against: Viruses, worms, Trojans, ransomware, spyware, and other malicious software.
- Detection and prevention: Scans files, network traffic, and system activity for suspicious behavior and known malware signatures.
- Real-time monitoring: Provides continuous protection against constantly evolving threats.

Features:

- Automatic updates to maintain protection against new threats.
- Scheduled scans for deeper analysis.
- Sandboxing for suspicious files to isolate potential threats.
- Integration with other security tools for comprehensive defense.

Anti-virus:

- Subset of malware protection: Specifically focuses on detecting and blocking viruses.
- Lighter footprint: Often less resource-intensive compared to broader malware protection solutions.
- Effective for known viruses: Identifies and quarantines viruses based on signatures in a virus database.
- May not cover all threats: Not as effective against other types of malware like ransomware or zero-day attacks.

Q.5) What is Endpoint Security ?

Endpoint security involves safeguarding devices - endpoints - like laptops, desktops, mobile phones, and tablets from unauthorized access, malicious software, and cyberattacks.

Endpoint security protects these endpoints by implementing various measures, including:

- Anti-malware and anti-virus solutions: Detect and block viruses, worms, and other malicious software.
- Application control: Restrict the applications users can run, preventing unauthorized software installation.
- Data loss prevention (DLP): Monitor and control how data is transferred and accessed, preventing unauthorized data leaks.
- Intrusion detection and prevention systems (IDS/IPS): Identify and block suspicious network activity and attempted intrusions.
- Endpoint detection and response (EDR): Continuously monitor endpoints for suspicious activity and respond quickly to potential threats.
- Device encryption: Encrypt sensitive data stored on devices to protect it from unauthorized access in case of theft or loss.

Q.6) What is a shadow password?

A shadow password refers to a technique used to enhance the protection of user passwords on Unix-based systems. Traditionally, user passwords were stored in plain text or a simple encryption format within a file called /etc/passwd. This posed a significant security risk, as anyone with access to this file could potentially glean password information and attempt unauthorized logins.

To address this vulnerability, the concept of shadow passwords emerged. It involves separating the password data from the /etc/passwd file and storing it in a separate file called /etc/shadow. This shadow file is only accessible to the system administrator (root user) and employs several security measures to safeguard the sensitive password information:

- Password Hashing: Instead of storing passwords in plain text, the /etc/shadow file stores them as hashes. A hash is a unique mathematical string generated by applying a hashing algorithm to the original password. Even if an attacker gains access to the /etc/shadow file, they cannot directly decipher the actual password from the hash value.
- Salt Addition: To further strengthen password security, shadow passwords incorporate a technique called salting. A random string of characters, the salt, is appended to the password before it's hashed. This prevents attackers from pre-computing hashes for common passwords and using them to crack actual user passwords.
- Restricted Access: Unlike the /etc/passwd file, which is often world-readable, the /etc/shadow file has strict access controls. Only the root user possesses the necessary permissions to read or modify its contents.

Q.7) **What is OWASP? Explain the common security issues in Web applications.**

OWASP (Open Web Application Security Project) is a non-profit organization dedicated to improving the security of web applications. It's a global community of security experts, developers, and researchers who work together to identify and address web application vulnerabilities.

OWASP's Top 10 list is a widely recognized resource that highlights the most critical web application security risks.

Rank	Issue	Description
1	Broken Access Control	Failure to properly restrict access to sensitive data and functionality based on user roles
2	Cryptographic Failures	Weaknesses in encryption and hashing algorithms, allowing attackers to steal or tamper with data
3	Injection	Exploitation of untrusted data inputs to inject malicious code and control application behavior
4	Insecure Design	Flaws in the design of the application itself, even if implemented correctly
5	Security Misconfiguration	Improper configuration of security settings, leaving vulnerabilities open
6	Vulnerable and Outdated Components	Using components with known vulnerabilities that have not been patched
7	Identification and Authentication Failures	Weak or improperly implemented authentication mechanisms, allowing unauthorized access to accounts
8	Software and Data Integrity Failures	Failure to protect software and data from unauthorized modification
9	Security Logging and Monitoring Failures	Lack of proper logging and monitoring of security events, hindering incident detection and response
10	Server-Side Request Forgery (SSRF)	Tricking the application into making requests to unintended servers, potentially accessing sensitive data or internal systems

Q.8) What is XSS?

XSS, or Cross-Site Scripting, is a serious web security vulnerability that allows attackers to inject malicious scripts into a website. These scripts can then be executed by the user's browser, potentially causing a variety of harm, such as:

Stealing sensitive information: Attackers can use XSS to steal cookies, session IDs, credit card numbers, and other sensitive data from users' browsers.
Hijacking user accounts: By stealing cookies or session IDs, attackers can take over legitimate user accounts.
Defacing websites: Attackers can use XSS to inject malicious code that defaces a website, displaying unwanted content or redirecting users to malicious websites.
Spreading malware: XSS can be used to spread malware by tricking users into clicking on malicious links or downloading infected files.

Q.9) What is SQL Injection and explain its types.

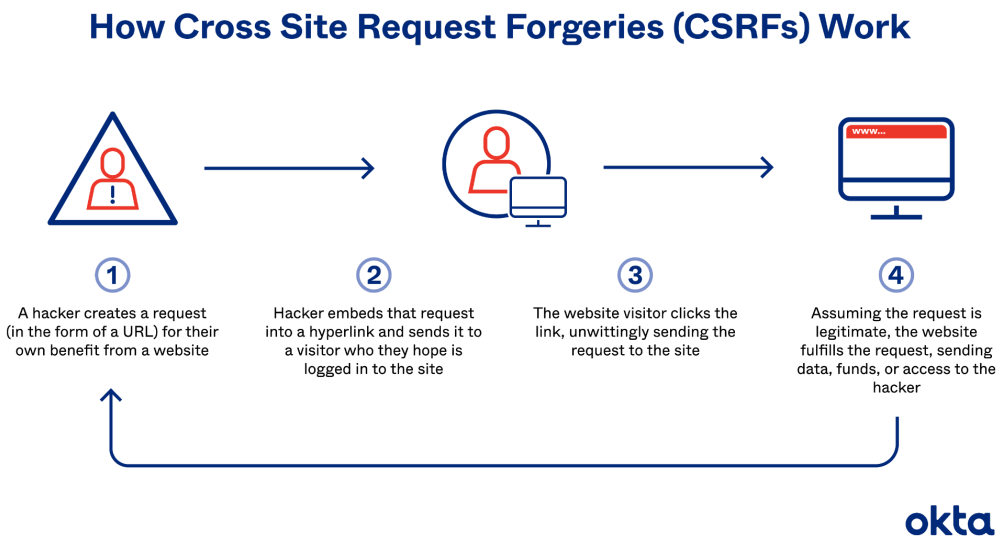
SQL Injection (SQLi) is a common web security vulnerability that allows attackers to interfere with the queries that an application makes to its database. This can have severe consequences, including:

- Unauthorized data access: Attackers can view sensitive data that they are not normally allowed to access, such as customer information, financial records, or trade secrets.
- Data modification or deletion: Attackers can modify or delete data in the database, potentially corrupting or destroying valuable information.
- Administrative access: In some cases, attackers can gain administrative access to the database, allowing them to fully control the application and its data.

Type of SQL Injection	Description	Example
In-band SQLi	The attacker can see the results of the attack directly within the application's response.	Entering `' OR 1=1 --` in a login form to bypass authentication and view all user records.
Blind SQLi	The attacker doesn't see results directly but can infer information based on the application's behavior, such as different response times or error messages.	Using time-based techniques to determine whether a specific user exists in the database.
Out-of-band SQLi	The attacker can send the results of the attack to a different system they control, such as by using DNS queries or email.	Using techniques like `UNION SELECT` to send sensitive data to an attacker-controlled server.
Error-based SQLi	The attacker leverages database error messages to gather information about the database structure or content.	Injecting invalid input to trigger error messages that reveal sensitive data.

Q.10) What is CSRF? How does CSRF work?

CSRF (Cross-Site Request Forgery), also known as a one-click attack, is a web security vulnerability that allows an attacker to trick a user's browser into making unwanted requests to a web application that the user is currently authenticated to. This can result in unauthorized actions being performed on the user's behalf, without their knowledge or consent.



Example:

Consider a banking website that allows users to transfer funds online. An attacker could craft a malicious link that, when clicked, would send a request to the bank's website to transfer funds from the victim's account to the attacker's account. If the victim is currently logged in to the bank's website, their browser would automatically include their authentication cookies in the request, making it appear as if the victim is initiating the transfer.

Q.11) What are different Password vulnerabilities?

Passwords are often the first line of defense against unauthorized access to our online accounts and systems. However, weak passwords and improper password management practices can create vulnerabilities that attackers can exploit to gain access.

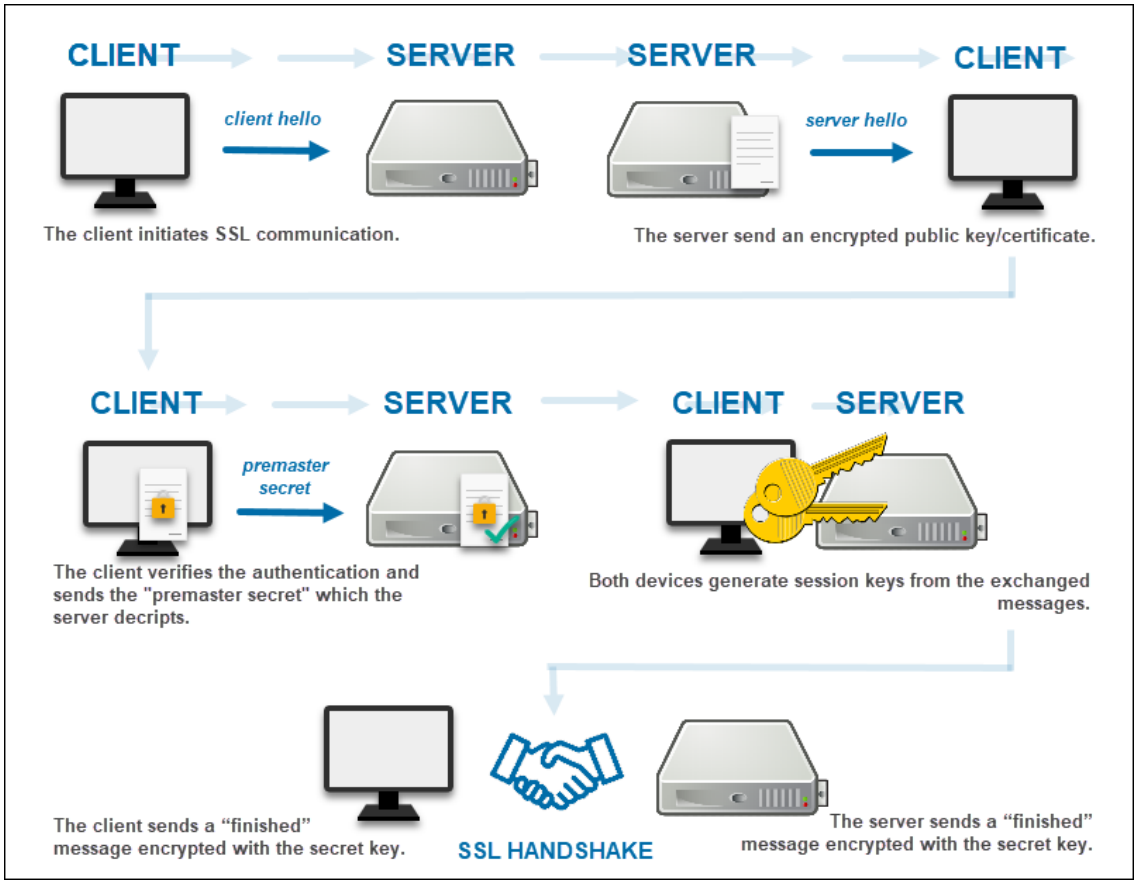
Following are different password vulnerabilities:

Password Vulnerabilities: A Tabular Breakdown

Category	Vulnerability	Description
Weak Passwords	Short passwords	Less than 12 characters, easily guessed or cracked.
	Simple passwords	Dictionary words, names, birthdays, predictable.
	Predictable passwords	Sequential numbers, keyboard patterns, easy to guess.
	Reused passwords	Same password for multiple accounts, single breach = all compromised.
Storage Vulnerabilities	Plain text storage	Passwords stored unprotected, anyone with database access can see them.
	Weak hashing algorithms	MD5 or similar, easier for attackers to crack passwords from hashes.
	Lack of salting	No random string before hashing, pre-computed hash attacks more effective.
Social Engineering & Phishing	Phishing attacks	Tricked into revealing passwords through fake emails or websites.
	Social engineering	Deceived into giving up passwords through manipulation or trickery.
Malware	Keyloggers	Record everything you type, including passwords.
	Credential stealers	Malware steals passwords from browser or applications.
Physical Access	Shoulder surfing	Someone observes you entering your password.

Q.12) What is SSL? How does it work ?

SSL (Secure Sockets Layer) is a security protocol that creates an encrypted connection between a web server and a web browser. This encrypted link ensures that all data transmitted between the server and browser remains private and secure, protecting sensitive information like credit card numbers, passwords, and personal data from eavesdropping or tampering.



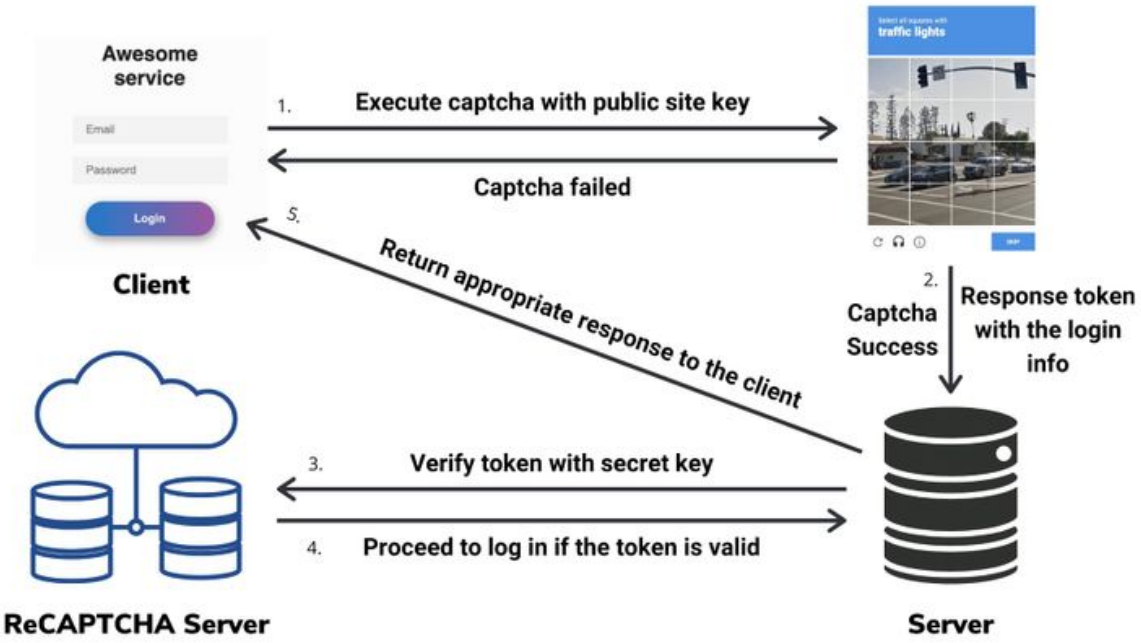
1. Client Hello: The process begins when a user attempts to connect to a secure website (HTTPS). The web browser sends a "Client Hello" message to the web server, indicating its desire to establish a secure connection.
2. Server Hello: The server responds with a "Server Hello" message, which includes its SSL certificate. This certificate contains important information, such as the server's public key and its identity (verified by a trusted third-party certificate authority).
3. Authentication and Key Exchange: The browser verifies the authenticity of the server's certificate using the trusted certificate authority. If the certificate is valid, the browser proceeds to generate a session key, a unique, temporary key used to encrypt the communication.
4. Session Key Exchange: The browser encrypts the session key using the server's public key and sends it to the server. Only the server, with its corresponding private key, can decrypt the session key.
5. Secure Symmetric Encryption: Once both the server and browser have the shared session key, they use it to encrypt and decrypt all subsequent communication. This symmetric encryption method ensures data confidentiality and integrity.

6. Secure Data Exchange: The encrypted data is transmitted back and forth between the browser and server, ensuring that any intercepted data remains unreadable to unauthorized parties.

Q.13) What is CAPTCHA? How does it work ?

CAPTCHA stands for Completely Automated Public Turing Test to Tell Computers and Humans Apart. It's a security challenge designed to distinguish between real humans and automated bots attempting to access an online service.

This is diagramatic representation of how CAPTCHA works:



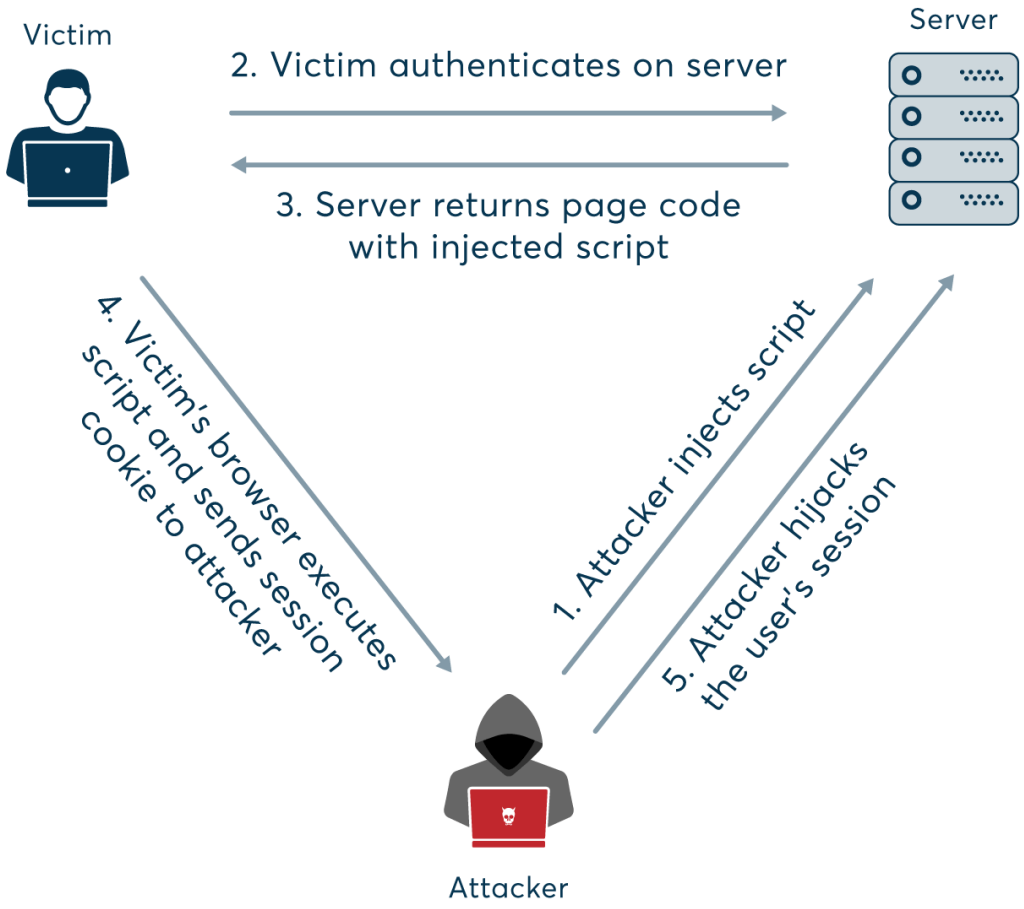
Types of CAPTCHAs:

- Text-based CAPTCHAs: These traditional options present distorted text or numbers that users need to identify and type correctly.
- Image-based CAPTCHAs: These show images and ask users to select specific ones based on criteria like objects, animals, or specific features.
- Audio-based CAPTCHAs: These play audio clips like spoken words or numbers, and users need to identify and type them correctly.
- Advanced CAPTCHAs: Newer advancements like checkbox grids, puzzles, or invisible challenges further enhance bot detection.

Q.14) What is Session Hijacking? How does it work ?

Session hijacking is an online attack where a hacker takes control of an active user session on a website or application. Once in control, the attacker can impersonate the legitimate user, perform unauthorized actions, and potentially steal sensitive information.

Following diagram represents, how session hijacking works:



Q.15) What is Local and Remote file inclusion in Web Security?

Local File Inclusion (LFI) and Remote File Inclusion (RFI) are web application vulnerabilities that allow attackers to include and execute arbitrary files on a web server.

Local and Remote File Inclusion Vulnerabilities in Web Security

Vulnerability	Description	Example
Local File Inclusion (LFI)	Attacker includes and executes malicious code or sensitive files on the server by injecting user-supplied input into a vulnerable application.	User injects `../etc/passwd` to read user credentials from a system file.
Remote File Inclusion (RFI)	Attacker includes and executes malicious code from a remote server by injecting a URL into a vulnerable application.	User injects URL to a malicious script that allows attacker control of the server.
Prevention Measures	Validate and sanitize user input, whitelist allowed file paths/URLs, disable dynamic file inclusion, update software, use WAFs.	

Q.16) What is an audit trail in Information Security ?

An audit trail acts as a meticulous record-keeper, documenting every occurrence and action within a system.

Key Elements of an Effective Audit Trail

Element	Description	Importance
Comprehensiveness	Captures all relevant events (logins, data modifications, configurations, security alerts).	Provides complete picture of system activity for investigations and monitoring.
Timestamping	Records precise date and time of each event.	Enables accurate reconstruction of event timeline and identifying sequence of actions.
User Identification	Links events to specific users whenever possible.	Facilitates accountability and tracks user behavior patterns.
Data Integrity	Ensures data within the trail is tamper-proof and secure.	Guarantees its reliability as evidence in case of legal or security incidents.
Accessibility	Allows authorized personnel to easily access and analyze the audit trail.	Enables timely response to potential security breaches and performance issues.

Implementing Audit Trails: Tools and Techniques

Tool or Technique	Description
Logging Mechanisms	Robust logging frameworks to capture system events.
Security Information and Event Management (SIEM) Systems	Aggregate and analyze data from various sources, including audit trails.
Web Application Firewalls (WAFs)	Often include logging functionalities for audit trail purposes.
Database Auditing Tools	Monitor and record database activities.