Security – types of attacks and preventive measures

1. File Inclusion Attacks: File inclusion attacks occur when an attacker exploits vulnerable file upload forms or includes files from untrusted sources, potentially allowing them to execute malicious code on the server.
   Preventive Measures:
   - Validate file extensions and content types during file uploads.
   - Implement proper access controls to restrict file inclusion to trusted directories.

2. Session Hijacking/Session Fixation: Session hijacking involves an attacker stealing or intercepting a valid session token to impersonate a legitimate user. Session fixation involves an attacker forcing a user to use a known session ID.
   Preventive Measures:
   - Use HTTPS to encrypt data in transit, especially session tokens.
   - Implement secure session management techniques, including rotating session IDs and using strong random tokens.

3. Security Misconfigurations: Security misconfigurations occur when a system or application is not properly configured, leaving it vulnerable to various attacks.
   Preventive Measures:
   - Regularly conduct security assessments, such as penetration testing and vulnerability scanning.
   - Follow best practices for server and application configuration and apply security patches promptly.

4. Clickjacking Attacks: Clickjacking involves tricking a user into clicking on something different from what the user perceives, potentially leading to unintended actions.
   Preventive Measures:
   - Implement X-Frame-Options headers to prevent web pages from being embedded into iframes without permission.
5. Man-in-the-Middle (MitM) Attacks: MitM attacks occur when an attacker intercepts and potentially alters communications between two parties without their knowledge.
   Preventive Measures:
   - Use HTTPS with strong encryption to secure communications.
   - Implement public key pinning and certificate validation.

6. XML External Entity (XXE) Attacks: XXE attacks exploit vulnerabilities in XML parsers, allowing attackers to read sensitive files, execute remote requests, and gain unauthorized access.
   Preventive Measures:
   - Disable external entity references in XML parsers.
   - Use a secure XML parser that is not vulnerable to XXE attacks.

7. Insecure Deserialization: Insecure deserialization occurs when untrusted data is deserialized, potentially leading to remote code execution or other types of attacks.
   Preventive Measures:
   - Implement proper input validation and sanitize data before deserialization.

- Use secure deserialization libraries and frameworks.

8. Data Breaches and Information Leakage: Data breaches involve unauthorized access to sensitive information, which can lead to its exposure or theft.
   Preventive Measures:
   - Encrypt sensitive data at rest and in transit.
   - Implement access controls and data minimization practices.

9. Zero-Day Vulnerabilities: Zero-day vulnerabilities are previously unknown security flaws that are exploited before a fix or patch is available.
   Preventive Measures:
   - Stay updated on security advisories and patches.
   - Implement intrusion detection systems and behavior-based anomaly detection.

10. Shell Script Upload: Shell script upload attacks occur when an attacker uploads malicious shell scripts to a web server, enabling them to execute arbitrary commands on the server.
    Preventive Measures:
    - File Type Validation: Ensure that file uploads are restricted to specific file types (e.g., images, documents) and do not allow executable files like shell scripts.
    - File Size Limitations: Implement size limitations for uploaded files to prevent the upload of excessively large or potentially malicious files.
    - Use Secure File Permissions: Configure file and directory permissions appropriately to restrict execution of uploaded files.
    - Regular Scans and Monitoring: Conduct periodic security scans to detect and remove any malicious files that may have been uploaded.
    - Input Sanitization: Validate and sanitize user inputs, particularly during file uploads, to prevent the execution of malicious scripts.