Understanding 200 Advanced Cybersecurity Threats and How to Defend Against

1. Phishing Attacks

- **How it Works**: Phishing involves sending deceptive emails or messages to trick users into revealing personal information like passwords or credit card numbers.
- **Prevention**: Use email filters, multi-factor authentication (MFA), educate users on identifying phishing emails, and implement domain-based message authentication.

2. Spear Phishing

- **How it Works**: A targeted phishing attack where the attacker customizes the message to a specific individual or organization, often by gathering information beforehand.
- **Prevention**: Employee training, email verification, MFA, and advanced email security software.

3. Man-in-the-Middle (MitM) Attack

- **How it Works**: The attacker intercepts communication between two parties, either to eavesdrop or alter the data being transmitted.
- **Prevention**: Use encryption protocols like SSL/TLS, VPNs, and always verify the authenticity of websites and connections.

4. Distributed Denial of Service (DDoS) Attack

- **How it Works**: A DDoS attack floods a target system with excessive traffic, causing it to become overwhelmed and unavailable to legitimate users.
- **Prevention**: Implement rate limiting, firewalls, anti-DDoS systems, and content delivery networks (CDNs).

5. SQL Injection

- **How it Works**: An attacker inserts malicious SQL code into input fields to manipulate the database and gain unauthorized access to data.
- **Prevention**: Use parameterized queries, validate input, and ensure proper database access controls.

6. Cross-Site Scripting (XSS)

- **How it Works**: XSS attacks inject malicious scripts into web pages viewed by other users. These scripts can steal session cookies or perform actions on behalf of the user.
- **Prevention**: Sanitize and validate user inputs, use content security policies (CSP), and encode outputs.

7. Malware (Malicious Software)

- **How it Works**: Malware is any software specifically designed to disrupt, damage, or gain unauthorized access to systems. This includes viruses, worms, ransomware, etc.
- **Prevention**: Install antivirus software, keep systems updated, and avoid downloading files from untrusted sources.

8. Ransomware

- **How it Works**: Ransomware encrypts a victim's data and demands a ransom to restore access.
- **Prevention**: Regular backups, strong access controls, and endpoint protection software. Educate users on avoiding malicious links and attachments.

9. Advanced Persistent Threat (APT)

- **How it Works**: APTs involve long-term, targeted attacks, often state-sponsored, where attackers infiltrate networks and remain undetected for extended periods.
- **Prevention**: Use advanced threat detection tools, continuous network monitoring, patch management, and least privilege access controls.

10. Zero-Day Exploits

- **How it Works**: A zero-day exploit targets vulnerabilities in software that the vendor has not yet discovered or patched.
- **Prevention**: Regularly update systems and software, use security patches, and deploy threat intelligence services.

11. Credential Stuffing

- **How it Works**: Attackers use stolen username and password combinations from previous breaches to gain unauthorized access to other accounts.
- **Prevention**: Implement MFA, use password managers, and enforce strong password policies.

12. Brute Force Attack

- **How it Works**: A brute force attack involves trying many passwords or encryption keys until the correct one is found.
- **Prevention**: Use account lockout mechanisms, CAPTCHAs, and complex passwords with special characters.

13. Eavesdropping (Sniffing)

- **How it Works**: Attackers intercept communication between two parties, often using packet-sniffing tools, to capture sensitive information like login credentials.
- **Prevention**: Use encryption protocols (SSL/TLS, VPNs), avoid public Wi-Fi networks for sensitive communications.

14. Session Hijacking

- **How it Works**: In this attack, the attacker steals a user's session cookie or token to impersonate the user.
- **Prevention**: Use secure HTTP headers, implement session expiration, and secure cookies using HTTPOnly and Secure flags.

15. Privilege Escalation

- **How it Works**: Attackers exploit weaknesses in a system to gain elevated access rights, allowing them to perform unauthorized actions.

- **Prevention**: Regularly audit permissions, apply the principle of least privilege, and patch vulnerabilities.

16. Social Engineering

- **How it Works**: Attackers manipulate individuals into revealing confidential information by exploiting human psychology rather than technical vulnerabilities.
- **Prevention**: User training, verifying identities, and cautious handling of sensitive information.

17. Drive-By Download

- **How it Works**: A drive-by download happens when a user visits a malicious website and unintentionally downloads malware onto their device.
- **Prevention**: Use updated browsers, block pop-ups, and deploy endpoint protection software.

18. DNS Spoofing

- **How it Works**: The attacker corrupts the DNS cache to redirect users to malicious websites without their knowledge.
- **Prevention**: Implement DNSSEC (Domain Name System Security Extensions) and monitor DNS traffic for anomalies.

19. Cross-Site Request Forgery (CSRF)

- **How it Works**: CSRF tricks the victim's browser into performing unwanted actions on a web application where the user is authenticated.
- **Prevention**: Use anti-CSRF tokens, validate user input, and implement proper session management.

20. Keylogging

- **How it Works**: Keyloggers are malicious programs or hardware devices that record keystrokes on a computer to steal sensitive information such as passwords.
- **Prevention**: Use endpoint protection software, avoid untrusted devices, and implement secure input methods like on-screen keyboards.

21. Watering Hole Attack

- **How it Works**: In a watering hole attack, attackers compromise a website that is frequently visited by the target organization. Once users visit the infected site, malware is delivered to their systems.
- **Prevention**: Regular patching, web security monitoring, use of website reputation tools, and endpoint protection.

22. Malicious Insider Threats

- **How it Works**: Malicious insiders exploit their access to an organization's systems, networks, or data for personal gain or to damage the organization.
- **Prevention**: Implement strict access controls, monitor user activity, conduct regular audits, and provide employee security training.

23. Supply Chain Attack

- **How it Works**: Attackers infiltrate a third-party vendor or software provider in order to compromise their systems, which can then be used to attack the organization using that vendor's services or software.
- **Prevention**: Thorough vetting of third-party vendors, implement multi-layered security protocols, and use network segmentation.

24. Fake Wireless Networks (Evil Twin Attack)

- **How it Works**: In an evil twin attack, attackers create a fake Wi-Fi network that appears to be a legitimate one. Users unknowingly connect to this network, giving attackers access to sensitive data.
- **Prevention**: Educate users not to connect to unverified networks, use VPNs, and ensure proper Wi-Fi encryption (WPA3).

25. Clickjacking

- **How it Works**: Clickjacking tricks users into clicking on something different from what they perceive, potentially revealing confidential information or triggering unwanted actions.
- **Prevention**: Use frame-busting code to prevent content from being embedded in an iframe, and implement content security policies (CSP).

26. Fake Antivirus Software

- **How it Works**: Attackers create fake antivirus programs that appear to scan and protect a system, but in reality, they install malware or steal sensitive data.
- **Prevention**: Avoid downloading software from untrusted sources, use reputable antivirus programs, and perform regular system scans.

27. IoT (Internet of Things) Attacks

- **How it Works**: Attackers target vulnerabilities in connected devices (like smart thermostats, security cameras, etc.) to gain unauthorized access to the network or device.
- **Prevention**: Ensure IoT devices are updated, segment IoT devices from critical networks, and use strong authentication methods.

28. Domain Kiting

- **How it Works**: Domain kiting refers to the process of registering a domain, using it briefly, and then deleting it within the allowed five-day window to avoid paying for it, often for malicious purposes like spamming.
- **Prevention**: Implement domain monitoring and use of proper DNS configurations to block domain hijacking.

29. Zero-Day Malware

- **How it Works**: Zero-day malware exploits unknown vulnerabilities in software or hardware. Because the vulnerabilities are not yet discovered or patched, the malware remains undetected.
- **Prevention**: Keep systems and software up to date, deploy intrusion detection systems (IDS), and regularly review vulnerability reports.

30. Logic Bomb

- **How it Works**: A logic bomb is a piece of malicious code hidden within a program that activates when specific conditions are met (such as a certain date or event).
- **Prevention**: Code review processes, continuous monitoring, and use of behavior-based detection tools.

31. Fake Social Media Accounts

- **How it Works**: Attackers create fake social media profiles to impersonate a person or organization to spread misinformation, conduct scams, or gather sensitive information.
- **Prevention**: Implement social media monitoring tools, educate employees about social media security, and verify identities.

32. Cryptojacking

- **How it Works**: Cryptojacking occurs when attackers hijack a user's computer to mine cryptocurrency without their knowledge, slowing down systems and increasing electricity usage.
- **Prevention**: Use antivirus and anti-malware software, block access to mining sites, and monitor system performance.

33. Rogue Security Software

- **How it Works**: This attack involves fake security software that falsely claims to detect issues on a device, prompting the user to pay for unnecessary services or software.
- **Prevention**: Avoid downloading unknown software, use trusted security software, and educate users about the risks of rogue security tools.

34. Side-Channel Attacks

- **How it Works**: Side-channel attacks exploit physical or environmental factors like power consumption, electromagnetic leaks, or timing information to gain access to encrypted data or secrets.
- **Prevention**: Use hardware security features, reduce signal emissions, and employ strong encryption algorithms.

35. Click Fraud

- **How it Works**: In click fraud, attackers click on online ads with the intent of generating false impressions or clicks, costing advertisers money.
- **Prevention**: Use anti-fraud software, monitor traffic for suspicious activity, and implement CAPTCHA mechanisms to verify clicks.

36. Bait and Switch

- **How it Works**: Attackers lure users to click on an attractive offer or download a file, but once accessed, it contains malicious software or redirects users to unwanted websites.
- **Prevention**: Avoid clicking on unknown links, use email filters, and deploy ad-blocking software.

37. Data Exfiltration

- **How it Works**: Data exfiltration is the unauthorized transfer of sensitive data from an organization’s network to an external location, often for malicious purposes.
- **Prevention**: Encrypt sensitive data, monitor outbound network traffic, and enforce data access policies.

38. DNS Amplification Attack

- **How it Works**: In this DDoS attack, attackers exploit DNS servers to flood a target with traffic by using DNS queries to amplify the volume of data.
- **Prevention**: Disable open DNS resolvers, deploy rate-limiting, and configure firewalls to block large DNS responses.

39. Session Fixation Attack

- **How it Works**: Attackers set a session ID for a user before they log in, allowing them to hijack the session after login and impersonate the user.
- **Prevention**: Regenerate session IDs after login, implement secure cookie attributes, and use HTTPS.

40. Pharming

- **How it Works**: Pharming attacks redirect users from legitimate websites to malicious ones by compromising the DNS resolution process or altering host files.
- **Prevention**: Use DNSSEC, ensure software and OS are up to date, and educate users on website legitimacy.

41. DNS Tunneling

- **How it Works**: DNS tunneling exploits the DNS protocol to transmit data over port 53, which is typically open on most firewalls. Attackers encode data within DNS queries to bypass security measures.
- **Prevention**: Monitor DNS traffic for unusual patterns, use DNS filtering, and implement firewalls that block unauthorized outbound DNS traffic.

42. Click Fraud

- **How it Works**: Click fraud occurs when an attacker intentionally clicks on digital ads to generate revenue for themselves or to deplete an advertiser's budget. This is done through bots or fake accounts.
- **Prevention**: Use click fraud detection software, filter suspicious traffic patterns, and monitor ad clicks closely.

43. Privilege Abuse

- **How it Works**: Privilege abuse occurs when a user with elevated permissions takes advantage of their access rights to perform unauthorized actions or gain financial benefits.
- **Prevention**: Implement least privilege principles, conduct regular audits of access rights, and monitor user activity to detect unusual behavior.

44. Email Spoofing

- **How it Works**: In email spoofing, an attacker sends an email that appears to come from a trusted source but is actually from an attacker. The goal is often to trick recipients into revealing sensitive information or downloading malware.
- **Prevention**: Implement email authentication protocols like SPF, DKIM, and DMARC, and use advanced email filtering systems to detect suspicious emails.

45. Session Hijacking

- **How it Works**: Session hijacking involves an attacker stealing a session token or cookie from a user to impersonate them and gain unauthorized access to their account.
- **Prevention**: Use secure cookies (HTTPOnly and Secure flags), regenerate session IDs after login, and implement multi-factor authentication (MFA).

46. Fake App Attacks

- **How it Works**: Attackers distribute malicious mobile or desktop applications that look legitimate but contain malware designed to steal data or perform malicious actions.
- **Prevention**: Only download apps from trusted sources, use application whitelisting, and use endpoint protection software.

47. Rogue Access Points

- **How it Works**: In this attack, an attacker sets up a rogue Wi-Fi access point with a similar name to a legitimate network. Unsuspecting users connect to it, allowing attackers to intercept sensitive data.
- **Prevention**: Use strong encryption (WPA3), require users to verify Wi-Fi networks, and deploy wireless intrusion detection systems.

48. Password Cracking

- **How it Works**: Attackers attempt to gain access to accounts by trying numerous password combinations until the correct one is found. This can be done using brute force or dictionary attacks.
- **Prevention**: Use strong, complex passwords, enforce password expiration policies, and implement account lockout after a certain number of failed attempts.

49. Cache Poisoning

- **How it Works**: Cache poisoning is an attack where the attacker introduces malicious data into a cache, which is then served to users who rely on the cached content, leading to malware or other vulnerabilities.
- **Prevention**: Validate and sanitize cached data, use secure cache control mechanisms, and monitor cache entries for anomalies.

50. Typosquatting

- **How it Works**: Typosquatting involves registering domains with misspellings of popular websites (such as "faceboook.com") to trick users into visiting the malicious site, where they can be exposed to phishing or malware.

- **Prevention**: Register common misspellings of your domain name, and use domain monitoring services to detect typosquatting attacks.

51. Cross-Site Request Forgery (CSRF)

- **How it Works**: CSRF tricks a user into performing actions on a website without their consent. For example, a user logged into their bank account may unknowingly authorize a transaction.
- **Prevention**: Use anti-CSRF tokens, check the referer header, and ensure that all state-changing requests require authentication.

52. Drive-By Downloads

- **How it Works**: In a drive-by download attack, malicious software is automatically downloaded onto a victim's device simply by visiting a compromised website.
- **Prevention**: Use up-to-date antivirus and anti-malware software, ensure browsers are patched, and avoid visiting suspicious websites.

53. DNS Spoofing

- **How it Works**: DNS spoofing, also known as DNS cache poisoning, involves altering the DNS records of a website, redirecting users to malicious sites without their knowledge.
- **Prevention**: Implement DNSSEC, configure DNS servers securely, and monitor DNS traffic for unusual patterns.

54. Keylogging

- **How it Works**: Keylogging involves the installation of software or hardware to monitor and record keystrokes on a victim's device. The attacker can then capture sensitive data like login credentials or credit card numbers.
- **Prevention**: Use antivirus software, employ virtual keyboards for sensitive inputs, and ensure devices are physically secure.

55. Packet Sniffing

- **How it Works**: Packet sniffing involves intercepting data packets on a network to extract sensitive information, such as login credentials or personal data.
- **Prevention**: Use encryption (SSL/TLS), secure your network with VPNs, and use intrusion detection systems (IDS) to monitor for suspicious activity.

56. Botnets

- **How it Works**: Botnets are networks of infected devices that can be controlled remotely by attackers to perform malicious actions such as launching DDoS attacks or sending spam emails.
- **Prevention**: Deploy firewalls, regularly update systems, and use anti-malware software to detect and prevent botnet infections.

57. Man-in-the-Browser Attack

- **How it Works**: In a man-in-the-browser attack, malware infects a user's browser to intercept and alter data in real-time without the user's knowledge, often for financial fraud.

- **Prevention**: Use advanced endpoint protection, monitor browser activity, and implement multi-factor authentication.

58. Replay Attacks

- **How it Works**: A replay attack occurs when an attacker intercepts and retransmits valid data (like login credentials or payment details) to gain unauthorized access.
- **Prevention**: Use timestamping to detect and prevent the retransmission of old messages, and employ encryption and authentication techniques like digital signatures.

59. Social Engineering

- **How it Works**: Social engineering exploits human psychology to manipulate individuals into divulging confidential information. This can include phishing, baiting, or impersonation.
- **Prevention**: Regular employee training, implementing strict verification protocols, and using multi-factor authentication.

60. Eavesdropping

- **How it Works**: Eavesdropping occurs when attackers intercept communications, such as phone calls, emails, or network traffic, to gather sensitive information.
- **Prevention**: Use encryption (SSL/TLS for emails, VPNs for networks), and ensure physical security for communications devices.

61. Credential Stuffing

- **How it Works**: Credential stuffing involves using automated tools to try large sets of username and password combinations (usually from previous data breaches) in order to gain unauthorized access to multiple accounts.
- **Prevention**: Implement multi-factor authentication (MFA), use rate limiting on login attempts, and encourage users to use unique passwords for each account.

62. Vishing (Voice Phishing)

- **How it Works**: Vishing involves attackers impersonating legitimate authorities or organizations over the phone to steal sensitive information, like bank account details or personal identification numbers (PINs).
- **Prevention**: Verify caller identities, be cautious when sharing personal information over the phone, and implement call-blocking solutions.

63. SMS Phishing (Smishing)

- **How it Works**: Smishing involves sending fraudulent SMS messages that trick the user into revealing sensitive information, such as login credentials or banking details, by clicking on malicious links.
- **Prevention**: Educate users not to click on links in unsolicited messages, and implement URL filtering solutions that detect malicious links.

64. Click Fraud

- **How it Works**: Click fraud happens when an attacker simulates clicks on ads to generate revenue for themselves or deplete an advertiser's budget. It can be performed through bots or fake accounts.
- **Prevention**: Monitor click patterns for unusual activities, use click fraud detection software, and implement CAPTCHA tests for interactions with ads.

65. Formjacking

- **How it Works**: Formjacking involves injecting malicious code into online forms (e.g., payment forms) to steal sensitive information such as credit card numbers during online transactions.
- **Prevention**: Use secure HTTPS connections for all payment forms, implement web application firewalls (WAF), and ensure regular updates and patching.

66. Fuzzing

- **How it Works**: Fuzzing is a testing technique where random or malformed data is sent to a system, application, or network in order to identify vulnerabilities that can be exploited.
- **Prevention**: Use robust input validation, implement regular patching cycles, and adopt security coding practices to mitigate vulnerabilities.

67. Rogue Software

- **How it Works**: Rogue software refers to malicious programs that mimic legitimate software, tricking users into installing them by claiming to fix or optimize their systems. Once installed, it can perform malicious actions like data theft or spreading malware.
- **Prevention**: Use trusted antivirus software, avoid downloading programs from unverified sources, and educate users on identifying rogue software.

68. Buffer Overflow Attacks

- **How it Works**: Buffer overflow attacks occur when an attacker sends more data to a buffer than it can handle, causing data to overwrite adjacent memory locations, potentially leading to code execution and system compromise.
- **Prevention**: Use safe programming techniques (e.g., bounds checking), enable data execution prevention (DEP), and ensure systems are regularly updated with security patches.

69. Evil Maid Attack

- **How it Works**: An evil maid attack involves an attacker gaining physical access to a device (typically a laptop) and tampering with its firmware or software to steal data or install malware.
- **Prevention**: Use full-disk encryption, disable booting from external devices, and implement strong physical security controls to prevent unauthorized access.

70. Privilege Escalation

- **How it Works**: Privilege escalation involves exploiting vulnerabilities or misconfigurations to gain higher levels of access than initially granted, often allowing attackers to control systems or exfiltrate sensitive data.

- **Prevention**: Apply the principle of least privilege, use strong access controls, and regularly review and audit user permissions.

71. Backdoor Attack

- **How it Works**: A backdoor is a secret method of bypassing normal authentication or encryption in a system. Attackers may install backdoors to gain continuous access to a compromised system.
- **Prevention**: Regularly scan systems for unauthorized access points, use endpoint protection software, and employ network monitoring to detect unusual behavior.

72. Doxxing

- **How it Works**: Doxxing involves collecting and publicly disclosing private or personal information about an individual, such as their home address or phone number, often to cause harm or harassment.
- **Prevention**: Be cautious about sharing personal information online, monitor your digital footprint, and use privacy settings on social media platforms to restrict access to sensitive information.

73. Zero-Day Exploit

- **How it Works**: A zero-day exploit takes advantage of an unpatched vulnerability in software or hardware that the vendor or user is unaware of, allowing attackers to execute malicious actions before the flaw is fixed.
- **Prevention**: Keep all systems and software up to date, use intrusion detection systems (IDS) to identify abnormal behavior, and subscribe to security bulletins for vulnerabilities.

74. Drive-By Mining

- **How it Works**: Drive-by mining occurs when an attacker uses a user's device to mine cryptocurrency without their consent, often through malicious websites or ads running in the background.
- **Prevention**: Use ad blockers, avoid visiting untrusted websites, and employ endpoint protection tools to detect and prevent drive-by mining.

75. Fileless Malware

- **How it Works**: Fileless malware operates without using files stored on disk, making it harder to detect. It resides in system memory and exploits legitimate system tools to carry out malicious activities.
- **Prevention**: Use behavior-based detection systems, ensure that endpoint security software scans memory, and restrict the use of PowerShell or scripting tools in unauthorized contexts.

76. Key Reinstallation Attacks (KRACK)

- **How it Works**: KRACK exploits vulnerabilities in the WPA2 protocol, which is used to secure Wi-Fi networks. Attackers can intercept and manipulate data exchanged between a device and the router.
- **Prevention**: Ensure that devices and routers are updated with patches for WPA2 vulnerabilities, use WPA3 if possible, and avoid using public or unsecured Wi-Fi networks.

77. Blended Attacks

- **How it Works**: Blended attacks combine different attack techniques (such as malware, phishing, and social engineering) to increase the likelihood of successful compromise and evade detection.
- **Prevention**: Employ multi-layered security defenses, integrate endpoint detection and response (EDR), and conduct regular employee training on security awareness.

78. Cryptanalysis

- **How it Works**: Cryptanalysis involves attempting to break or decrypt encrypted data without knowledge of the encryption key. It may be used to exploit weaknesses in cryptographic algorithms.
- **Prevention**: Use strong, modern encryption standards (e.g., AES-256), and regularly review and update cryptographic protocols to stay ahead of cryptanalysis techniques.

79. Social Engineering Attack

- **How it Works**: Social engineering attacks manipulate individuals into revealing confidential information or performing actions that compromise security, such as divulging passwords or clicking on malicious links.
- **Prevention**: Conduct regular employee training on recognizing social engineering tactics, implement access controls, and verify the identities of individuals before disclosing sensitive information.

80. WIFI Eavesdropping

- **How it Works**: Attackers intercept and monitor wireless network traffic to gather sensitive information such as passwords, emails, or other personal data transmitted over unsecured Wi-Fi.
- **Prevention**: Use strong encryption (WPA2 or WPA3), implement VPNs for secure communications, and avoid using public or unencrypted Wi-Fi for sensitive activities.

81. Sudo Caching Attack

- **How it Works**: Sudo caching attacks involve exploiting a flaw in the sudo command used in Unix/Linux systems. Attackers can leverage sudo caching to execute commands as the root user by bypassing authentication checks.
- **Prevention**: Regularly patch sudo versions, disable caching of sudo credentials, and monitor sudo usage with auditing tools to detect suspicious activity.

82. Malicious USB Devices

- **How it Works**: Malicious USB devices, also known as "USB killers," can deliver malware when plugged into a computer. These devices can exploit vulnerabilities in USB ports to compromise the host system.
- **Prevention**: Disable unused USB ports, implement device control policies, use endpoint protection software, and restrict the use of USB drives to authorized devices.

83. Cross-Site Scripting (XSS) via WebSockets

- **How it Works**: Similar to traditional XSS attacks, this form of XSS exploits vulnerabilities in WebSocket connections used in web applications. It allows attackers to inject malicious scripts into WebSocket messages that can execute in the victim's browser.
- **Prevention**: Sanitize WebSocket messages, implement input validation, and use secure WebSocket connections (WSS) to prevent unauthorized access.

84. SQL Injection (Blind)

- **How it Works**: Blind SQL injection occurs when attackers inject SQL queries into an application's input fields, but the server does not directly reveal the results of the query. The attacker must infer the outcome based on behavior, such as response time or server errors.
- **Prevention**: Use parameterized queries, ensure proper validation and sanitization of user input, and adopt web application firewalls (WAF).

85. Password Spraying

- **How it Works**: Password spraying is a form of brute force attack where attackers try a few common passwords across many accounts, hoping to avoid triggering account lockouts by not targeting a single account too many times.
- **Prevention**: Use multi-factor authentication (MFA), set strong password policies, and implement account lockout mechanisms after a limited number of failed login attempts.

86. DNS Amplification DDoS

- **How it Works**: DNS amplification DDoS attacks exploit misconfigured DNS servers to amplify the volume of traffic directed at a target. Attackers send a small query to the DNS server that results in a large response, which is then directed at the target.
- **Prevention**: Configure DNS servers to block recursive queries from unauthorized IP addresses, implement rate limiting, and deploy anti-DDoS services.

87. Exploitation of Cloud Misconfigurations

- **How it Works**: Attackers exploit misconfigurations in cloud services (such as AWS, Azure, or Google Cloud) to gain unauthorized access to sensitive data or services. Common misconfigurations include overly permissive access controls and exposed storage buckets.
- **Prevention**: Implement strict access control policies, regularly audit cloud configurations, and use tools like AWS Config and Azure Security Center to detect misconfigurations.

88. Privilege Escalation via Insecure Libraries

- **How it Works**: Insecure libraries, particularly in open-source software, may contain vulnerabilities that attackers can exploit to escalate privileges on the system. These libraries may have outdated versions or weak security controls.
- **Prevention**: Regularly update and patch libraries, conduct code reviews to identify potential vulnerabilities, and use tools like Snyk or OWASP Dependency-Check to monitor for insecure libraries.

89. ATM Skimming

- **How it Works**: ATM skimming involves installing a small device on an ATM machine that captures the card's data and PIN when a user inserts their card. The attacker can then clone the card and perform fraudulent transactions.
- **Prevention**: Use EMV (chip) cards instead of magnetic stripe cards, install security cameras around ATMs, and conduct regular inspections of ATM machines for skimming devices.

90. Web Shell Attacks

- **How it Works**: A web shell is a script that allows an attacker to remotely control a compromised web server. Once uploaded, the attacker can execute commands, upload malicious files, or escalate privileges.
- **Prevention**: Implement secure file upload mechanisms, regularly scan web servers for unauthorized files, and restrict permissions on web directories.

91. Cryptomining Malware

- **How it Works**: Cryptomining malware installs cryptocurrency mining software on victim machines, using their resources to mine cryptocurrency without the user's consent. This malware may be spread through malicious ads, infected websites, or email attachments.
- **Prevention**: Use endpoint protection software, monitor network activity for signs of cryptomining, and block malicious websites and ads.

92. Firmware Rootkits

- **How it Works**: Firmware rootkits are malicious software programs embedded in the firmware of hardware devices (e.g., routers, hard drives, or motherboards). These rootkits are difficult to detect and allow attackers to persist on a compromised system.
- **Prevention**: Ensure firmware is up to date, use hardware with secure boot mechanisms, and perform regular integrity checks on hardware components.

93. Email Spoofing

- **How it Works**: In email spoofing, attackers manipulate email headers to make it appear as though the email is coming from a trusted sender. This is often used in phishing attacks or to deliver malicious attachments.
- **Prevention**: Implement SPF, DKIM, and DMARC protocols for email authentication, and use email filtering solutions to detect suspicious emails.

94. HTTP Response Splitting

- **How it Works**: HTTP response splitting occurs when an attacker sends a crafted HTTP request to a web server, which results in two responses being sent to the victim’s browser. This attack can lead to session fixation, cache poisoning, or redirecting users to malicious websites.
- **Prevention**: Properly sanitize user inputs and HTTP headers, and use web application firewalls (WAF) to block suspicious activity.

95. Hardware Keyloggers

- **How it Works**: Hardware keyloggers are small physical devices that are inserted between a keyboard and a computer to record keystrokes. These devices are typically undetectable by software.
- **Prevention**: Ensure physical security of devices, use on-screen keyboards for sensitive inputs, and inspect devices for unauthorized attachments.

96. Zombie Networks (Botnets)

- **How it Works**: A botnet consists of a network of compromised devices that are controlled remotely by an attacker. The botnet can be used to conduct DDoS attacks, spread malware, or harvest data.
- **Prevention**: Use firewall and intrusion detection systems to monitor for unusual traffic patterns, regularly update software and hardware, and implement endpoint protection to detect and block botnet activity.

97. Spoofing Attacks

- **How it Works**: Spoofing attacks occur when an attacker impersonates a legitimate entity to deceive a victim into providing sensitive information or performing an action. This can include IP spoofing, email spoofing, or ARP spoofing.
- **Prevention**: Implement strong authentication mechanisms, monitor network traffic for spoofed packets, and use encryption protocols such as TLS and VPNs to secure communications.

98. Evil Twin Wi-Fi Attacks

- **How it Works**: In an evil twin attack, an attacker sets up a rogue Wi-Fi network that mimics a legitimate one. When users connect, the attacker can intercept data, perform man-in-the-middle attacks, or inject malicious content.
- **Prevention**: Educate users on how to verify Wi-Fi network names, use encrypted connections (e.g., HTTPS), and implement network monitoring tools to detect rogue access points.

99. Form Spamming

- **How it Works**: Attackers submit fake or malicious data through online forms to overload the system, steal data, or submit spam content. This can also be used to bypass CAPTCHA systems.
- **Prevention**: Use CAPTCHA and reCAPTCHA mechanisms, implement form validation, and limit form submissions per IP address.

100. Application Layer DDoS

- **How it Works**: Unlike traditional DDoS attacks that flood the network, application layer DDoS attacks target specific web applications or services, consuming resources such as CPU and memory by sending complex requests.
- **Prevention**: Use application-layer firewalls, implement rate limiting, and deploy content delivery networks (CDNs) to mitigate the impact of such attacks.

101. Time-of-Check to Time-of-Use (TOCTOU) Attack

- **How it Works**: TOCTOU attacks involve exploiting the time difference between when a resource is checked and when it is actually used. The attacker modifies the resource in between these two operations to gain unauthorized access or cause harm.
- **Prevention**: Ensure proper synchronization of operations and use atomic operations that execute in one step to eliminate the possibility of time-based inconsistencies.

102. Race Condition

- **How it Works**: A race condition occurs when two processes or threads attempt to access shared resources simultaneously, which can lead to unpredictable outcomes. Attackers exploit this vulnerability to cause unintended behavior, such as privilege escalation or data corruption.
- **Prevention**: Implement proper synchronization mechanisms (e.g., locks, semaphores) and validate input thoroughly to avoid simultaneous access to critical resources.

103. Click Fraud via Ad Networks

- **How it Works**: Click fraud via ad networks involves attackers manipulating pay-per-click (PPC) advertising systems by generating fake clicks to deplete an advertiser’s budget or to create fake traffic.
- **Prevention**: Use click fraud detection algorithms, set limits on click-through rates, and monitor traffic behavior using web analytics tools to identify abnormal click patterns.

104. Malware-as-a-Service (MaaS)

- **How it Works**: Malware-as-a-Service allows attackers to buy malware tools or services, such as DDoS bots or ransomware, from underground forums or marketplaces. This democratizes cybercrime by making advanced tools accessible to less skilled attackers.
- **Prevention**: Monitor and block malicious traffic patterns, educate employees on identifying malicious attachments, and use threat intelligence to identify new malware variants.

105. DNS Poisoning

- **How it Works**: DNS poisoning, or cache poisoning, involves an attacker injecting corrupt DNS data into a DNS resolver’s cache, redirecting users to malicious sites even when they enter legitimate URLs.
- **Prevention**: Implement DNSSEC (Domain Name System Security Extensions) to authenticate DNS data and prevent unauthorized tampering.

106. Data Breaches via Third-Party Services

- **How it Works**: Attackers exploit vulnerabilities in third-party services that an organization relies on, such as cloud storage or service providers, to gain unauthorized access to sensitive data.
- **Prevention**: Conduct thorough risk assessments of third-party vendors, ensure that security measures like encryption and multi-factor authentication (MFA) are implemented, and regularly audit third-party access to systems.

107. Memory Corruption Vulnerabilities

- **How it Works**: Memory corruption vulnerabilities occur when software improperly accesses or manipulates memory, leading to unexpected behavior, such as arbitrary code execution or privilege escalation. Attackers exploit these bugs to execute malicious code in memory.
- **Prevention**: Use buffer overflow protections, implement stack canaries, and conduct extensive testing using fuzzing techniques to identify memory corruption issues.

108. XML External Entity (XXE) Attack

- **How it Works**: In an XXE attack, an attacker exploits a vulnerability in the processing of XML input by including external entities in the XML document, which can lead to unauthorized access to files, internal network services, or denial of service.
- **Prevention**: Disable external entity processing in XML parsers, implement input validation, and use secure coding practices to handle XML safely.

109. Trojan Horse

- **How it Works**: A Trojan horse is a type of malware that disguises itself as legitimate software to trick users into installing it. Once installed, it can steal information, corrupt data, or provide unauthorized access to attackers.
- **Prevention**: Use endpoint protection software, ensure software is downloaded from trusted sources, and educate users on identifying suspicious files and applications.

110. Fileless Ransomware

- **How it Works**: Fileless ransomware infects a system by exploiting existing software, such as web browsers or scripting languages, rather than writing files to the disk. This makes it more difficult to detect using traditional file-based detection methods.
- **Prevention**: Use behavior-based detection systems, apply principle of least privilege, and ensure that macro scripts and PowerShell are tightly controlled.

111. ARP Spoofing (Address Resolution Protocol Spoofing)

- **How it Works**: ARP spoofing involves sending falsified ARP messages to a local network, causing the target’s device to associate the attacker's MAC address with the IP address of a legitimate network resource (such as a router), allowing the attacker to intercept network traffic.
- **Prevention**: Use static ARP entries, implement port security, and deploy intrusion detection systems (IDS) to detect unusual ARP traffic.

112. IP Spoofing

- **How it Works**: IP spoofing occurs when an attacker sends packets from a forged IP address, making it appear as though they are coming from a trusted source. This can be used to bypass security filters or launch DDoS attacks.
- **Prevention**: Implement ingress and egress filtering on routers to ensure that packets with spoofed IP addresses are blocked.

113. Wireless Network Jamming

- **How it Works**: Wireless network jamming involves an attacker sending interference signals on the same frequency as a legitimate Wi-Fi network to disrupt communication, rendering the network inoperable.
- **Prevention**: Use secure and less congested wireless channels, implement 802.11ac (which offers better resistance to jamming), and deploy network monitoring to detect interference.

114. Command Injection

- **How it Works**: Command injection occurs when an attacker inserts malicious code into a system command via an input field, which the system executes. This allows attackers to execute arbitrary commands on the server.
- **Prevention**: Use input validation, parameterized commands, and ensure that user input cannot directly control system commands or script execution.

115. Social Media Hacking

- **How it Works**: Attackers target social media accounts by guessing passwords, stealing login credentials via phishing or data breaches, or exploiting vulnerabilities in social media platforms.
- **Prevention**: Use multi-factor authentication (MFA), employ strong password policies, and be cautious of phishing and suspicious links on social media.

116. Botnet DDoS Attacks

- **How it Works**: In a botnet DDoS attack, a network of compromised devices (bots) is used to generate massive amounts of traffic to overwhelm and disable a target website or service.
- **Prevention**: Use anti-DDoS mitigation services, deploy rate-limiting strategies, and utilize content delivery networks (CDNs) to absorb traffic spikes.

117. Session Fixation

- **How it Works**: In a session fixation attack, the attacker forces a victim to use a known session ID, which the attacker can then hijack to gain unauthorized access to the victim’s session.
- **Prevention**: Regenerate session IDs after login, enforce strong session timeouts, and ensure secure cookie handling with flags like HttpOnly and Secure.

118. Fake Antivirus Software

- **How it Works**: Fake antivirus software pretends to be a legitimate security program but actually performs malicious activities like stealing personal data or installing malware.
- **Prevention**: Download security software only from trusted sources, use reputable antivirus tools, and regularly update and scan systems for malware.

119. Man-in-the-Cloud Attacks

- **How it Works**: In a Man-in-the-Cloud (MITC) attack, attackers gain access to cloud storage accounts by exploiting synchronization processes or vulnerabilities in cloud storage apps, intercepting data exchanges between users and the cloud.
- **Prevention**: Use end-to-end encryption for cloud storage, enable multi-factor authentication, and ensure that cloud storage apps are configured securely.

120. DNS Flood

- **How it Works**: DNS flood attacks are a type of DoS attack where attackers send a large number of DNS requests to a DNS server, overwhelming it and preventing legitimate users from accessing services.
- **Prevention**: Implement rate limiting, deploy specialized DNS protection services, and use Anycast to distribute DNS queries across multiple servers.

121. Hypervisor Attacks

- **How it Works**: Hypervisor attacks target the virtualization layer that manages virtual machines (VMs). Attackers can exploit vulnerabilities in hypervisors to escape the virtual machine and gain access to the host system or other VMs running on the same hypervisor.
- **Prevention**: Use strong access controls, keep hypervisor software updated, isolate VMs with strict network segmentation, and apply security patches promptly.

122. Firmware Vulnerabilities

- **How it Works**: Firmware vulnerabilities allow attackers to manipulate the underlying hardware firmware of a device to install rootkits, disable security features, or create backdoors that persist even after the system is rebooted.
- **Prevention**: Regularly update firmware, implement secure boot processes, use hardware-backed security features like TPM (Trusted Platform Module), and monitor for firmware integrity.

123. Falsified Digital Certificates (Man-in-the-Middle via Certificate Forgery)

- **How it Works**: Attackers can forge or steal digital certificates to impersonate legitimate websites and decrypt encrypted communications, allowing them to conduct man-in-the-middle attacks and steal sensitive information.
- **Prevention**: Use certificate pinning, enforce strict SSL/TLS certificates validation, and implement public key infrastructure (PKI) for robust certificate management.

124. Exploit Kits

- **How it Works**: Exploit kits are toolkits used by attackers to deliver malware by exploiting known vulnerabilities in software. Once the exploit kit successfully identifies a vulnerability, it triggers the malware to be installed.
- **Prevention**: Apply regular security patches, use updated browsers, install antivirus software, and disable unnecessary plugins and scripting languages.

125. Rootkit Installation

- **How it Works**: A rootkit is a malicious program designed to gain administrative access to a system and hide its presence. It allows attackers to control a system remotely, steal data, and maintain persistent access.
- **Prevention**: Use behavior-based detection systems, run regular malware scans, and implement full disk encryption to detect unauthorized access.

126. Zombie Email Attack (Email Bombing)

- **How it Works**: In email bombing, attackers send an overwhelming number of emails to a target's inbox to cause disruption or overflow the system, making it difficult for the victim to access legitimate communication.
- **Prevention**: Implement spam filters, use email rate limiting, and monitor inbound email traffic for unusual volumes or patterns.

127. Privilege Escalation via Sudo

- **How it Works**: Attackers exploit misconfigurations or flaws in the sudo command used in Linux/Unix-based systems to escalate their privileges from normal user to root access.
- **Prevention**: Regularly audit sudo configurations, restrict sudo usage to trusted users, and ensure proper configuration of system permissions.

128. RAT (Remote Access Trojan)

- **How it Works**: RATs are a type of malware that provides an attacker with complete control over a victim's system. Attackers can use RATs to spy on the victim, steal data, or use the infected machine as a launchpad for further attacks.
- **Prevention**: Use firewalls, endpoint protection, and ensure regular software updates. Avoid downloading files from untrusted sources.

129. Password Hash Cracking

- **How it Works**: Password hash cracking involves using techniques like brute force or dictionary attacks to decrypt stored password hashes and gain unauthorized access to systems.
- **Prevention**: Use strong hashing algorithms (e.g., bcrypt, Argon2), employ salting techniques, and enforce strong password policies.

130. DNS Tunneling for Data Exfiltration

- **How it Works**: DNS tunneling allows attackers to encode data into DNS queries, bypassing traditional network security measures, and exfiltrating sensitive information from a compromised system.
- **Prevention**: Monitor DNS traffic for unusual patterns, implement DNS filtering solutions, and use encrypted communication channels like VPNs to secure data transfers.

131. Cross-Site Scripting (XSS) via WebSockets

- **How it Works**: WebSocket XSS occurs when attackers inject malicious JavaScript into WebSocket messages that will be processed by the target's browser, leading to data theft or session hijacking.
- **Prevention**: Sanitize WebSocket inputs, use secure WebSocket connections (WSS), and implement content security policies (CSP).

132. TCP/IP Hijacking

- **How it Works**: TCP/IP hijacking is an attack where the attacker intercepts and manipulates TCP/IP sessions between a client and a server, allowing them to assume control of an active connection.

- **Prevention**: Use secure tunneling protocols like SSL/TLS, implement IPsec for secure communication, and deploy intrusion detection/prevention systems (IDS/IPS) to detect suspicious network activities.

133. Cold Boot Attack

- **How it Works**: A cold boot attack exploits vulnerabilities in the physical memory (RAM) of a system by freezing the system and quickly rebooting it. This allows attackers to recover encryption keys or other sensitive data stored in RAM.
- **Prevention**: Use full-disk encryption, ensure system shutdown properly clears sensitive data from RAM, and use physical security measures like screen locks and access control.

134. Redirection via HTTP Response Splitting

- **How it Works**: HTTP response splitting allows attackers to inject malicious headers into an HTTP response, which can be used to redirect users to a malicious site or perform cache poisoning.
- **Prevention**: Sanitize all input fields, validate HTTP headers, and implement strict content security policies (CSP).

135. Cryptographic Attacks (Brute Force)

- **How it Works**: Cryptographic brute force attacks involve systematically trying every possible key combination to decrypt encrypted data. This is effective if weak encryption or short keys are used.
- **Prevention**: Use strong encryption algorithms with long key lengths (e.g., AES-256) and implement rate limiting on decryption attempts.

136. DNS Amplification Attack

- **How it Works**: In a DNS amplification attack, attackers exploit misconfigured DNS servers to generate large responses that are directed at the target, overwhelming the target's network infrastructure.
- **Prevention**: Configure DNS servers to reject requests from unauthorized IP addresses and implement rate limiting and anti-DDoS protection.

137. Bluetooth Hacking (Bluejacking, Bluesnarfing)

- **How it Works**: Bluetooth hacking attacks can involve Bluejacking, where unsolicited messages are sent to Bluetooth-enabled devices, or Bluesnarfing, where attackers gain unauthorized access to data on Bluetooth devices.
- **Prevention**: Disable Bluetooth when not in use, enable Bluetooth visibility only when necessary, and ensure devices use secure pairing and encryption.

138. DNS Cache Poisoning

- **How it Works**: DNS cache poisoning attacks target DNS resolvers by injecting fake DNS records into the cache, causing users to be redirected to malicious websites when trying to access legitimate domains.
- **Prevention**: Implement DNSSEC (DNS Security Extensions), use DNS filtering, and ensure DNS caches are periodically cleared and validated.

139. Eavesdropping on Wireless Networks

- **How it Works**: Wireless network eavesdropping involves intercepting data being transmitted over an unsecured Wi-Fi network. Attackers can steal sensitive data such as passwords, email contents, and other personal information.
- **Prevention**: Use WPA3 encryption for wireless networks, disable open networks, and use VPNs to encrypt traffic over Wi-Fi.

140. Clickjacking via iFrames

- **How it Works**: Clickjacking involves embedding a transparent, malicious frame on a legitimate website. When a user clicks on a seemingly harmless element, the attacker's hidden frame triggers malicious actions without the user's knowledge.
- **Prevention**: Use frame-busting JavaScript to prevent the site from being embedded in an iframe, and implement Content Security Policy (CSP) to restrict page embedding.

141. Spyware

- **How it Works**: Spyware is a type of malware that collects sensitive information about a user or organization without their knowledge. This can include personal data, browsing habits, or login credentials.
- **Prevention**: Use anti-spyware tools, ensure operating systems and applications are regularly updated, and educate users about the risks of downloading malicious software.

142. Man-in-the-Middle (MITM) via SSL Stripping

- **How it Works**: SSL stripping is a type of MITM attack where attackers downgrade a secure HTTPS connection to an unencrypted HTTP connection, allowing them to intercept and manipulate the data.
- **Prevention**: Use HTTP Strict Transport Security (HSTS) to enforce secure connections and avoid accepting untrusted certificates.

143. Trojans via Malicious Software Downloads

- **How it Works**: Trojans are typically distributed as malicious software downloads disguised as legitimate files. Once installed, they can steal data, create backdoors, or spread malware further.
- **Prevention**: Avoid downloading software from untrusted sources, use endpoint protection software, and educate users on recognizing suspicious downloads.

144. Cryptojacking via Browser Mining

- **How it Works**: In cryptojacking via browser mining, attackers use a victim's web browser to mine cryptocurrency without their consent. This is often done through malicious JavaScript embedded in websites or ads.
- **Prevention**: Use ad blockers, block crypto mining scripts, and ensure web browsers are updated with security patches.

145. SQL Injection via Blind Injection

- **How it Works**: In a blind SQL injection attack, attackers inject SQL queries that return no direct output, relying on inference (such as timing) to extract data or modify database records.
- **Prevention**: Use parameterized queries, implement stored procedures, and validate and sanitize all user inputs.

146. Session Hijacking via Session Cookies

- **How it Works**: In session hijacking, attackers steal a valid session cookie from a victim to gain unauthorized access to their online session, such as logging into an account without the victim's knowledge.
- **Prevention**: Use secure (HTTPS) communication to prevent cookie interception, implement session timeout policies, and use HttpOnly and Secure flags for session cookies to prevent client-side access.

147. Code Injection

- **How it Works**: Code injection attacks occur when attackers insert malicious code into a vulnerable application. This code is then executed by the system, leading to potential control over the system, data breaches, or privilege escalation.
- **Prevention**: Implement proper input validation and output encoding, use secure coding practices, and apply least privilege access control policies.

148. SQL Injection via Union Select

- **How it Works**: SQL injection via Union Select allows attackers to combine results from multiple queries into a single response. By injecting a malicious "UNION" SQL query, attackers can extract data from other database tables.
- **Prevention**: Use parameterized queries, limit user input in database queries, and employ web application firewalls (WAF) to block suspicious query patterns.

149. IP Spoofing in DDoS Attacks

- **How it Works**: In IP spoofing, attackers falsify the source IP address of packets to make them appear as though they are from a trusted source. This can be used in Distributed Denial of Service (DDoS) attacks to overwhelm a target network.
- **Prevention**: Use ingress and egress filtering, configure routers to block traffic from spoofed IP addresses, and deploy anti-DDoS solutions to absorb large volumes of traffic.

150. ARP Cache Poisoning

- **How it Works**: In ARP cache poisoning, attackers send false ARP messages to a local network, causing devices to associate their MAC address with the IP address of a legitimate device, allowing attackers to intercept or redirect network traffic.
- **Prevention**: Use static ARP entries, monitor network traffic for suspicious ARP activity, and deploy ARP spoofing detection tools.

151. Malicious Insider Attacks

- **How it Works**: Malicious insiders exploit their access to systems, applications, or data to intentionally cause harm, steal data, or disrupt operations. This could involve leaking sensitive information or sabotaging systems.
- **Prevention**: Implement strong access controls, monitor user activity, and establish a clear policy for insider threat management, including reporting mechanisms and security training.

152. Web Shells

- **How it Works**: Web shells are scripts or programs uploaded to a web server that allow an attacker to remotely execute commands. This enables them to steal data, launch further attacks, or modify the server's configuration.
- **Prevention**: Use secure file upload mechanisms, implement file integrity monitoring, and scan web applications for unauthorized files and scripts.

153. Ransomware-as-a-Service (RaaS)

- **How it Works**: Ransomware-as-a-Service is a business model where cybercriminals provide ransomware tools to other criminals in exchange for a percentage of the ransom payment. This makes ransomware attacks accessible even to non-technical attackers.
- **Prevention**: Regularly back up critical data, implement endpoint protection with real-time scanning, and train employees to identify phishing emails, which are a common vector for ransomware delivery.

154. Web Application Vulnerabilities (OWASP Top 10)

- **How it Works**: The OWASP Top 10 refers to the most critical web application security risks, such as injection flaws, broken authentication, and insecure direct object references. Attackers exploit these vulnerabilities to compromise web applications and steal data.
- **Prevention**: Follow secure coding practices, implement input validation, use security frameworks and libraries, and regularly test applications using penetration testing and static analysis tools.

155. DNS Rebinding Attack

- **How it Works**: In a DNS rebinding attack, an attacker manipulates the DNS resolution process to make a victim's browser connect to an internal system (like a router or internal network service) without the victim's knowledge.
- **Prevention**: Restrict the use of DNS and configure servers to only allow connections to trusted internal addresses, and apply strict Same-Origin Policy (SOP) in web browsers.

156. Exploitation of Public Key Infrastructure (PKI)

- **How it Works**: Attackers exploit flaws in public key infrastructure (PKI), such as the improper issuance of certificates or the use of weak encryption algorithms, to impersonate legitimate websites or systems and decrypt sensitive communications.
- **Prevention**: Use strong encryption algorithms, monitor and audit the issuance of digital certificates, and ensure that all PKI components are properly configured and secured.

157. SIM Swapping

- **How it Works**: SIM swapping occurs when an attacker tricks a mobile carrier into transferring a victim's phone number to a SIM card in the attacker’s possession. This allows the attacker to intercept two-factor authentication (2FA) codes and gain access to accounts.
- **Prevention**: Use multi-factor authentication that doesn't rely solely on SMS, alert users to changes in SIM card details, and secure mobile accounts with strong PINs and account recovery processes.

158. Privilege Escalation via Setuid

- **How it Works**: Attackers exploit vulnerabilities in the Unix/Linux setuid function, which allows a program to execute with the privileges of the file owner. This can lead to privilege escalation if exploited by malicious actors.
- **Prevention**: Regularly audit setuid binaries, restrict use of setuid files, and follow the principle of least privilege to minimize access rights.

159. Botnet as a Service (BaaS)

- **How it Works**: Botnet as a Service enables cybercriminals to rent access to botnets for DDoS attacks, sending spam emails, or launching malware. This allows attackers to perform high-impact attacks without having to maintain their own botnet infrastructure.
- **Prevention**: Implement DDoS mitigation services, use network traffic monitoring tools, and ensure that devices are protected with up-to-date security patches and endpoint protection.

160. Social Engineering via Pretexting

- **How it Works**: Pretexting is a form of social engineering in which an attacker creates a fabricated scenario or pretext to trick the victim into providing sensitive information, such as personal identification numbers or access credentials.
- **Prevention**: Educate employees about common social engineering tactics, implement identity verification procedures, and discourage sharing sensitive information over the phone or email.

161. Zero-Day Malware

- **How it Works**: Zero-day malware exploits vulnerabilities that are unknown to the software vendor and have not yet been patched. Because there is no fix, these attacks can be extremely effective and difficult to detect.
- **Prevention**: Implement endpoint protection systems that use behavioral analysis to detect unknown threats, regularly update software, and apply patches as soon as they are released.

162. Email Spoofing via Display Name

- **How it Works**: Email spoofing involves sending emails that appear to come from a trusted source by manipulating the "display name" field, which can trick users into opening malicious emails or downloading attachments.
- **Prevention**: Use email authentication methods like SPF, DKIM, and DMARC, and train users to verify the sender’s email address before opening attachments or clicking links.

163. Exploitation of IoT Devices

- **How it Works**: IoT devices (smart cameras, thermostats, etc.) often have weak security protocols, such as default passwords or unpatched firmware. Attackers can exploit these vulnerabilities to gain access to sensitive data or launch attacks like DDoS.
- **Prevention**: Change default credentials, regularly update IoT device firmware, and isolate IoT devices on separate networks from critical infrastructure.

164. Pharming via DNS Spoofing

- **How it Works**: Pharming involves redirecting users to malicious websites without their knowledge, often by manipulating DNS queries. DNS spoofing is commonly used to alter the resolution of domain names to point to a fake website.
- **Prevention**: Use DNSSEC to ensure DNS integrity, monitor DNS traffic for unusual patterns, and implement secure communication protocols like HTTPS.

165. Fake Technical Support Scam

- **How it Works**: Fake technical support scams involve attackers impersonating legitimate tech support agents from well-known companies. They convince victims to grant remote access to their systems or pay for unnecessary services.
- **Prevention**: Educate users not to trust unsolicited phone calls, verify the legitimacy of any tech support service before granting access, and report suspicious activity to relevant authorities.

166. Business Email Compromise (BEC)

- **How it Works**: Business Email Compromise is a sophisticated scam where attackers impersonate an executive or employee to request wire transfers or access to confidential business information, often by compromising email accounts.
- **Prevention**: Implement multi-factor authentication (MFA) for email accounts, train employees to recognize phishing and social engineering attempts, and use email filtering tools to block suspicious emails.

167. Firmware Rootkit (Hardware Rootkit)

- **How it Works**: A firmware rootkit is a type of malicious software that targets the firmware of a hardware device. Once installed, it allows attackers to control the hardware, bypassing normal security measures and remaining persistent even after system reboot.
- **Prevention**: Use hardware security modules (HSM), verify firmware integrity, and ensure systems are equipped with trusted boot mechanisms to prevent unauthorized firmware installation.

168. Malware Dropper

- **How it Works**: A malware dropper is a type of malicious software designed to deliver and install other malware onto a victim’s system. The dropper itself may be relatively harmless but opens the door for more dangerous attacks, such as ransomware or spyware.
- **Prevention**: Use comprehensive endpoint protection software, block suspicious downloads, and implement advanced behavioral detection techniques that identify dropper behavior.

169. Key Reinstallation Attack (KRACK)

- **How it Works**: KRACK exploits a vulnerability in the WPA2 protocol used to secure Wi-Fi networks. It allows attackers to intercept and decrypt Wi-Fi traffic by reinstalling an already used key, compromising the confidentiality of the encrypted communication.
- **Prevention**: Update Wi-Fi routers and client devices with the latest security patches, and use WPA3, which provides stronger security than WPA2.

170. Whaling

- **How it Works**: Whaling is a type of phishing attack that specifically targets high-level executives or important personnel in an organization. The attacker crafts highly targeted and personalized messages, often appearing as legitimate business communications, to deceive the victim into revealing confidential information or authorizing a financial transaction.
- **Prevention**: Implement multi-factor authentication (MFA) for executives, train employees to recognize phishing attempts, and use email filters to detect and block suspicious emails.

171. Software Supply Chain Attack

- **How it Works**: In a software supply chain attack, attackers compromise software or its updates by injecting malicious code into legitimate software updates or third-party software components. This type of attack can affect a wide range of organizations that rely on the same software.
- **Prevention**: Regularly audit software updates, use trusted vendors, monitor code repositories for suspicious activity, and employ security scanning tools to check software integrity before deployment.

172. Web Application Firewall (WAF) Bypass

- **How it Works**: WAF bypass techniques involve exploiting weaknesses in the configuration of Web Application Firewalls (WAF) to circumvent security measures that would otherwise block malicious web traffic. This could include obfuscating malicious payloads or using vulnerabilities in WAF logic.
- **Prevention**: Ensure proper configuration and regular updating of WAF systems, conduct penetration testing, and use web application security scanners to identify and fix bypass vulnerabilities.

173. Deserialization Attacks

- **How it Works**: A deserialization attack occurs when attackers exploit a flaw in the deserialization process of an application to inject malicious objects into the application. This allows attackers to execute arbitrary code and gain control over the application or system.
- **Prevention**: Avoid deserializing untrusted data, use strong validation and integrity checks on serialized data, and apply secure coding practices like using whitelisted classes for deserialization.

174. Drive-By Downloads via Malicious Ads

- **How it Works**: Drive-by downloads happen when a user unknowingly downloads malware by simply visiting a compromised website or viewing a malicious ad. These ads can automatically trigger downloads without the user's consent or interaction.
- **Prevention**: Use ad blockers, implement browser security features like pop-up blocking, ensure that browsers and plugins are up to date, and employ anti-malware solutions to detect and block malicious ads.

175. Credential Harvesting

- **How it Works**: Credential harvesting involves stealing user credentials (like usernames and passwords) using methods such as phishing, social engineering, or malware. Attackers may use these credentials to access user accounts, conduct identity theft, or launch further attacks.
- **Prevention**: Use MFA to protect accounts, encourage the use of strong, unique passwords, and educate users on how to recognize phishing attempts and avoid suspicious websites.

176. Bluetooth Low Energy (BLE) Attacks

- **How it Works**: BLE attacks target devices that use Bluetooth Low Energy for communication, exploiting vulnerabilities in Bluetooth protocols to either gain unauthorized access to the device or intercept data being transmitted.
- **Prevention**: Disable Bluetooth when not in use, use strong encryption for Bluetooth connections, and ensure devices are updated with the latest firmware patches to mitigate known vulnerabilities.

177. Botnet Proxying

- **How it Works**: In botnet proxying, attackers use compromised devices in a botnet to proxy malicious traffic or hide the origin of an attack. This can be used to launch attacks such as DDoS or to anonymize other criminal activities.
- **Prevention**: Monitor network traffic for unusual patterns, implement strong security on IoT devices, and deploy DDoS mitigation tools to prevent attacks from botnets.

178. Cross-Site Scripting (XSS) via DOM Manipulation

- **How it Works**: DOM-based XSS attacks manipulate the DOM (Document Object Model) of a web page, leading to the execution of malicious scripts in the user's browser without affecting the server. This type of XSS attack is executed in the victim's browser, and it can bypass traditional server-side security measures.
- **Prevention**: Sanitize user input, use secure coding practices like input validation and output encoding, and employ content security policies (CSP) to prevent the execution of unauthorized scripts.

179. HTTP Response Splitting

- **How it Works**: HTTP response splitting is an attack that allows an attacker to inject headers or modify HTTP responses sent by a web server. This can lead to cache poisoning, redirect users to malicious sites, or manipulate web applications.

- **Prevention**: Validate all user inputs, prevent newlines or control characters from being passed in HTTP headers, and configure web servers to sanitize responses before sending them to clients.

180. Time-of-Check to Time-of-Use (TOCTOU) Vulnerability

- **How it Works**: TOCTOU vulnerabilities occur when an attacker exploits the time gap between when a system checks a resource and when it uses that resource. The attacker can modify the resource in between these two operations to perform malicious actions.
- **Prevention**: Use atomic operations, synchronize resource access properly, and implement proper access control to ensure that time-sensitive actions are protected from manipulation.

181. Malicious USB Devices (USB Drop Attacks)

- **How it Works**: In USB drop attacks, attackers intentionally leave infected USB drives in public areas, hoping that someone will plug them into a computer. Once inserted, the USB device can spread malware, steal data, or gain unauthorized access to systems.
- **Prevention**: Disable USB ports when not needed, implement USB device control software, and educate users on the risks of inserting unknown USB devices into systems.

182. Evil Twin Attacks

- **How it Works**: An evil twin attack occurs when an attacker sets up a rogue wireless access point with the same SSID as a legitimate network. Victims unknowingly connect to the rogue access point, allowing the attacker to intercept data, perform man-in-the-middle attacks, or inject malware.
- **Prevention**: Use WPA3 encryption for Wi-Fi networks, avoid connecting to untrusted public networks, and use a VPN to encrypt traffic when using public Wi-Fi.

183. Cache Poisoning

- **How it Works**: Cache poisoning involves inserting malicious content into a website's cache. Once cached, this content is served to users, potentially leading to data theft, session hijacking, or the spreading of malware.
- **Prevention**: Use proper cache control mechanisms, regularly validate cache entries, and implement a web application firewall (WAF) to detect suspicious content in caches.

184. Keylogging via Software or Hardware

- **How it Works**: Keylogging is a technique where an attacker records every keystroke a user makes, typically to capture sensitive information like passwords and credit card numbers. This can be achieved either through malicious software or by using physical devices.
- **Prevention**: Use on-screen keyboards for entering sensitive information, employ endpoint protection software, and educate users about the risks of using untrusted devices or software.

185. Side-Channel Attacks

- **How it Works**: Side-channel attacks involve extracting information from a system by measuring physical attributes such as power consumption, electromagnetic emissions, or even the time taken to execute certain operations. These attacks can help break cryptographic algorithms or expose sensitive data.

- **Prevention**: Implement countermeasures such as noise generation, use hardware security modules (HSMs), and limit the physical exposure of systems handling sensitive data.

186. Credential Stuffing via Automated Tools

- **How it Works**: Credential stuffing involves using automated tools to try various username and password combinations, often from previous data breaches, to gain unauthorized access to multiple accounts.
- **Prevention**: Use CAPTCHA to block automated login attempts, implement rate-limiting and account lockout mechanisms, and encourage users to use strong, unique passwords across different services.

187. Buffer Overflow in Web Applications

- **How it Works**: Buffer overflow attacks exploit vulnerabilities where an application writes more data to a buffer than it can handle. This allows attackers to overwrite adjacent memory locations, leading to arbitrary code execution or a system crash.
- **Prevention**: Use bounds checking, implement secure coding techniques, and enable stack protection mechanisms like stack canaries and ASLR (Address Space Layout Randomization).

188. Rogue Security Software

- **How it Works**: Rogue security software masquerades as legitimate antivirus or system optimization tools. Once installed, it often displays false alerts to convince users to pay for unnecessary services, or it may install actual malware.
- **Prevention**: Download security software only from trusted sources, check reviews and ratings before installation, and use trusted endpoint protection to detect and block rogue software.

189. Exploit of Vulnerable APIs

- **How it Works**: Exploiting vulnerabilities in web or software APIs (Application Programming Interfaces) allows attackers to bypass authentication mechanisms, retrieve sensitive data, or perform unauthorized actions within the application.
- **Prevention**: Use strong authentication for APIs (e.g., OAuth, API keys), implement input validation and rate limiting, and regularly audit API security for vulnerabilities.

190. SSL/TLS Stripping

- **How it Works**: SSL/TLS stripping attacks downgrade an HTTPS connection to HTTP, allowing attackers to intercept and manipulate data transmitted between a client and server. This is often used in man-in-the-middle attacks.
- **Prevention**: Use HTTP Strict Transport Security (HSTS) to enforce secure connections and implement certificate pinning to avoid SSL/TLS certificate manipulation.

191. Cross-Site Request Forgery (CSRF) via Token Invalidation

- **How it Works**: CSRF attacks exploit the trust a website has in the user’s browser by tricking a victim into performing actions (like changing account settings or making transactions) without their consent. In token invalidation CSRF, attackers invalidate a valid CSRF token to exploit the vulnerability, enabling unauthorized actions to be performed.

- **Prevention**: Use anti-CSRF tokens for every state-changing request, and ensure tokens are validated with each request. Implement SameSite cookie attributes and avoid reliance on GET requests for sensitive actions.

192. Subdomain Takeover

- **How it Works**: Subdomain takeover occurs when an attacker identifies an unclaimed or misconfigured DNS record that points to an external service (like a cloud provider or a third-party hosting service). The attacker can claim the service and take control of the subdomain.
- **Prevention**: Regularly audit DNS records, remove unneeded or unused DNS entries, and monitor subdomain configurations to ensure they are linked to valid, secure services.

193. Bluetooth Man-in-the-Middle (MITM)

- **How it Works**: In a Bluetooth MITM attack, an attacker intercepts communication between two Bluetooth-enabled devices. The attacker can eavesdrop, alter data, or inject malicious code between the devices.
- **Prevention**: Ensure Bluetooth devices use strong encryption protocols (e.g., AES), disable Bluetooth when not in use, and pair devices only with trusted devices. Implement security patches for Bluetooth vulnerabilities.

194. Credential Stuffing via Botnets

- **How it Works**: Credential stuffing attacks involve using automated bots to test stolen username and password combinations on websites and services. Botnets, which are networks of compromised devices, are commonly used to scale these attacks and mask the attacker’s IP address.
- **Prevention**: Implement multi-factor authentication (MFA), use CAPTCHA to block bot traffic, limit login attempts, and use machine learning-driven solutions to identify and block bot activity.

195. Steganography in Malware

- **How it Works**: Steganography involves hiding data (such as malicious code or instructions) inside legitimate files, such as images, audio files, or videos, to avoid detection by traditional security tools.
- **Prevention**: Use security tools that can detect anomalies in files, employ advanced malware detection methods such as behavior analysis, and train users to avoid opening suspicious files from untrusted sources.

196. Mimikatz (Credential Dumping)

- **How it Works**: Mimikatz is a well-known post-exploitation tool that allows attackers to dump credentials from Windows systems, including clear-text passwords, Kerberos tickets, and password hashes. This tool is used after an attacker has gained access to a system to escalate privileges or perform lateral movement across a network.
- **Prevention**: Disable Windows SMBv1, enforce strong password policies, use least privilege access, implement credential guard tools, and monitor for suspicious network activity or tool execution.

197. Cross-Site Scripting (XSS) via DOM Clobbering

- **How it Works**: DOM clobbering is a form of XSS where an attacker can manipulate the DOM (Document Object Model) of a webpage. By overwriting document properties with malicious values, the attacker can inject scripts that execute in the victim's browser when the page is loaded.
- **Prevention**: Sanitize user inputs, avoid directly manipulating DOM elements with untrusted data, and implement content security policies (CSP) to restrict script execution.

198. File Inclusion Vulnerabilities (LFI/RFI)

- **How it Works**: Local File Inclusion (LFI) and Remote File Inclusion (RFI) vulnerabilities allow attackers to include files on a web server through the input of a URL or file path. LFI can lead to the exposure of sensitive files like configuration files, while RFI allows attackers to include external files, potentially executing malicious code.
- **Prevention**: Sanitize user inputs, use whitelist file inclusion, implement strict access control on file system access, and configure the server to block unsafe file inclusion practices.

199. DNS Flooding

- **How it Works**: DNS flooding is a type of DDoS attack that targets DNS servers by overwhelming them with a massive number of DNS queries. The goal is to exhaust server resources, causing legitimate requests to be delayed or denied.
- **Prevention**: Use rate limiting, anycast DNS, and deploy anti-DDoS services to filter malicious traffic. Ensure DNS servers are resilient and capable of handling large volumes of traffic.

200. LDAP Injection

- **How it Works**: LDAP injection is an attack where attackers manipulate input data to alter the structure of an LDAP query. This can allow them to bypass authentication, retrieve sensitive data, or even modify directory contents.
- **Prevention**: Sanitize and validate user inputs, use prepared statements for LDAP queries, and implement strict access controls on directory services.