**Module 11: Session Hijacking**

**Concept**

* An attack in which an attacker seizes control of **a valid TCP communication session** between two computers
* As most authentication only occur at the **start of a TCP session**, allowing the attacker…
* Attackers can sniff all the traffic from the established TCP sessions and perform identify theft, info theft, traud, etc.
* Steal a valid session ID and use it to authenticate himself

**Why is Session Hijacking Successful**

* Absence of account lockout for **invalid session IDs**
* Weak **session-ID generation algorithm** or small session IDs
* **Insecure handling** of session IDs
* Indefinite **session timeout**
* Most computeres using **TCP/IP are vulnerable**
* Most countermeasures do not work without encryption

**Process**

* Sniff
* Monitor
* Session Desynchronization
* Session ID prediction
* Command Injection

**Types of Session Hijacking**

* **Passive:** The attacker hijacks a session but sits back, watches, and records all the traffic.
* **Active:** The attacker finds an active session and seizes control of it.

**Session Hijacking in OSI Model**

* **Network Level Hijacking:** Can be defined as the interception of packets during the transmission between a client and the server in a TCP or UDP session
* **Application Level Hijacking:** Gain control over the HTTP’s user session by obtaining the session IDs

**Application Level Session Hijacking**

* A session is stolen or a valid session taoken is predicted **to gain unauthorized access** to the web server
* A session token can be compromised in various ways
  + **Session sniffing:** Use a sniffer to capture a valid session token or session ID
  + **Predictable session token:** Predict session IDs generated by weak algorithms and impersonate a website user.
  + **MITM attack**
  + **Man-in-the-browser attack:** Use a trojan horse to intercept the calss between the browser and its security mechanisms or libs
  + **XSS:** Inject malicious client-side scripts into the web pages…
  + **CSRF:** Exploit the victim’s active session with a trusted site to perform malicious activities
  + **Session replay attack:** Use the authentication token to replay the request
  + **Session fixation attack:** Attackers provide a valid SID to a victim and lure him to authenticate using that SID.
  + **CRIME (Compression Ratio Info-Leak Made Easy) attack:** A client-side attack that exploit the vulnerabilities present in the data compression feature of protocols, such as SSL/TLS, SPDY, and HTTPS.
  + **Forbidden attack:** A type of MITM. Exploit the reuse of cryptographic nonce during the TLS handshake
  + **Session donation attack:** An attacker donates his own SID to the target user.

**Network Level Session Hijacking**

* Rely on hijacking transport and Internet Protocols used by web applications in the application layer
* **Blind hijacking:** Inject malicious data or cmds into the intercepted communications in the TCP session even if the source-routing is disabled. But **the attacker cannot see the response**
* **UDP hijacking:** Attackers send **forged server reply** to a civtim’s UDP request before the intended server replies to it.
* **TCP/IP hijacking:** Use spoofed packets to seize control of a connection. The attacker must be be on the same network as the victim.
* **RST hijacking:** Inject an authentic-looing RST packet using a spoofed source address and predict the ack number. An attacker can reset a victim’s connection if it uses an accurate ack num
* **MITM: Packet sniffer:** Change the **default gateway** of the client’s gateway and attempt to reroute packets. The packets between the C and S are routed through the hijacker’s host using **Forged ICMP** and **ARP Spoofing**.
* **IP spoofing: Source routed packets:** Spoof the host’s IP so that the server managing a session with the host accepts the packets from the attacker.

**Session Hijacking Tools:**

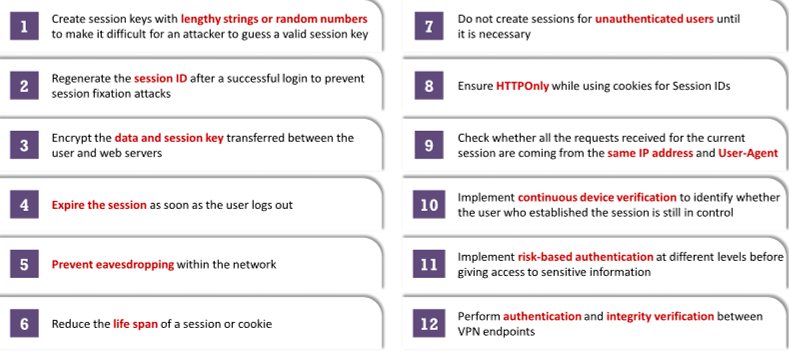
* Burpsuite
* OWASP ZAP
* bettercap
* ssltrip

**Countermeasures**

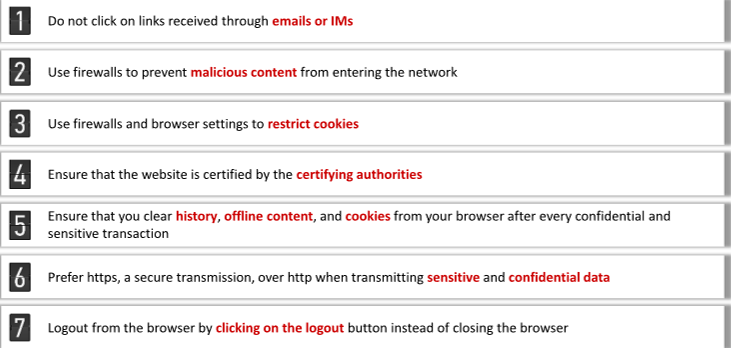
* Detection Method
  + Manual Method
    - Using packet sniffing software
  + Automatic Method
    - IDS
    - IPS
* Protect against Session Hijacking



* Web Dev Guielines to Prevent Session Hijacking



* Web User Guidelines to Prevent Session Hijacking



* Detection Tools
  + AlienVault USM
  + Wireshark
* Approaches Causing Vulnerability to Session Hijacking and Their Preventative Solutions
  + Telnet, rlogin -> OpenSSH or ssh
  + FTP -> SFTP, AS2, MFT, FTPS
  + HTTP -> SSL or TLS
  + IP -> IPsec
  + Any remote connection -> VPN
  + SMB -> SMB signing
  + Hub Network -> Switch Network
* Approaches to Prevent Session Hijacking
  + **HSTS (HTTP StrictTransport Security):** A **web security policy** that protects HTTPs against MITM. Allow web server to **enforce web browsers** to interact with it using HTTPS
  + **Token Binding:** When a user logs on a web app, it generates a cookie with an **SID**, called **token**.
  + **HPKP (HTTP Public Key Pinning):** A **TOFU (Trust on First Use)** technique used in an HTTP header. Allow a web client to **associate a specific public key certificate** with a particular server to minimize the risk of MITM
* Approaches to Prevent MITM Attacks
  + WEP/WPA Encryption
  + VPN
  + TFA
* **IPSec**
  + A protocol suite developed by the IETF for **securing IP communication** by **authenticating and encrypting** each IP packet of a session.
  + Deployed widely to implement **VPNs** and for **remote user access** through dial-up connection to pricate networks
  + **Components:**
    - **IPsec Driver:** Software that performs protocol-level functions required to encrypt and decrypt packets
    - **IKE (Internet Key Exchange):** An protocol that produces security keys for IPsec and other protocols
    - **ISAKMP (Internet Securtity Association Key Management Protocol):** Software that allows two computers to communicate by encrypting the data exchanged between them
    - **Oakley:** A protocol that uses the **Diffie-Hellman algorithm** to create a master key and a key that is specific to each session in IPsec data transfer
    - **IPsec Policy Agent:** A service included in Windows OS that enforces IPsec policies for all the network communications initiated from that system
  + **Benefits**
    - Network-level peer authentication
    - Data origin authentication
    - Data integrity
    - Date confidentiality
    - Replay protection
  + Modes of IPsec
    - **Transport Mode:** Also ESP, encrypts only the **payload** of the IP packet, leaving the header untouched.
    - Tunnel Mode: Also AH, encrypt **both the payload and header**. More secure than the transport mode.
  + IPsec Architecture:
    - **AH (Authentication Header):** Offer **integrity** and data origin authentication, with optional anti-replay features.
    - **ESP (Encapsulating Security Payload):** Offer all the services offered by AH as well as **confidentiality**
    - **DOI (IPsec Domain of Interpretation):** Define the payload formats, types of exchange, and naming conventions for security info such as cryptographic algorighms or security policies
    - **ISAKMP (Internet Security Association and Key Management Protocol):** A key protocol in the IPsec architecture that establishes the required security for various communications over the Internet, such as gov, private, and com communications.
    - **Policy:** Define when and how to secue data, as well as security methods to use at differnet level in the network.
* Session Hijacing Prevention Tools:
  + CxSAST
  + Fiddler