In session hijacking attack, a session token is stolen or a valid session token is predicted to gain unauthorized access to the web server

A session token can be compromised in various ways

1. Session sniffing
2. Predictable session token
3. Man-in-the-middle attack
4. Man-in-the-browser attack
5. Cross-site scripting (XSS) attack
6. Cross-site Forgery attack
7. Session replay attack
8. Session fixation attack
9. CRIME attack
10. Forbidden attack

Session hijacking as defined focuses on the application layer of the OSI model. In the application layer hijacking process, the attacker is looking for a legitimate session ID from the victim in order to gain access to an authenticated session which allows the attacker to avail web resources. For example, attacker, with an application layer hijacking can access the website resources secured for authenticated users only. The web server may assume that the incoming request forms the known host whereas an attacker has been hijacked the session by predicting the session ID.

**Compromising Session IDs using Sniffing**

Attacker uses a sniffer to capture a valid session token or ID. He then uses that token to gain unauthorized access to the web server

**Compromising Session IDs by Predicting Session Token**

Predicting the session ID is the process of observing the currently occupied session IDs by the client. By observing the common and variable part of the session key, an attacker can guess the next session key.

**Attack using Session Fixation:**

* Attacker exploits the vulnerability of a server which allow a user to use fixed SID
* Attacker logins using valid session ID and then try to lure user authenticate himself and steals his valid session ID

**Attack using Proxy Servers:**

* By using bogus link, attacker convince user to click on a link which will look like a legit website but is attacker’s website.
* Attacker then send that traffic to his system using proxy and then he captures the session ID

**Attack using CRIME:**

* Compression ratio Info-Leak Made easy is a client side attack which exploits the vulnerabilities present in data compression features of protocols like TLS and HTTPS

**Attack using Forbidden Attack:**

* It is a type of man in the middle attack that hijacks HTTPS sessions
* It exploits reusing of cryptographic nonce during TLS handshake