Parte “A” – BackEnd

Session Hijacking

Il **Session Hijacking** (in italiano **Dirottamento di Sessione**) è un attacco che permette ad un aggressore di accedere alle aree riservate delle applicazioni Web senza neppure effettuare il login, sfruttando la sessione validata per un altro utente.

Sessioni stateful

Una volta effettuata il login, le applicazioni Web che utilizzano la sessione stateful rilasciano un cookie di sessione. Ciò significa che si affidano a questo cookie per tenere traccia del client. All'interno del cookie è salvato un codice univoco che consente il riconoscimento del client.

Ora capiamo bene che per il server, chiunque sia in possesso di questo codice univoco sarebbe il client autenticato. Se un attaccante riuscisse a entrare in possesso di questo identificativo, potrebbe sfruttare la sessione validata originariamente per la sua vittima.

Mitigazioni contro il Session Hijacking

L'utente può fare ben poco contro un attacco **Session Hijacking**. Al contrario però, l'applicazione può accorgersi

che un device diverso si è collegato con uno stesso identificativo di sessione.

E facendo affidamento a ciò, possiamo progettare delle mitigazioni:

- Fingerprint del client: associare ad ogni sessione alcune caratteristiche tecniche del device collegato, così da rilevare un cambiamento dei parametri registrati. Queste informazioni vanno salvate nel cookie (per le sessioni stateful) o nel JWT (per le sessioni stateless) assolutamente crittografate.
- Sistemi in grado di rilevare accessi da device o paesi inusuali
- Se la sessione fosse basata su cookie, rilasciare il cookie con l'attributo *HTTPOnly* così da renderlo inarrivabile in caso di attacco XSS
- Per le sessioni stateless con il token salvato in *sessionStorage* non esiste una protezione simile ad *HTTPOnly*, quindi è necessario fare la massima attenzione contro l'attacco XSS