11.a Milyen forgalom védelmét látja el az SSL (Secure Socket Layer) protokoll és miben befolyásolhatja ez a tervezést? Mutassa be az SSL protokoll felépítését és működését!

Az SSL protokoll célja és feladata, szerkezeti felépítése, alprotokolljai és feladatuk

SSL célja

- Titkosított kommunikációt biztosító protokoll, ami nyílt hálózatokban, kapcsolatorientált kommunikációban nyújt védelmet.
- Csak egy-egy kommunikációs csatornát biztosít.
- Gyakran használják a weboldalak biztonságos titkosítására is.

SSL szerkezeti felépítése

- Minden egyes kapcsolat egyedi kulccsal titkosít.
- Tanúsítvány igazolja a szervert.
- Biztosítja az adatintegritást. (MD5, SHA-1)

SSL működése

1. Kliens csatlakozik a kiszolgálóhoz.
2. Kiszolgáló elküldi a hitelesítési tanúsítványt a kliensnek.
3. Kliens ellenőrzi a tanúsítvány hitelességét, majd létrehozza a titkosított kapcsolatot a kiszolgálóval.
4. Kliens és kiszolgáló között így már biztonságosan lehet adatokat cserélni.
5. Ha az SSL kapcsolat megszakad, akkor a kliens és a kiszolgáló kapcsolata is megszakad.

SSL alprotokolljai

Rekord protokoll

- Feladata a kliens és a szerver és a felsőbb SSL protokoll entitások védelme:
    - Titkosítás, integritásvédelem, üzenet-visszajátszás elleni védelem

Handshake protokoll

- Rekord protokollban használt kriptográfiai algoritmusok és paramétereik egyeztetése.
- Kulcscsere és hitelesítés

Change-Cipher-Spec protokoll

- Egyetlen üzenetből áll, ami a Handshake protokoll kulcscsere részének végét jelzi.
- Ezt az üzenetet elküldi, utána az adott fél az új algoritmusokat és kulcsokat kezdi használni a küldése.
    - A vétel még mindig a Handshake előtti állapot szerint történik.

Alert protokoll

- Figyelmeztető és hibaüzenetek továbbítása.

A handshake, valamint a record alprotokoll feladata, működése és üzenetei

Rekord protokoll működése

- A felsőbb protokoll rétegektől érkező üzeneteket:
  - Fragmentálja, ha szükséges.
  - Fragmenseket tömöríti
  - Tömörített fragmenseket fejléccel látja el
  - Fejléccel ellátott, tömörített fragmensre üzenethitelesítő kódot/MAC-et számol és azt a fragmenshez csatolja.
  - Az üzenethitelesítő kóddal ellátott fragmenst rejtjelezi.

Rekord üzenetei

- **type**: Rekord üzenetben melyik felsőbb protokoll található.
- **version:** SSL verzió
- **length:** Fragmens hosszát tartalmazza bájtban mérve.
- **MAC:** Üzenethitelesítő kód generálása

Handshake protokoll működése

1. **fázis:** Kliens és szerver elküldi a tulajdonságait, megállapodnak
2. **fázis:**
   a. Kulcscseremódszertől függ
   b. Szerver elküldi a tanúsítványát és kéri a kliens tanúsítványát.
3. **fázis:** Tanúsítvány ellenőrzés és kulcscsere folytatása
4. **fázis:** Kulcscsere életbelépése, befejezése

Handshake üzenetei

- **KliensHello:**
  - Kliens küldi ezt az üzenetet az SSL Handshake kezdeményezésére.
  - Kliens verzió, véletlenszám, viszonyazonosító, biztonsági algoritmusok, tömörítő algoritmusok
- **SzerverHello:**
  - Kiszolgáló küldi a **KliensHello** üzenetre válaszul.
  - Szerver verzió, véletlenszám, viszonyazonosító, biztonsági algoritmusok, tömörítő algoritmusok
- **Szerver kulcscsere üzenet**
- **Tanúsítvány kérés**
  - Előfordulhat olyan eset is, amikor a tanúsító hatóságok listája üres.
    - Ilyenkor a kliens eldöntheti, hogy elküldi-e az ügyféltanúsítványt vagy sem.
- **Kliens tanúsítvány**
  - A kliens bemutatja a tanúsítványláncát a kiszolgálónak.
- **Kliens kulcscsere üzenet**
  - Lényege, hogy létrehozza a közös kulcsot a kliens és a kiszolgáló között anélkül, hogy azt egy kívülálló számára felfedné.
- **Kész üzenet**
  - Első olyan üzenet, ami már az új algoritmusokat használva, az új kulcsokkal van kódolva.

Az SSL és TLS protokoll értékelése

- SSL a TLS elődje, de már nem biztonságos.
- SSL utolsó verziója 3.0-a, amit 1996-ban adtak ki.
- TLS sokkal biztonságosabb, aminek jelenlegi verziója 1.3, amit 2018-ban adtak ki.
    - Például továbbított titoktartás támogatása és biztonságosabb rejtjelkészletek
- Különböző port számokat használnak, az SSL 443, a TLS 587-es portot.