# **12.a** Ismertesse a hálózati kommunikáció védelmére alkalmazott kriptográfiai algoritmusokat! Magyarázza el működésüket!

# Kriptográfia

* A kriptográfia lényege, hogy az adatokat biztonságban tárolhassuk az illetéktelen hozzáférések ellen és adatküldésnél a CIA elvek alapján biztonságban áramoljon az információ.
* **Elvárások:** Gyors encryptelés és a megfelelő decrypt kulcs esetén visszafejthetőség vagy egyirányú legyen.
* **Kriptoanalízis:** A titkosítás megfejtésének tudománya.
* **Kriptológia:** Kriptográfia és kriptoanalízis együtt.

## Rejtjel (cipher)

* Karakterről karakterre átalakítás
* Bitről bitre átalakítás

## Kód (code)

* Egy szó helyettesítése egy másik szóval vagy szimbólummal.

A képen szöveg, képernyőkép, Betűtípus, diagram látható

Automatikusan generált leírás

## Támadási lehetőségek

|  |  |
| --- | --- |
| Passzív támadás | Aktív támadás |
| Üzenet lehallgatása | Üzenet megváltoztatása |
|  |  |

## Szimmetrikus titkosítás

* **A képen szöveg, képernyőkép, diagram, Betűtípus látható

  Automatikusan generált leírás**A titkosításhoz és a visszafejtéshez ugyanazt a kulcsot használják.
* Gyorsabb, mint az aszimmetrikus kriptográfia.
* **AES – Advanced Encryption Standard:** 
  + Alacsony memóriaigény, gyors, leváltotta a **DES**-t.
* **DES – Data Encryption Standard**
  + Blokkrejtjelező,
  + Eredetileg 56 bites kulcs hossz.
  + 64 bites input blokkokat fogad és 64 bites rejtjelezett szöveged eredményez.
* **3DES – Tripla DES:** 
  + Kettő vagy három titkosító kulcsot használ

## Asszimmetrikus titkosítás

* A képen szöveg, képernyőkép, diagram, Betűtípus látható

  Automatikusan generált leírásA titkosításhoz és a visszafejtéshez különböző kulcsokat használnak.
* Lehetővé teszi a hitelesítést és az adatok védelmét közvetlen kulcsmegosztás nélkül.
* **RSA - Manapság leggyakrabban használt**
  + Titkosításhoz egy nyílt és egy titkos kulcs tartozik.
    - **Például** egy postaláda, ahova a postás be tudja dobni a levelet, de ki nem tudja szedni, csak mi tudjuk a postaláda kulcsával.
  + Nyílt kulcs bárki számára elérhető, és ezzel lehet kódolni a másoknak szánt üzenetet.
  + Titkos kulccsal lehet megfejteni a nyílt kulccsal kódolt üzenetet.
* **DSA**
  + Privát kulcsot használjuk az üzenetek digitális aláírásának létrehozásához.
  + Nyilvános kulcsot használjuk az aláírás ellenőrzéséhez.
* **Diffie-Hellman kulcscsere**
  + Biztonságos kommunikációs csatornát hoz létre, de úgy, hogy közbe nem kell a titkos kulcsot közvetlenül átadniuk.
  + A képen diagram, képernyőkép, sor, Tervrajz látható

    Automatikusan generált leírásKét fél a privát kulcsaikat használja a titkos kulcs létrehozásához, amit csak egymás között használhatnak.

## Hash-függvények

* Bemenő adatokból rövid, állandó hosszúságú hash-t állítanak elő.
* A hash függvényeket a hitelesítéshez és az adatok integritásának ellenőrzéséhez használják.

## Titkosító protokollok

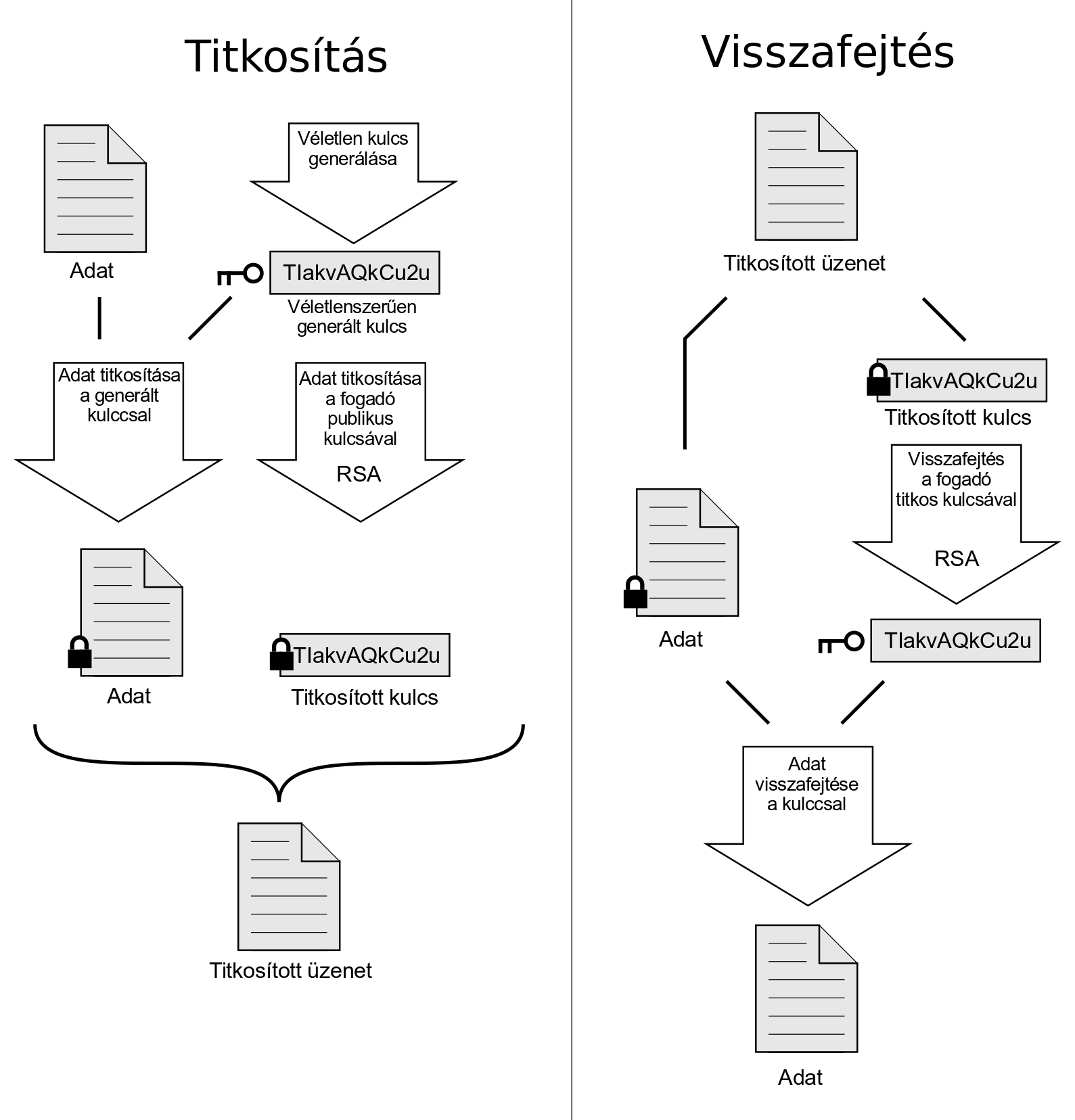
* Adatkapcsolati rétegbeli titkosítás
* Hálózati rétegbeli titkosítás (IPSec)
* Szállítási rétegbeli titkosítás (SSL, TLS)
* Alkalmazási rétegbeli titkosítás (PGP)

## A képen diagram, képernyőkép, szöveg látható Automatikusan generált leírásDigitális aláírás

* Olyan elektronikus aláírás, amit digitális tanúsítványt hitelesít.
* Az aláírás tartalmaz egy ellenőrző összeget, amihez szükség van egy hashfüggvényre (SHA-1 vagy MD5).
* **Hozzáfűzzük:**
  + Aláíró nevét vagy azonosítóját
  + Aláírás idejét
  + Hashfüggvény nevét
  + Egyéb dolgok, amiket fontosnak tartunk

## PGP – Pretty Good Privacy

* Ötvözi a szimmetrikus kulcsú titkosítás gyorsaságát az aszimmetrikus kulcsú titkosítás biztonságával, ezért hibrid titkosítási módszernek nevezzük.



# 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:** 
   1. Kulcscseremódszertől függ
   2. 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.

# IPSec

## AH – Authentication Header

* Sértetlenséget, hitelesítést és visszajátszás elleni védelmet biztosít.
* Beszúr egy AH fejlécet, ami egy MAC-et tartalmaz.
* A visszajátszás detektálásának érdekében, az IP csomagokat sorszámozza.
* Az AH fejlécben található MAC érték a sorszámot is védi.

## ESP – Encapsulated Security Payload

* Feladata az IP csomag tartalmának rejtése és opcionálisan a tartalom integritásának védelme.
* IP csomag tartalmának rejtését rejtjelezéssel oldja meg.
* **Tartalom integritásának védelme:** ESP fejlécre és a csomag tartalmára számít MAC kódot és azt a csomaghoz csatolja.
* ESP MAC nem védi az IP fejléc mezőit.

## ISAKMP – Internet Security Association and Key Management Protocol

* Általános célú keretprotokoll, ami bármilyen konkrét kulcscsere protokoll üzeneteit képes szállítani.

## IKE – Internet Key Exchange

* IPSec hivatalos kulcscsere protokollja.
* A host-ok ebben a fázisban hitelesítik egymást shared secret vagy RSA kulcs segítségével.
* Felépítenek egy kétirányú ISAKMP SA-t.
* Az ISAKMP SA-t alkalmazva megvitatják az egyirányú IPSec SA-kat.

# Az IPsec protokollok paramétereinek konfigurálási megfontolásai és lépései

## Megfontolások

* **Titkosítási módszer:** DES, 3DES, AES, stb
* **Autentikációs módszer:** Például SHA, MD5, stb
* **Kulcsrotációs periódus:** Mennyi ideig használhatjuk ugyanazt a titkosítási és autentikációs kulcsot.
* **Pre-shared key:** Összes hálózati eszköz ismeri a kulcsot.
* **Perfect Forward Secrecy**: A régi kulcsok már nem használhatóak.

# IPsec üzemmódok jellemzői, működése, konfigurálása, tesztelése

## Üzemmódok

* **Szállítási (transport) mód**
  + Az AH vagy az ESP fejléc a csomag eredeti IP fejléce és a felsőbb szintű protokoll fejléce közé kerül.
* **Alagút (tunnel) mód**
  + Az eredeti IP csomagot teljesen beágyazzuk egy másik IP csomagba.
  + Az AH vagy az ESP fejléc az új és az eredeti IP fejléc közé kerül.
  + Az AH fejléc vagy az ESP trailer következő fejléc mezője IP-re utal.

## IPSec működése

* Adatgyűjtés
* Titkosítás
* Autentikáció
* Csomagolás
* Továbbítás
* Titkosítás feloldása
* Adatok fogadása