Óbudai Egyetem

Bánki Donát Gépész és Biztonságtechnikai Mérnöki Kar

Műszaki területek informatikai biztonsága
SSL

ZENTAI DÁNIEL

ZENTAI.DANIEL@BGK.UNI-OBUDA.HU

SSL (Secure Socket Layer) alprotokollok

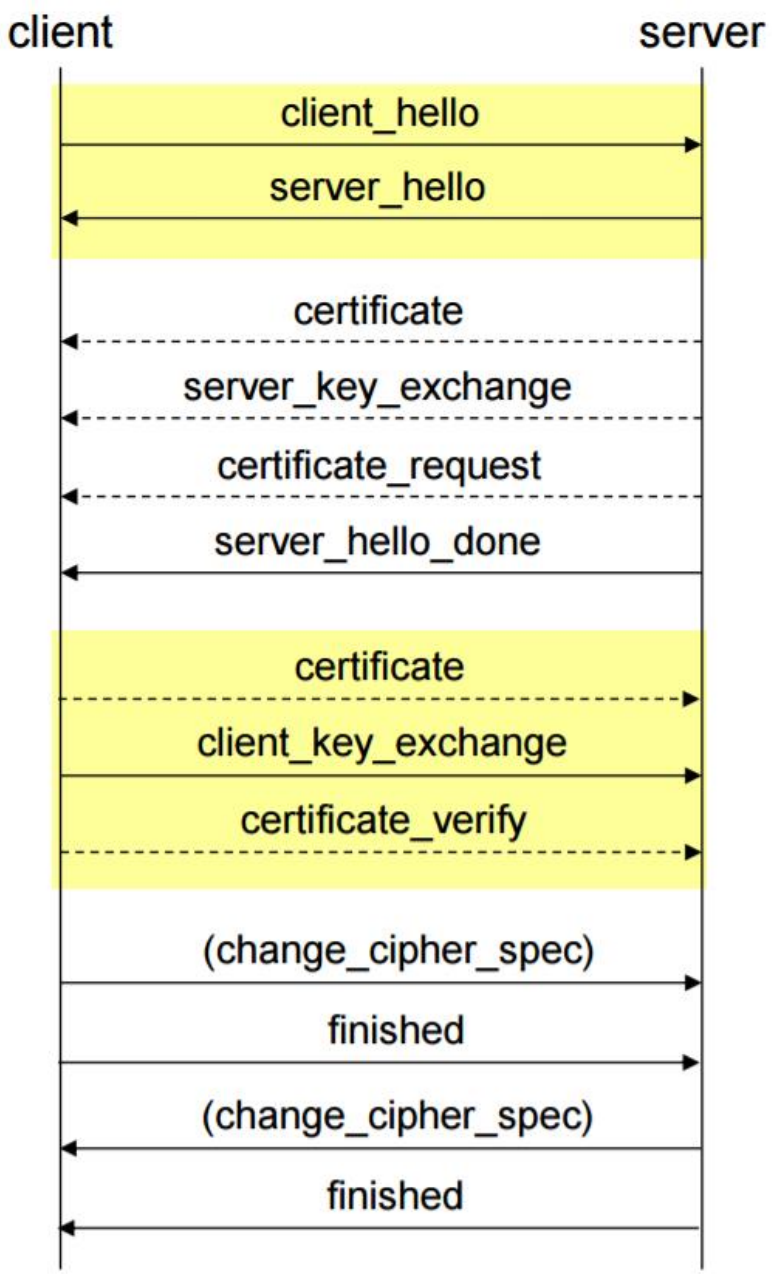
SSL Handshake Protocol:
- Megegyezés az algoritmusokban, paraméterekben
- Kulcscsere
- Autentikáció

SSL Record Protocol:
- Fragmentálás
- Tömörítés
- MAC
- Titkosítás

SSL Alert Protocol:
- Hibaüzenetek

SSL Change Cipher Spec Protocol:
- Egy darab üzenet, a Handshake kulcscseréjének a végét jelzi

SSL Handshake



SSL Handshake

A Hello üzenetek tartalma:

- version: A legfrissebb támogatott verzió
- random: aktuális idő + 28 byte-os random szám
- session_id: Az aktuális session azonosító, új session esetén a kliensnél ez üres
- cipher_suites: A kliensnél a támogatott algoritmusok listája, a szerver ezek közül választ; Pl: SSL-RSA-with-3DES-EDE-CBC-SHA1

SSL Handshake

A támogatott kulcscsere protokollok:

RSA alapú

Fix Diffie-Hellman

Egyszer használatos Diffie-Hellman

Anonim Diffie-Hellman

SSL Handshake

Certificate:

Minden kulcscsere metódushoz tartozik tanusítvány, az anonim DH-t kivéve

Server_key_exchange:

Csak akkor van rá szükség, ha a tanusítvány nem tartalmaz elég információt a hitelesítéshez

Certificate_request:

Akkor küldi a szerver, ha a kliensnek autentikálnia kell magát

Server_hello_done:

A hello üzenetek végét jelöli, ezután a kliens még visszajelez

SSL Handshake

Certificate:

Minden kulcscsere metódushoz tartozik tanusítvány, az anonim DH-t kivéve

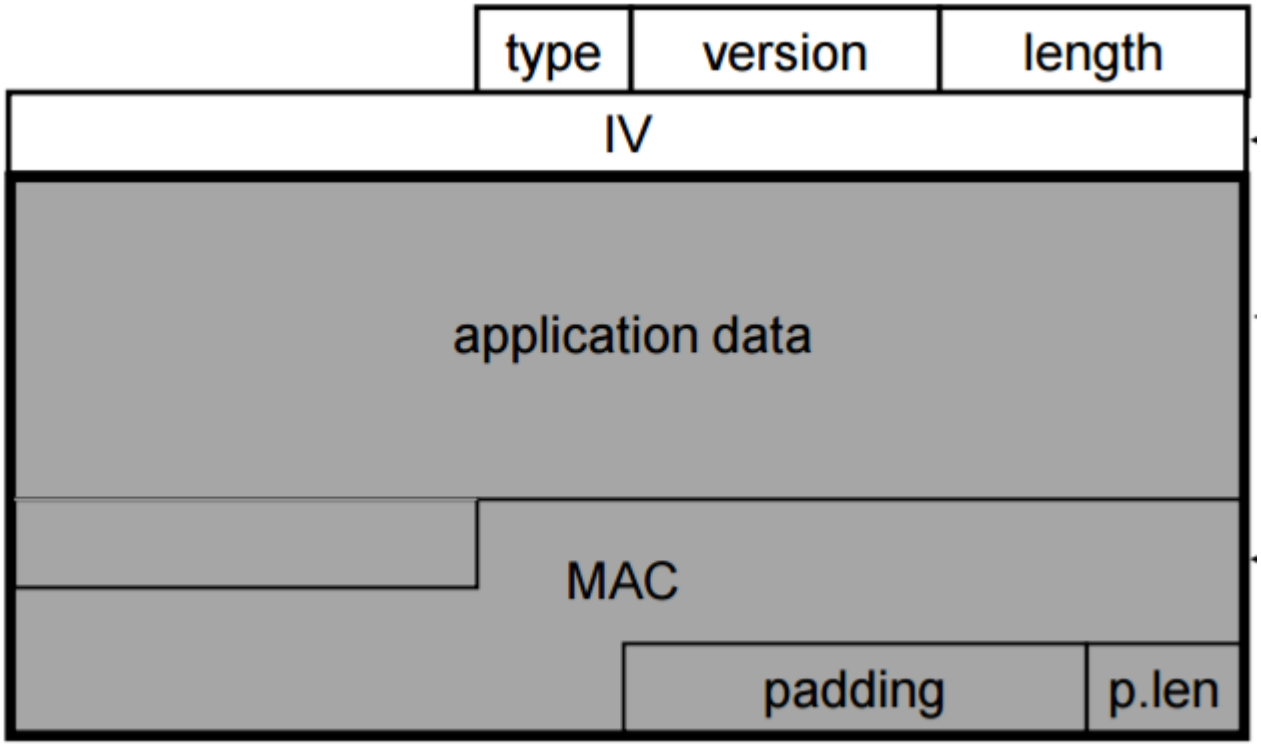
Client_key_exchange:

Erre mindig szükség van, a kliens hozzájárulása a kulcscseréhez

Certificate_verify:

A szerver visszajelez, hogy elfogadta a kliens tanusítványát

SSL Record

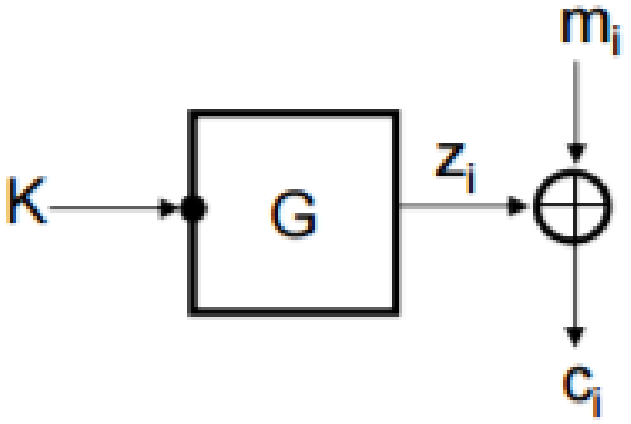


SSL Record

Támogatott titkosító algoritmusok:

3DES-EDE, AES128, AES256, RC4

RC4: 128 bites kulcsfolyam titkosító



RC4

- initialization:

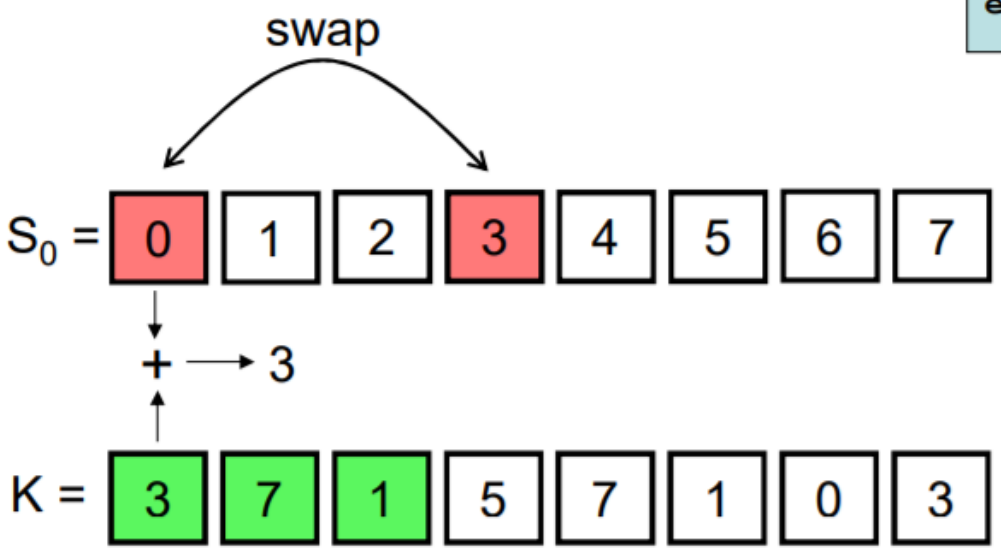
```
for i = 0 to 255 do
  S[i] = i
end

j = 0
for i = 0 to 255 do
  j = j+S[i]+K[i mod len(K)] mod 256
  swap(S, i, j)
end

i = 0
j = 0
```

- generation:

```
i = i+1 mod 256
j = j+S[i] mod 256
swap(S, i, j)
return S[ S[i]+S[j] mod 256 ]
```

RC4

```
for i = 0 to 7 do
    j = j + S[i] + K[i] mod 8
    swap (S, i, j)
end
```

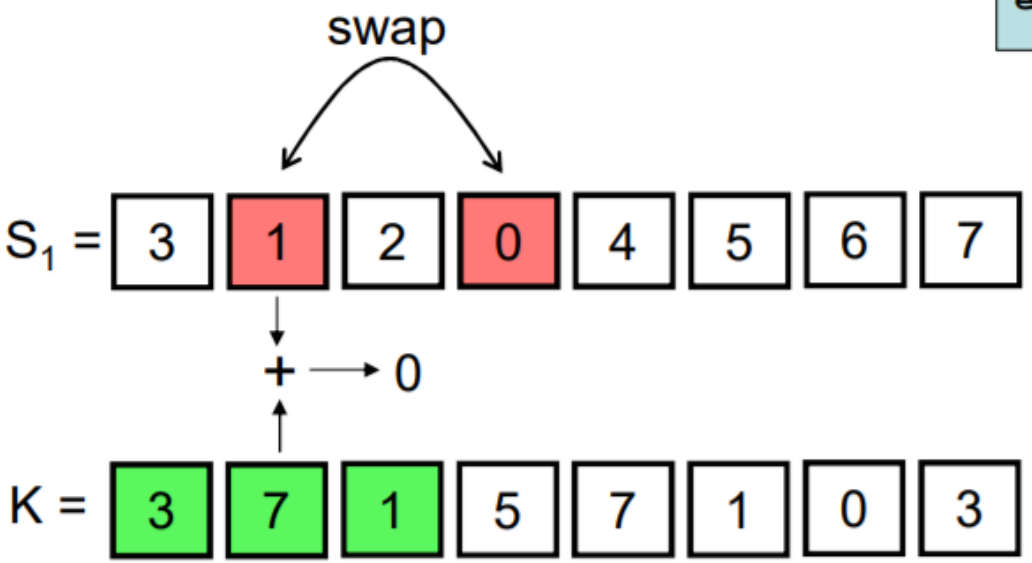


j = 0

i = 0 $j_1 = 3$

RC4

```
for i = 0 to 7 do
    j = j + S[i] + K[i] mod 8
    swap (S, i, j)
end
```



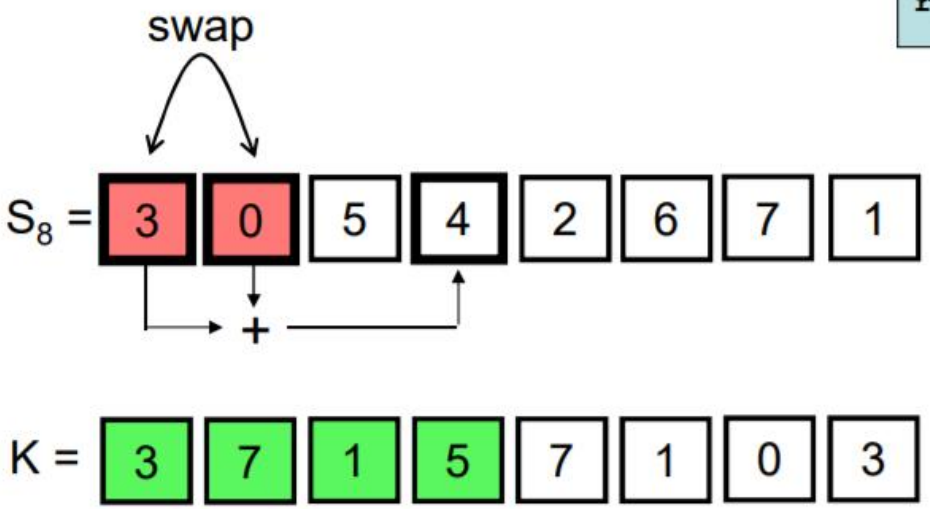
j = 0

i = 0 $j_1 = 3$

i = 1 $j_2 = 3$

RC4

```
i = i+1 mod 8
j = j + S[i] mod 8
swap (S, i, j)
return S[ S[i]+S[j] mod 8 ]
```



i = 0 j = 0

i = 1 j = 0

output X = 4

$K[3] = S_3^{-1}[X] - j_3 - S_3[3] = 4 - 6 - 1 = 5$

SSL Record

MAC:

A MAC inputja: MAC_write_key, seq_num | type | version | length | payload

A támogatott MAC algoritmusok: HMAC a következő hash függvényekkel: MD5, SHA1, SHA256

HMAC emlékeztető

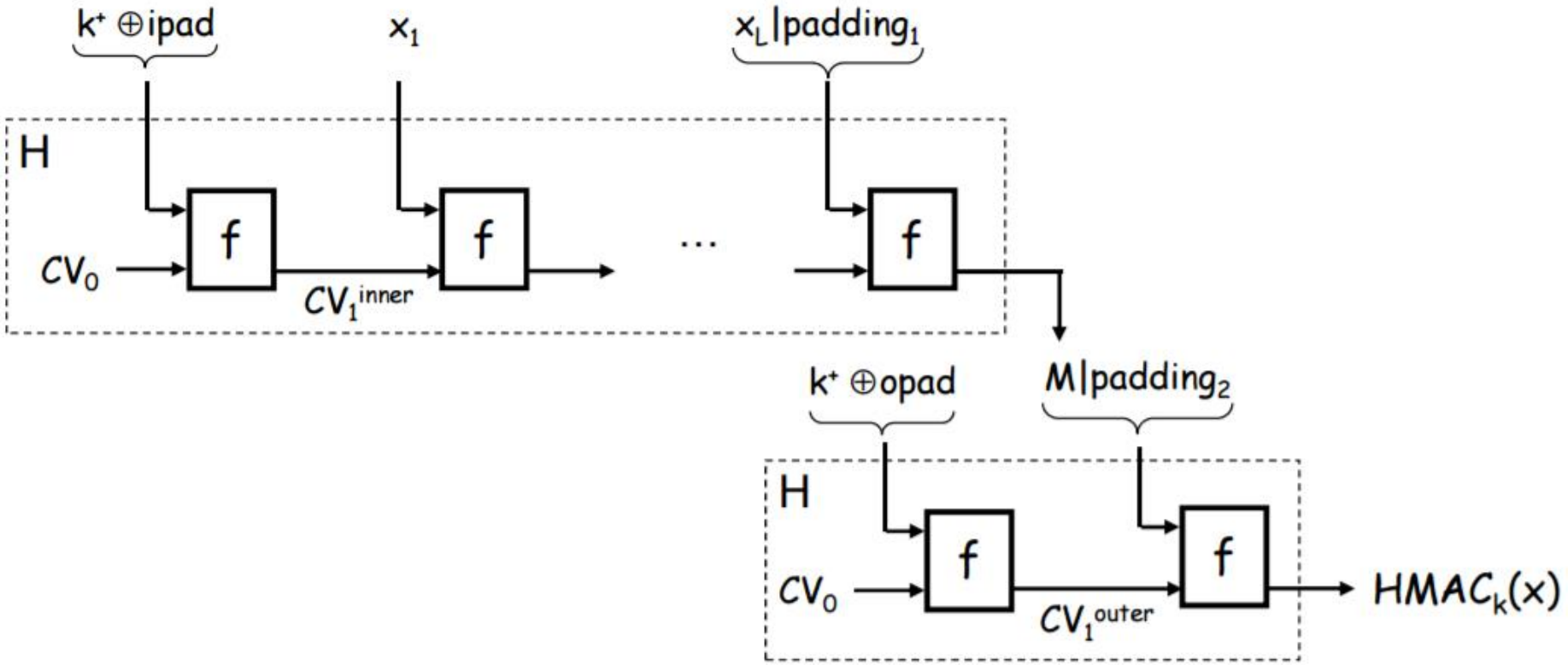
$HMAC_K(x) = H((K^+ \oplus opad) \mid H(K^+ \oplus ipad \mid x))$

K^+=K kiegészítve 0-kal 1 blokk méretűre

ipad=00110110 ismételve 1 blokk hosszan

opad=01011100 ismételve 1 blokk hosszan

HMAC emlékeztető



Köszönöm a figyelmet!