Practice09

Friday, May 5, 2023 2:00 PM

	Hashing	MAC	Digital Sign.
Integrity	✓	✓	✓
Authenticity	×	✓	✓
Non-repudiation	×	×	✓

HMAC

(Hash-based Message Auth. code)

$$HMAC(H, K, m) =$$

H: hash func, K: key, m: message

$$H\left(\left(K' \oplus opad\right) \| H\left(\left(K' \oplus ipad\right) \| m\right)\right)$$

$$K' = \begin{cases} H(K) & \text{if } len(K) > H.digest\_size \\ K & = \\ K \| 0 \ldots 0 & < \end{cases}$$

$$len(K') = H.digest\_size$$

$$opad = 0x5c\ 0x5c \ldots$$

$$len(opad) = H.digest\_size$$

$$ipad = 0x36, \ldots$$

ipad = 0x36 ....

len(ipad) = len(K')

A and B agrees on H and K

A $\xrightarrow{m \| HMAC(H,K,m)}$ B

m || h

Check HMAC(H,K,m) = h

Key exchange

block stream

HMAC, AES, RC4 (work with same secret key on both sides, symmetric)

1° We can use assymetric encryption for key exchange since we don't need to share secret key for that.

2° We can use the shared secret as

Seed for PRNG

PRF in TLS 1.2 (RFC 5246)

↑ Transaction layer Security

PRF(H, secret, label, seed, N) =

seed = label || seed

P(H, secret, seed, N) =

HMAC(H, secret, A(1) || seed) ||
HMAC(H, secret, A(2) || seed) || ...

A(0) = seed

A(i) = HMAC(H, Secret, A(i-1))

Padding of messages

1° (OAEP)

00 ... 01 || m

2 bytes block size for MiniAES

if len(m) is odd

001 || m

2° Encode the lenght

len(m) || m || 0 ... 0

fixed number of bytes

~~Pad with know bytes~~