Message Authentication Codes (MACs) a.k.a. keyed Hash Functions

→ Can be used as an (efficient) replacement for signing if

* Alice and Bob have the same shared secret (followed by e.g. DH key exchange).
* Alice and Bob don't care about non-repudiation:

Bob cannot convince a third party that a message is from Alice, since he would have made the same signature.

extra sec. property → For Eve not knowing the key any output should be equally likely.

Weak constructions:

→ $h_k(m) = h(m \| k)$ (Suffix MACs)
if h has MD (unstrengthened) design then

$h(m) = h(m') \Rightarrow h_k(m) = h_k(m')$
Hence false signatures can be generated without knowing the key and security reduces to plain hashing.
If length(m) and length(m') are equal also the strengthened MD construction fails.

→ $h_k(m) = h(k \| m)$ (Prefix MACs)
again if h has ~~strengthened~~ (unstrengthened) MD design then.
if $m = m_1, \dots, m_n$
$m' = m_1, \dots, m_{n+1}$

can be computed by anyone.

$h_k(m') = h(k \| m_1 \| \dots \| m_{n+1}) = h(h(k \| m_1 \| \dots \| m_n) \| m_{n+1}) = h(h_k(m) \| m_{n+1})$

Hence without knowledge of k a valid signature can be computed on a different message.

Does not work in the strengthened version.

fixed values

Typical constructions – HMAC : $h_k(m) = h((k \oplus opad) \| h[(k \oplus ipad) \| m])$

~~$Enc_{k_1}(m) \| Enc_{k_2}(h_{k_3}(m))$~~

E~~xtra Security : $Enc_{k_1}(m) \| Enc_{k_2}(h_{k_3}(m))$~~

~~instead of $m \| h_k(m)$~~

Alice
(k)

Bob
(k)
m
$\sigma = h_k(m)$

m, σ

accept m as message from Bob if $h_k(m) = \sigma$.

In practice 3 pieces of shared secret used to improve security:
$E_{k_1}(m) \| E_{k_2}(h_{k_3}(m))$ is sent instead.