Message Authentication Codes (MACs)

Recap: PRF $F: \mathcal{K} \times X \to Y$ is indist. from random func. in $Funs[X,Y]$

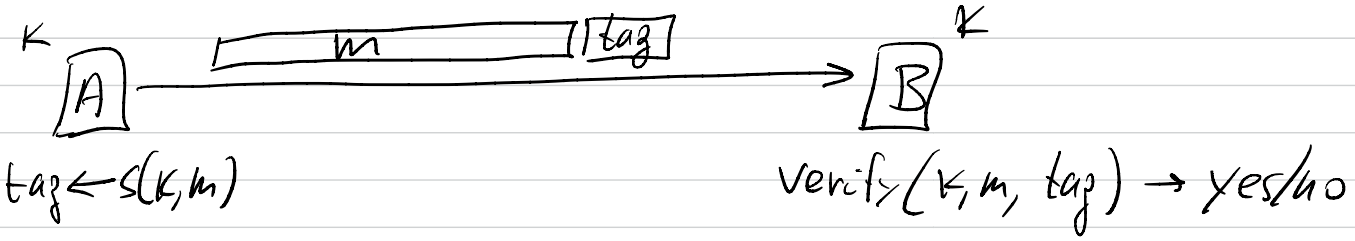
PRP $E: \mathcal{K} \times X \to X$ —"— " perm. in $Perms[X]$

and is eff. invertible

A PRP over $(\mathcal{K}, X)$ is a PRF over $(\mathcal{K}, X, X)$

one-time key: ctr-mode, many-time key (CPA): rand-ctr-mode.

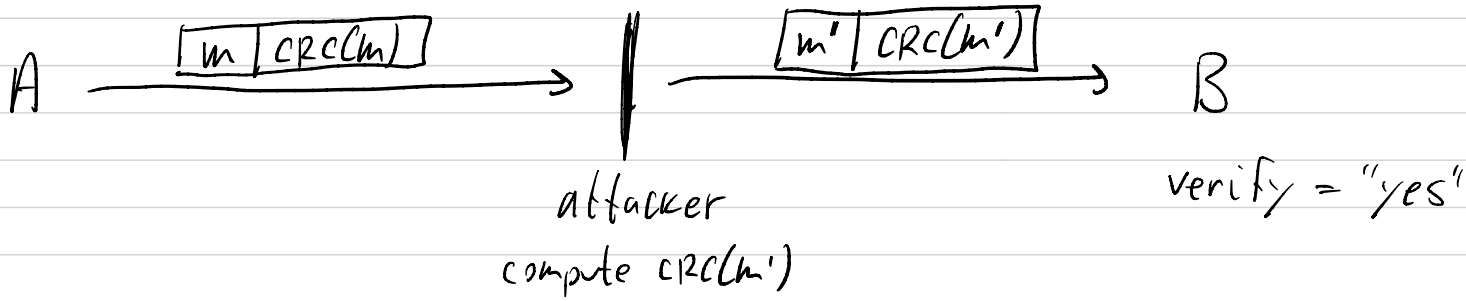
Message integrity:



- used for data integrity, not privacy
- requires a shared secret key:

  checksum (CRC) protects from <u>random</u> errors

  not malicious errors.



Def: A MAC def. over $(\mathcal{K}, \mathcal{M}, \mathcal{T})$ is a pair of "eff" alg. (S, V) where

- $S(k, m) \rightarrow t \in \mathcal{T}$
- $V(k, m, t) \rightarrow$ yes/no

s.t. $\forall k \in \mathcal{K}, m \in \mathcal{M}$: $V(k, m, S(k, m)) = \text{yes}$

What is a secure MAC?

Attacker's power: chosen msg attack

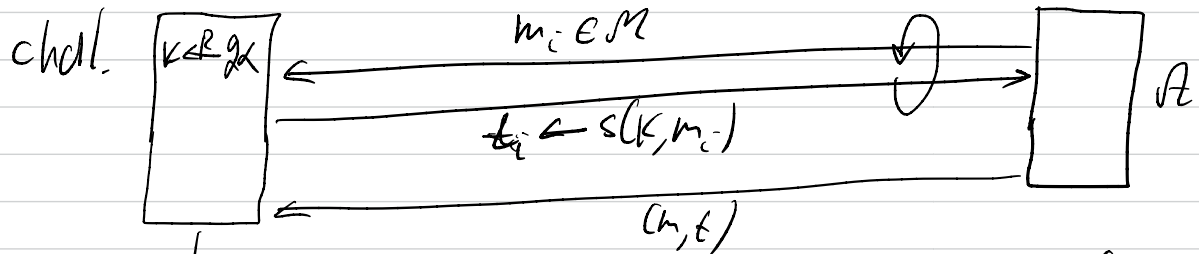
For $m_1, \ldots, m_q \in \mathcal{M}$ attacker gets $t_i \leftarrow S(k, m_i)$

Attacker's goal: existential forgery

produce valid (m, t) where $(m, t) \notin \{(m_1, t_1), \ldots, (m_q, t_q)\}$

$\Rightarrow$ attacker cannot produce a valid tag for any msg.

Formally: For a MAC (S, V) and adv. A def. MAC game:



$\downarrow$ {win: if $V(k, m, t) = yes$ and $(m, t) \notin \{(m_i, t_i)\}$
{lose: otherwise

Def: $I = (S, V)$ is secure MAC if for all "eff" A

$$\text{MAC adv}[A, I] := \Pr[\text{chal. outputs "win"}]$$

is negligible.

Note: For a secure MAC, given valid (m, t), adv. can't build (m, t') where $t \neq t'$.

Construction: every secure PRF (with a sufficiently large range) gives a secure MAC.

Let F be a PRF over $(\mathcal{K}, X, Y)$

Def: $I_F = (S, V)$ as: $S(k, m) := F(k, m)$

$$V(k, m, t) := \begin{cases} \text{yes} & \text{if } t = F(k,m) \\ \text{no} & \text{otherwise} \end{cases}$$

Thm: F a secure PRF over $(\mathcal{K}, X, Y)$ where $1/|Y|$ is negligible (e.g. $\leq 1/2^{128}$)

Then I_F is a secure MAC

In particular, For every MAC adv. A there is a PRF adv. B (where $time(B) \approx time(A)$) s.t.

$$\text{MAC adv}[A, I_F] \leq \text{PRF adv}[B, F] + \frac{1}{|Y|}$$

$\Rightarrow I_F$ is secure if $|Y|$ is large, e.g. $\geq 2^{128}$.

Proof sketch:

step 1: replace $F(k, \cdot)$ by a random function $f \xleftarrow{R} \text{Funs}[X, Y]$

adv. A can't tell the difference.

step 2: A queries $f()$ at $m_1, \ldots, m_q \in \mathcal{M}$.

must guess $f(m)$ For $m \notin \{m_1, \ldots, m_q\}$.

But $f(m)$ is uniform in Y, so A succeeds with prob. $\frac{1}{|Y|}$.

■

So: AES gives a secure MAC for 16-byte msgs.

Main question: Small MAC $\Rightarrow$ Big MAC

(actually Small PRF $\Rightarrow$ Big PRF)

Two constructions:

(1) CBC-MAC: used in banking, CMAC std.

(2) HMAC: used in Internet protocols

(3) PMAC: parallel MAC (not used)

Remark: truncating MACs

Suppose MAC I_F is built from a PRF and outputs n-bit tags ($Y = \{0,1\}^n$)

OK to truncate MAC to $w < n$ bits as long as $1/2^w$ is considered negligible

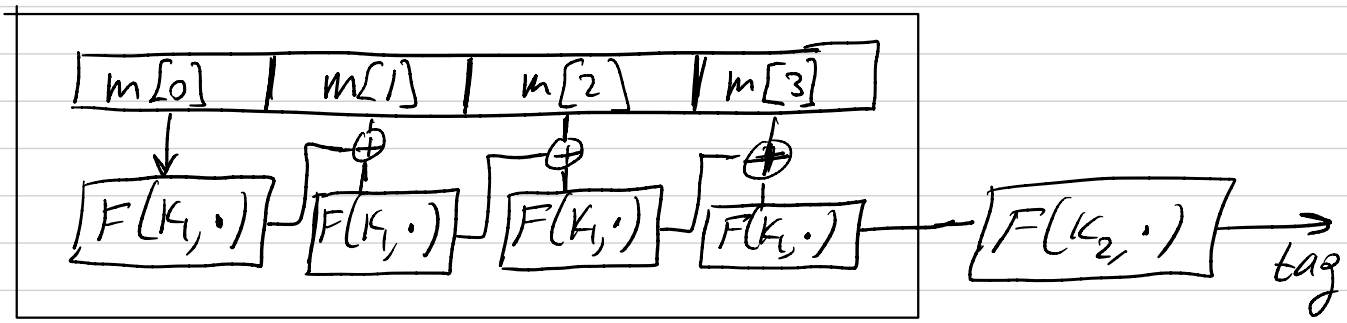
(truncating a secure PRF is a secure PRF)

Ex: Galileo GPS uses a 32-bit MAC (or less)

Construction 1: encrypted CBC-MAC

Let F be a PRF over $(\mathcal{K}, X, X)$ where $X = \{0,1\}^n$ (e.g. AES)

Define a new PRF F_{CBC}: (and therefore also a MAC)



Raw CBC

F_{CBC} is a PRF over $(\mathcal{K}^2, X^{\leq L}, X)$

secret key $= (k_1, k_2)$

$X^{\leq L}$ ← msgs of at most L blocks.

CBC-MAC Thm: For bounded $L > 0$:

F secure PRF $\Rightarrow$ F_{CBC} secure PRF

In particular, for every q-query adv. A attacking F_{CBC} there is an adv. B (where time$(B) \approx$ time(A)) s.t.

$$\text{PRF adv}[A, F_{CBC}] \leq \text{PRF adv}[B, F] + \frac{q^2 \cdot L^{O(1)}}{|X|}$$

$\Rightarrow$ CBC-MAC is secure as long as $q \ll \sqrt{|X|}$

Why the last step? Answer: Raw CBC is insecure!

Adv: (1) choose arbitrary $m \in X$

(2) request tag for m. Get $t = \text{RawCBC}(K, m) = F(K, m)$

(3) output (msg, tag) forgery where $msg = (m, t \oplus m) \in X^2$, $tag = t$

Then: $\text{RawCBC}(K, (m, t \oplus m)) = F(K, \underbrace{F(K,m)}_{t} \oplus (t \oplus m)) = F(K, m) = t$

Note: RawCBC is a secure PRF for fixed size messages.

(or for prefix free messages, e.g. by prepending message length)

- CBC MAC padding: what if msg len is not multiple of block size?

  - Can't simply pad with 0's:

    attack: ask for tag for $\boxed{1 \text{ block msg } m}$

    get tag for $\boxed{m \| 0}$.

    The problem: there are collisions on padding function

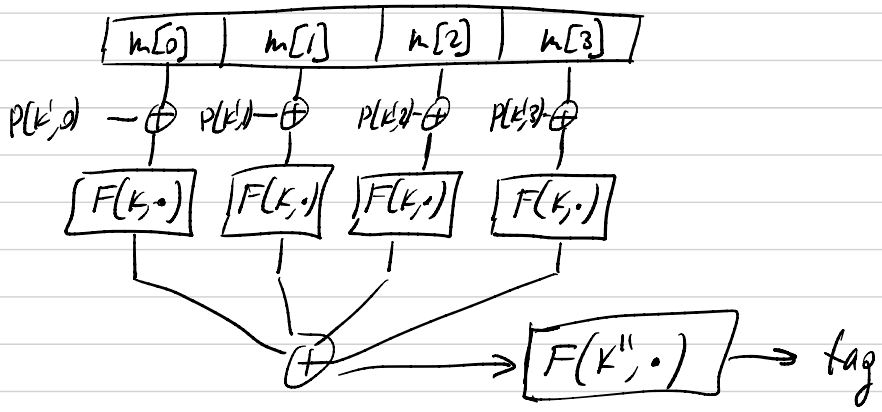
    $$pad(m) = pad(m \| 0)$$

  - Instead: use a one-to-one padding function:

    pad with "100...0" or add dummy block of "100...00".

  - OR: CMAC - never adds new block.

Problem: CBC-MAC is sequential. Better: parallel MAC (PMAC)



$P(K', i)$ - an easy to compute function

without $P()$ MAC is insecure!

Note: F is a PRP $\Rightarrow$ MAC is incremental.

Construction 2: HMAC

First: collision resistant hashing.

Let $H: M \to T$ be a hash function $(|T| \ll |M|)$

a collision for H is a pair $m_0, m_1 \in M$

s.t. $m_0 \neq m_1$ and $H(m_0) = H(m_1)$

Def: A function $H: M \to T$ is collision resistant

if for all (explicit) "eff" algs. A:

$$CRadv[A, H] = Pr[A \text{ outputs collision for } H]$$

is negligible.

Std examples: SHA256, SHA384, SHA512 (2001)

SHA3-256, SHA3-384, SHA3-512 (2014)

How to construct CRHF ?? Next lecture.

An immediate application: Small-mac ⇒ Big-mac

(S, V) a secure MAC over $(\mathcal{K}, \mathcal{M}, T)$ for short msgs.

$H: \mathcal{M}^{big} \to \mathcal{M}$ a CRHF

Define: (S', V') a MAC over $(\mathcal{K}, \mathcal{M}^{big}, T)$ where:

$$S'(k, m) := S(k, H(m))$$
$$V'(k, m, t) := V(k, H(m), t)$$

Thm: (S, V) a secure MAC, H a CRHF.

Then (S', V') is a secure MAC.

"Proof:" Suppose an adv. $\mathcal{A}$ attacks (S', V'):

(1) requests tag on $m_1, m_2, \ldots \in \mathcal{M}$ and get $t_1, t_2, \ldots$.

(2) outputs a Forgery (m, t).

Then either:

(1) $\exists i: H(m) = H(m_i)$ ⇒ Attack on CRHF.

(2) $\forall i: m \neq m_i$ ⇒ Attack on (S, V). ∎

Why CRHF is necessary!

Suppose adv. has $m_0 \neq m_1 \in \mathcal{M}$ s.t. $H(m_0) = H(m_1)$.

attack on (S', V'): request tag on m_0, get t_0.

output Forgery (m_1, t_0).