CIS 5560

Cryptography
Lecture 10

Course website:
pratyushmishra.com/classes/cis-5560-s25/

Slides adapted from Dan Boneh and Vinod Vaikuntanathan

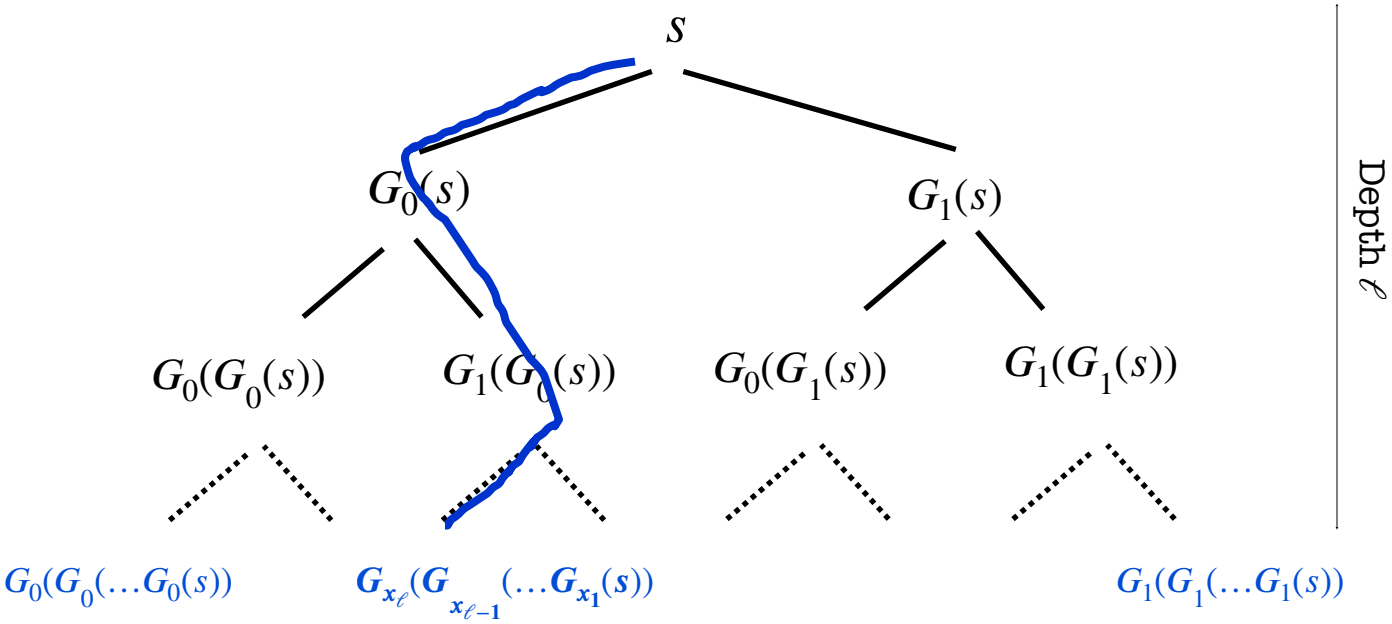
Announcements

- **HW 3 due on Friday 2/21 5PM**
- **HW 4 out on Wednesday 2/19**
  - Due **Friday**, 2/28 at 5PM on Gradescope
  - Covers MACs, and CRHFs

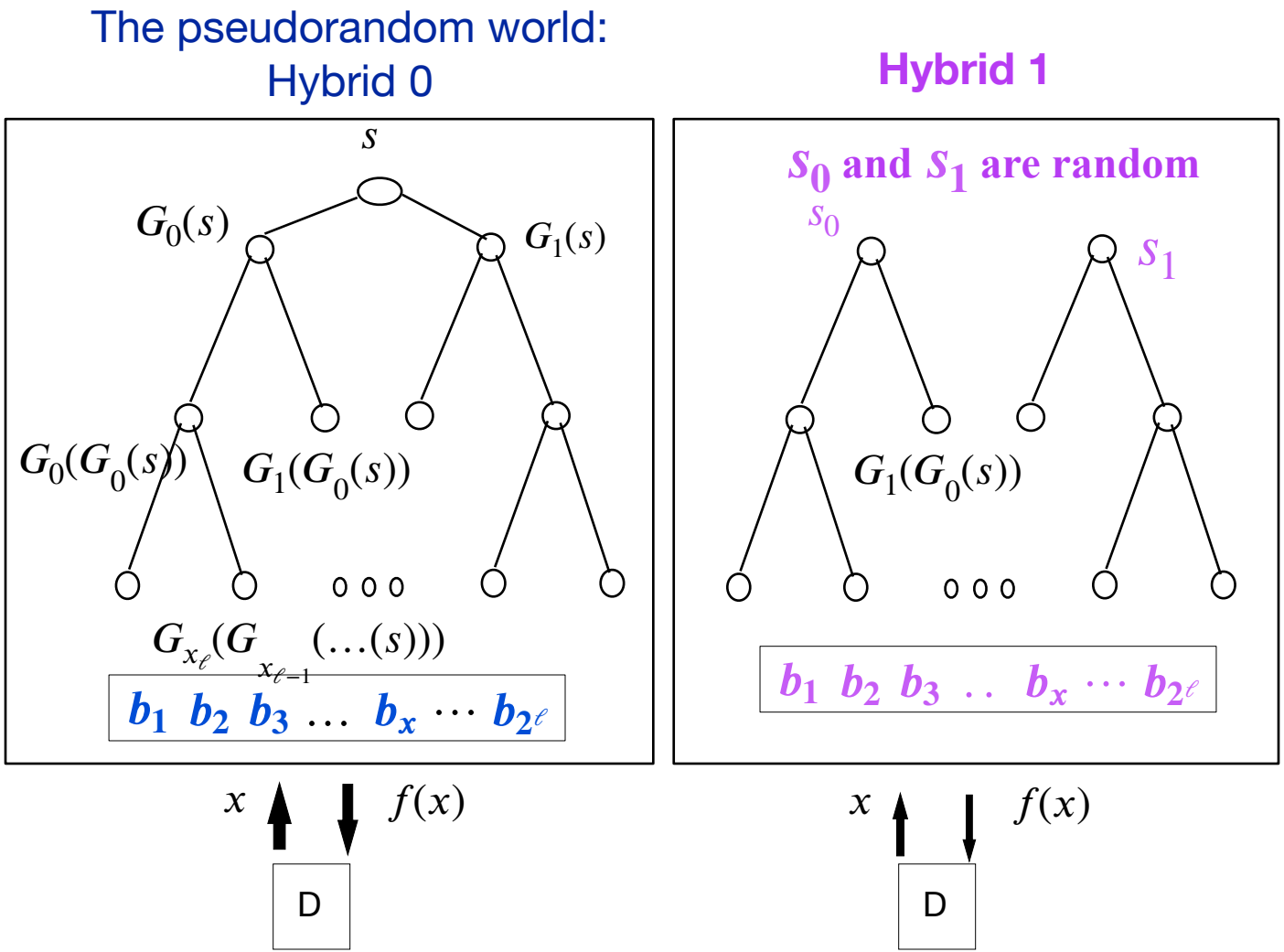
Recap of last lecture

Goldreich-Goldwasser-Micali PRF

Construction: Let G(s) = $G_0(s)||G_1(s)$ where $G_0(s)$ and $G_1(s)$ are both n bits each.



Each path/leaf labeled by $x \in \{0,1\}^\ell$ corresponds to $f_s(x)$.


The pseudorandom world:
Hybrid 0
Hybrid 1
s
$G_0(s)$
$G_1(s)$
$G_0(G_0(s))$
$G_1(G_0(s))$
$G_{x_\ell}(G_{x_{\ell-1}}(\ldots(s)))$
$b_1\ b_2\ b_3\ \ldots\ b_x\ \cdots\ b_{2\ell}$
s_0 and s_1 are random
s_0
s_1
$G_1(G_0(s))$
$b_1\ b_2\ b_3\ \ldots\ b_x\ \cdots\ b_{2\ell}$
x
f(x)
D
x
f(x)
D

Message Authentication Codes (MACs)

A triple of algorithms (Gen, MAC, Ver):

- Gen(1^n): Produces a key $k \leftarrow \mathcal{K}$.
- MAC(k, m): Outputs a tag t (may be deterministic).
- Ver(k, m, t): Outputs Accept or Reject.

Correctness: $\Pr[\text{Ver}(k, m, \text{MAC}(k, m)) = 1] = 1$

Security: *Hard to forge.* Intuitively, it should be hard to come up with a new pair *(m', t')* such that Ver accepts.

EUF-CMA Security

Existentially **U**n**f**orgeable against **C**hosen **M**essage **A**ttacks

$k \leftarrow K$

m_1

$t_1 = \text{MAC}(k, m_1)$

m_2

$t_2 = \text{MAC}(k, m_2)$

$\vdots$

$(m,\ t)$

Accept if $(m,\ t) \neq (m_i,\ t_i)$ for all i, and $\text{Ver}(k, m, t) = 1$

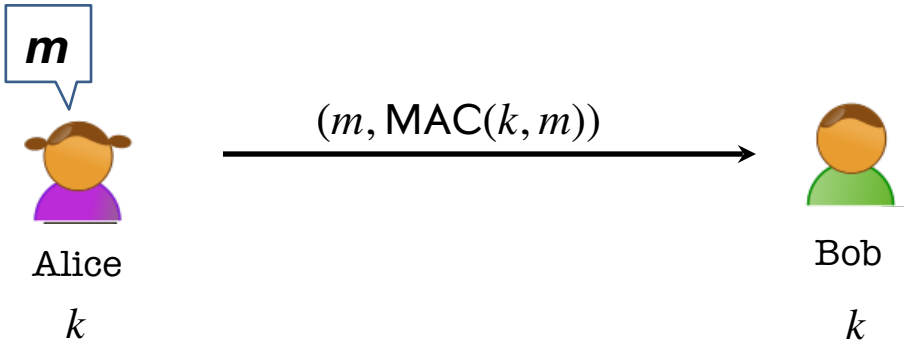
Want: $\Pr((m,\ t) \leftarrow A^{MAC(k,\ \cdot)}(1^n),\ Ver(k,\ m,\ t) = 1,\ (m,\ t) \notin Q)) = negl(n)$.

where Q is the set of queries $\left\{ (m_i,\ t_i) \right\}_i$ that A makes.

Today's Lecture

- Collision-resistant Hash Functions (CRHFs)
- Birthday bound
- CRH → MACs
  - HMAC

Constructing a MAC



Gen(1^n): Produces a PRF key $k \leftarrow K$.
MAC$(k,\ m)$: Output $F_k(m)$.
Ver$(k,\ m,\ t)$: Accept if $F_k(m) = t$, reject otherwise.

Security: ??

A bad example

Suppose **F: K × X ⟶ Y** is a secure PRF with **Y = {0,1}^{10}**

Does plugging F into the previous construction give a secure MAC?

- Yes, the MAC is secure because the PRF is secure
- ⟹ No tags are too short: anyone can guess the tag for any msg
- It depends on the function F
-

$Adv[A, I_F] = 1/1024$

Security

Thm: If **F: K×X⟶Y** is a secure PRF and 1/|Y| is negligible (i.e. |Y| is large) then the previous scheme is a secure MAC.

In particular, for every eff. MAC adversary A,

there exists a PPT PRF adversary B attacking F s.t.:

$$Adv_{MAC}[A, I_F] \leq Adv_{PRF}[B, F] + 1/|Y|$$

$\Rightarrow$ I_F is secure as long as |Y| is large, say $|Y| = 2^{80}$.

A Simple Lemma about Unpredictability

Let **F: K×X⟶Y** be a pseudorandom function.

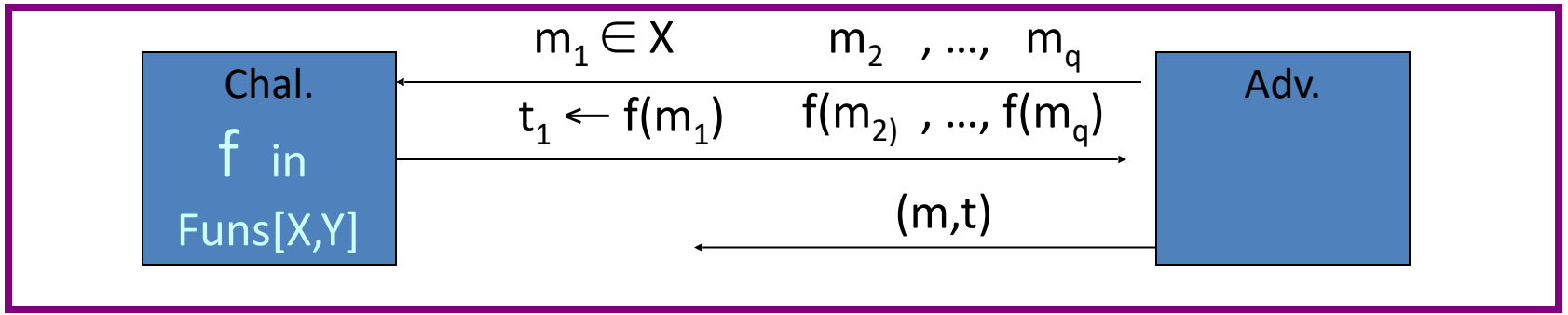
- Consider an adversary who requests and obtains $F_k(x_1), \ldots, F_k(x_q)$ for a polynomial $q = q(n)$.
- Can she predict $F_k(x^\star)$ for some x^* of her choosing where $x^* \notin \{x_1, \ldots, x_q\}$? How well can she do it?

Lemma: If she succeeds with probability $\frac{1}{2^m} + 1/\mathrm{poly}(n)$, then she broke PRF security.

Proof Sketch

Suppose F: **X** $\longrightarrow$ **Y** is a truly random function

Then MAC adversary A must win the following game:



A wins if **t = f(m)** and **$m \notin \{ m_1 , \ldots , m_q \}$**

$\Rightarrow$ Pr[A wins] = 1/|Y|

By PRF security,
same must hold for F(k,x)

MACs and PRFs

So far: secure PRF **F** $\Rightarrow$ secure MAC, as long as $|Y|$ is large

$$MAC(k, m) = F(k, m)$$

Our goal:

given a PRF for short messages (AES)

construct a PRF for long messages

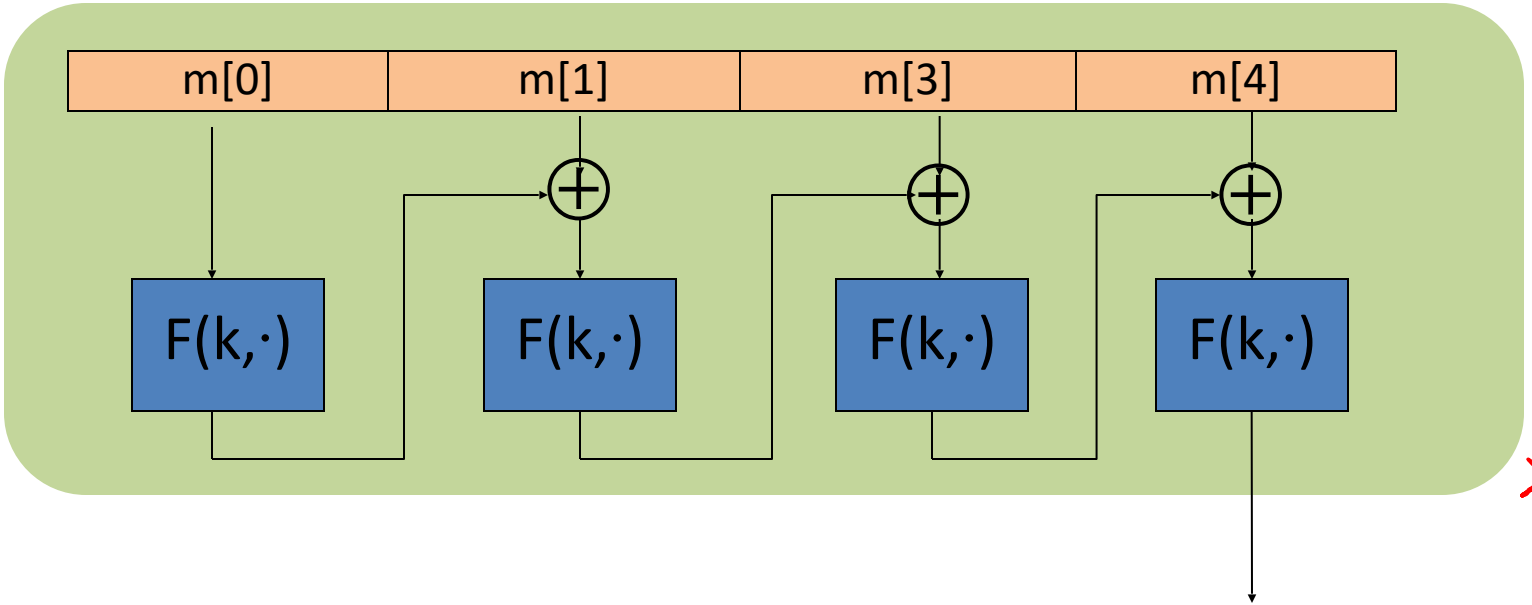
From here on let $X = \{0,1\}^n$ (e.g. n=128)

Ideas?

On board: rand-CTR-like scheme

Construction Attempt: just CBC-MAC

raw CBC



$$x^{\leq \ell} = \bigcup_{i=1}^{\ell} x^i$$

Why is this broken?

rawCBC is easily broken using a 1-chosen msg attack.

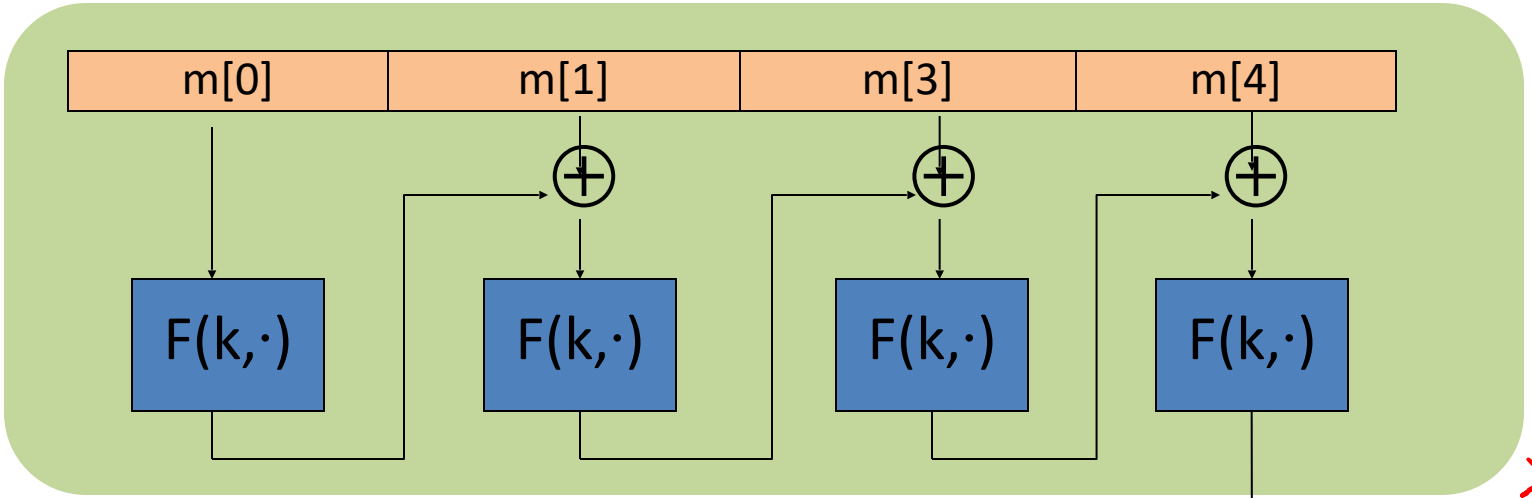
Adversary works as follows:

- Choose an arbitrary one-block message $m \in X$
- Request tag for m. Get $t = F(k,m)$
- Output t as MAC forgery for the 2-block message $(m, t \oplus m)$

Indeed: $\text{rawCBC}(k, (m, t \oplus m)) = F(k, F(k,m) \oplus (t \oplus m)) = F(k, t \oplus (t \oplus m)) = t$

Construction: encrypted CBC-MAC

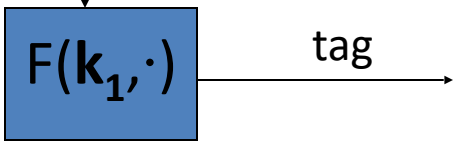
raw CBC



$X^{\leq L} = \bigcup_{i=1}^{L} X^i$

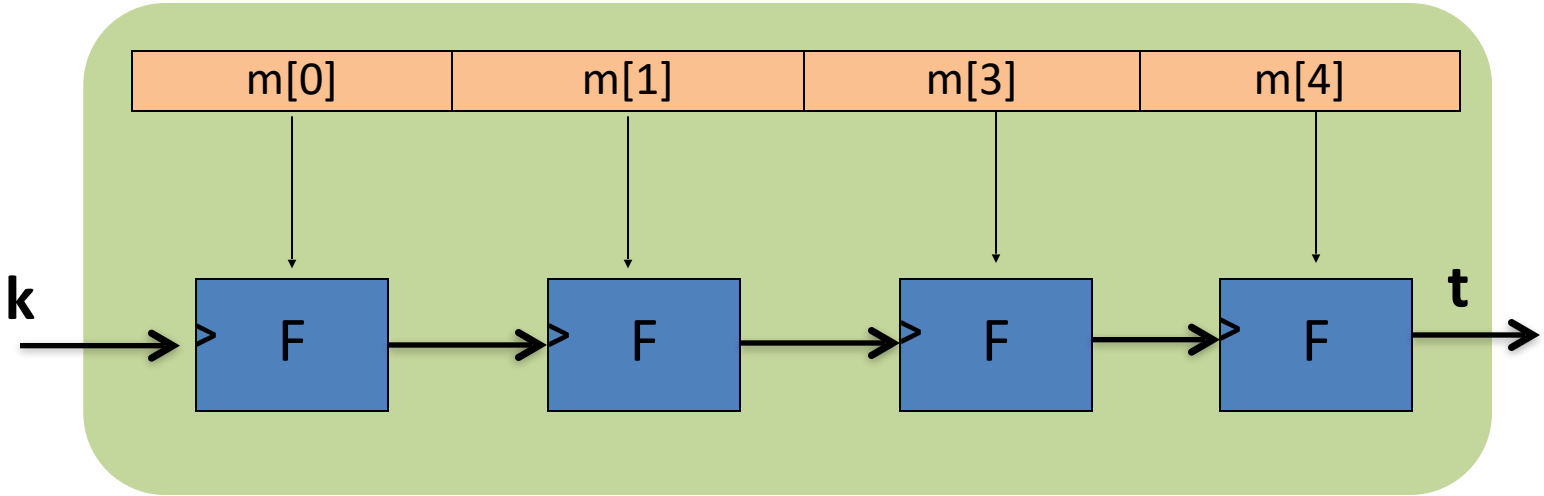
Let $F: K \times X \longrightarrow X$ be a PRP

Define new PRF $F_{ECBC}: K^2 \times X^{\leq L} \longrightarrow X$



Construction Attempt: Just Cascade

cascade



Does this work?

This MAC is secure

This MAC can be forged without any chosen msg queries

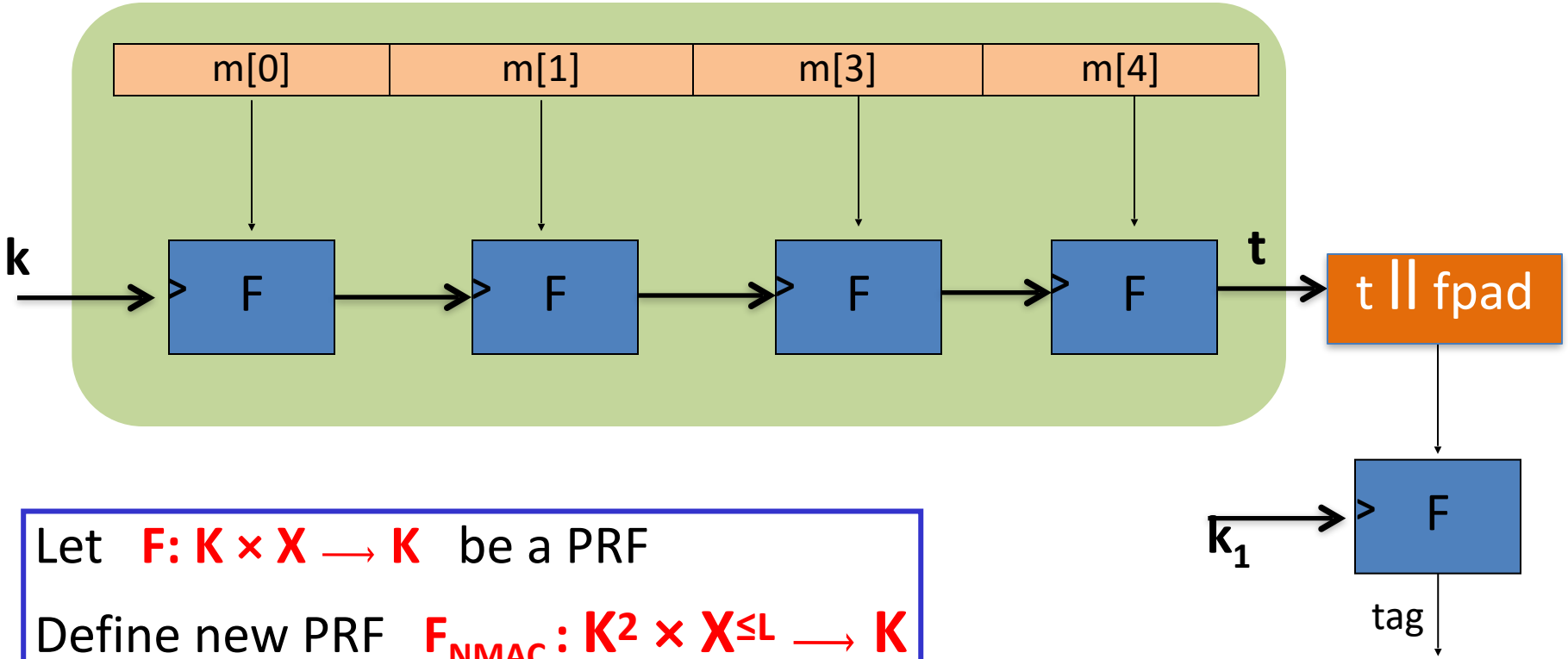
This MAC can be forged with one chosen msg query

This MAC can be forged, but only with two msg queries

$cascade(k, m) \Rightarrow cascade(k, m \| w)$ For any w

Construction: NMAC (nested MAC)

cascade



Let $F: K \times X \longrightarrow K$ be a PRF

Define new PRF $F_{NMAC}: K^2 \times X^{\leq L} \longrightarrow K$

Comparison

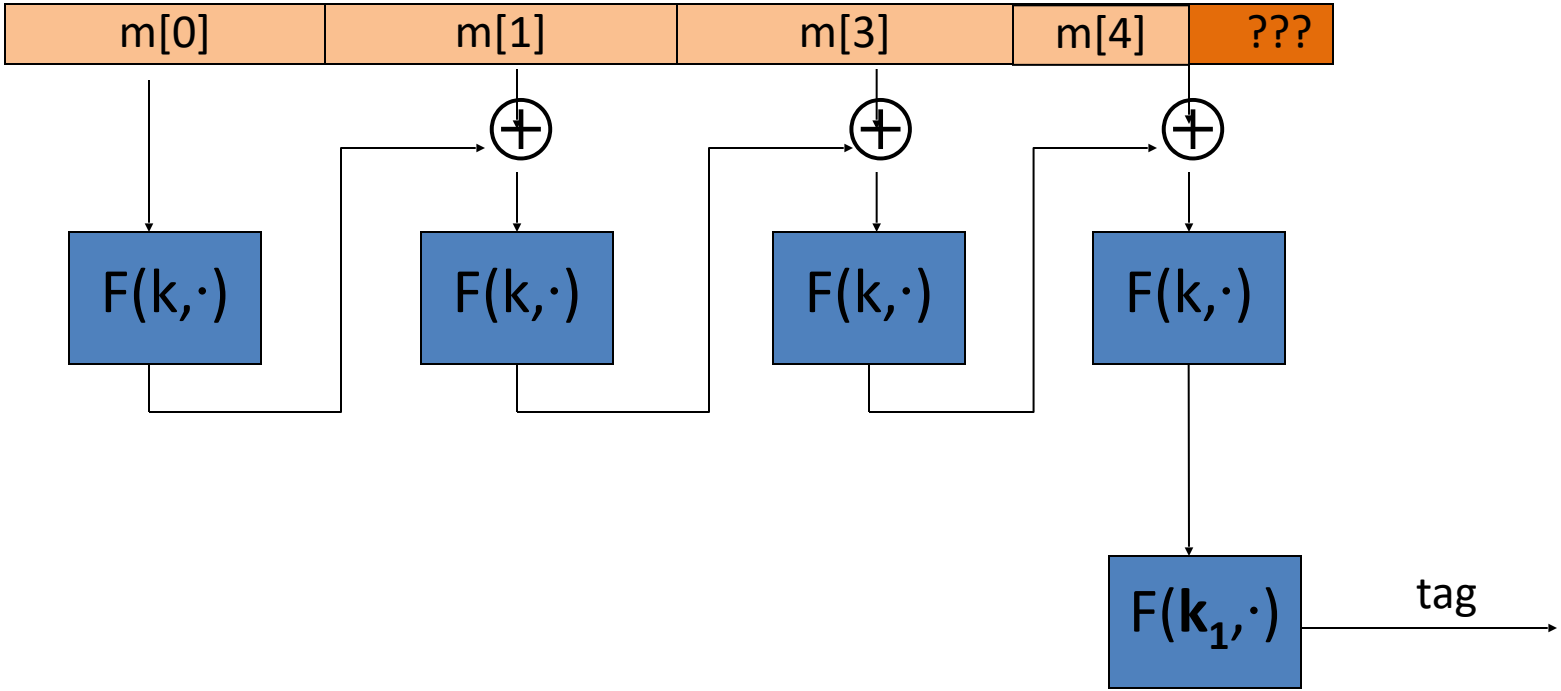
ECBC-MAC is commonly used as an AES-based MAC

- CCM encryption mode (used in 802.11i)
- NIST standard called CMAC

NMAC not usually used with AES or 3DES

- Main reason: need to change AES key on every block
  requires re-computing AES key expansion
- But NMAC is the basis for a popular MAC called HMAC (next)

What if msg. len. is not multiple of block-size?



CBC MAC padding

Bad idea: pad m with 0's

Is the resulting MAC secure?

- Yes, the MAC is secure
- It depends on the underlying MAC
- ⇒ No, given tag on msg **m** attacker obtains tag on **mll0**
-

Problem: pad(m) = pad(mll0)

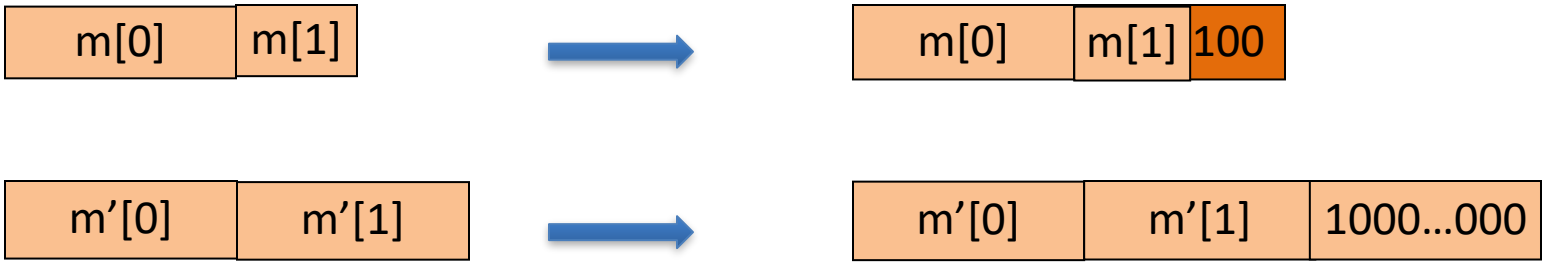
CBC MAC padding

For security, padding must be invertible !

$$m_0 \neq m_1 \quad \Rightarrow \quad pad(m_0) \neq pad(m_1)$$

ISO: pad with "1000…00". Add new dummy block if needed.

– The "1" indicates beginning of pad.

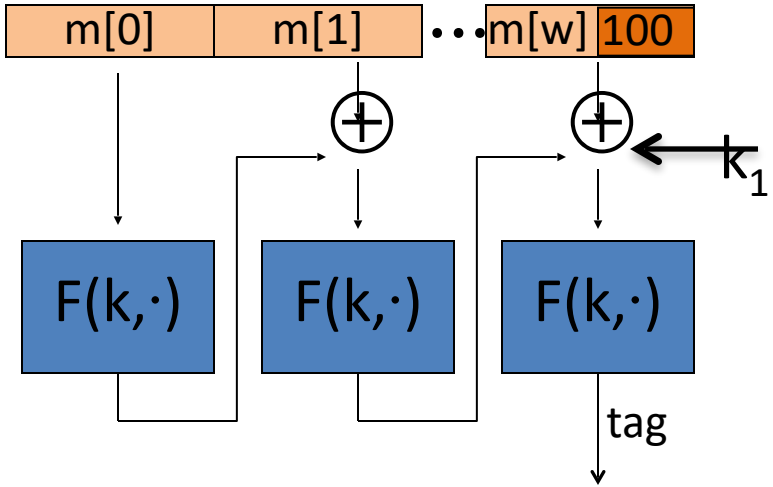


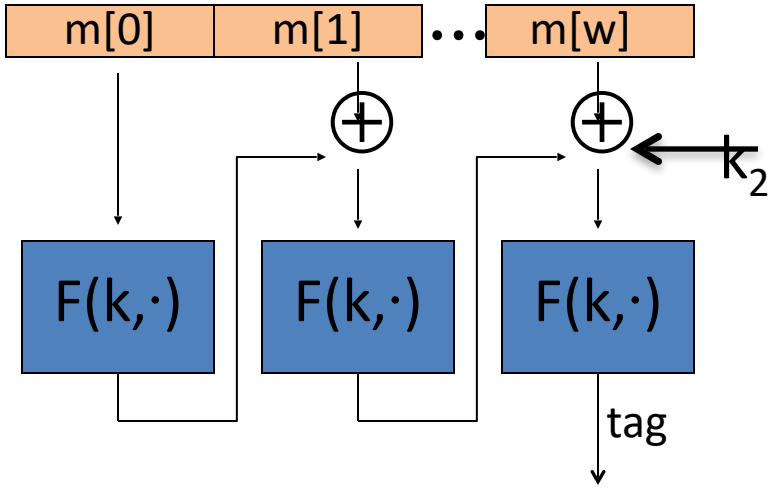
CMAC (NIST standard)

(k_1, k_2) derived from k

Variant of CBC-MAC where key = (k, k_1, k_2)

- No final encryption step (extension attack thwarted by last keyed xor)
- No dummy block (ambiguity resolved by use of k_1 or k_2)





Collision Resistance

Let $H: M \rightarrow T$ be a hash function ($|M| >> |T|$)

A **collision** for H is a pair $m_0, m_1 \in M$ such that:

$$H(m_0) = H(m_1) \quad \text{and} \quad m_0 \neq m_1$$

A function H is **collision resistant** if for all efficient algs. A:

$$\mathbf{Adv_{CR}[A,H] = Pr[A \text{ outputs collision for H}]}$$

is "neg".

Example: SHA-256 (outputs 256 bits)

Formal Definition: Collision-Resistant Hash Functions

A compressing **family of functions** $\mathcal{H} = \{h : \{0,1\}^m \to \{0,1\}^n\}$ (where $m > n$) for which it is computationally hard to find collisions.

Def: $\mathcal{H}$ is collision-resistant if for every PPT algorithm A, there is a negligible function μ s.t.

$$\Pr_{h \leftarrow \mathcal{H}}\left[A\left(1^n, h\right) = \left(x, y\right) : x \neq y,\ h(x) = h\left(y\right)\right] = \mu(n)$$

MACs from Collision Resistance

Let MAC be a MAC for short messages over (K,M,T) (e.g. AES)

Let H: $M^{big} \rightarrow M$ be a hash function

Def: $MAC^{big} = (MAC^{big}, Ver^{big})$ over (K, M^{big}, T) as:

$MAC^{big}(k,m) = S(k,H(m))$; $Ver^{big}(k,m,t) = V(k,H(m),t)$

Thm: If MAC is a secure MAC and H is collision resistant

then MAC^{big} is a secure MAC.

Example: $MAC(k,m) = AES_{2\text{-block-cbc}}(k, SHA\text{-}256(m))$ is a secure MAC.

MACs from Collision Resistance

$\mathbf{MAC^{big}(k, m) = MAC(k, H(m))}$;

$\mathbf{Ver^{big}(k, m, t) = V(k, H(m), t)}$

Collision resistance is necessary for security:

Suppose adversary can find $m_0 \neq m_1$ s.t. $H(m_0) = H(m_1)$.

Then: $\mathbf{MAC^{big}}$ is insecure under a 1-chosen msg attack

step 1: adversary asks for $t \leftarrow MAC(k, m_0)$

step 2: output (m_1, t) as forgery

How easy is it to find collisions?

Generic attack on CRHFs

Let $H : \mathcal{M} \rightarrow \{0,1\}^n$ be a hash function $(|\mathcal{M}| \gg 2^n)$

Generic algorithm to find a collision **in time O($2^{n/2}$)** hashes:

Algorithm:

1. Choose $2^{n/2}$ random messages in $\mathcal{M}$: $m_1, \ldots, m_{2^{n/2}}$ (distinct w.h.p)
2. For $i = 1,\ldots,2^{n/2}$ compute $t_i = H(m_i) \in \{0,1\}^n$
3. Look for a collision $(t_i = t_j)$. If not found, go back to step 1.

How well will this work?

The birthday paradox

Let $r_1, \ldots, r_n \in \{1, \ldots, B\}$ be IID integers.

Thm: When $n \approx \sqrt{B}$ then $\Pr[r_i = r_j \mid \exists i \neq j] \geq \frac{1}{2}$

Proof: for uniformly independent $r_1, \ldots, r_n$,

$$\Pr\left[\exists i \neq j : r_i = r_j\right] = 1 - \Pr\left[\forall i \neq j : r_i \neq r_j\right] = 1 - \left(\frac{B-1}{B}\right)\left(\frac{B-2}{B}\right) \cdots \left(\frac{B-n+1}{B}\right) =$$

$$= 1 - \prod_{i=1}^{n-1}\left(1 - \frac{i}{B}\right) \geq 1 - \prod_{i=1}^{n-1} e^{-i/B} = 1 - e^{-\frac{1}{B}\sum_{i=1}^{n-1} i} \geq 1 - e^{-n^2/2B}$$

$1 - x \leq e^{-x}$

$$\frac{n^2}{2B} = 0.72 \qquad \geq 1 - e^{-0.72} = 0.53 > \frac{1}{2}$$