Domain Extension for MAC/PRF

RECALL: $F^* = F(\mathcal{H})$

If F a PRF with domain $\{0,1\}^n$

$\mathcal{H}$ AU mapping $\{0,1\}^{nt} \to \{0,1\}^n$

Then F^* is a PRF (and thus a MAC) for domain $\{0,1\}^{nt}$

In the last lecture: information-theoretic almost universality for unbounded attackers.

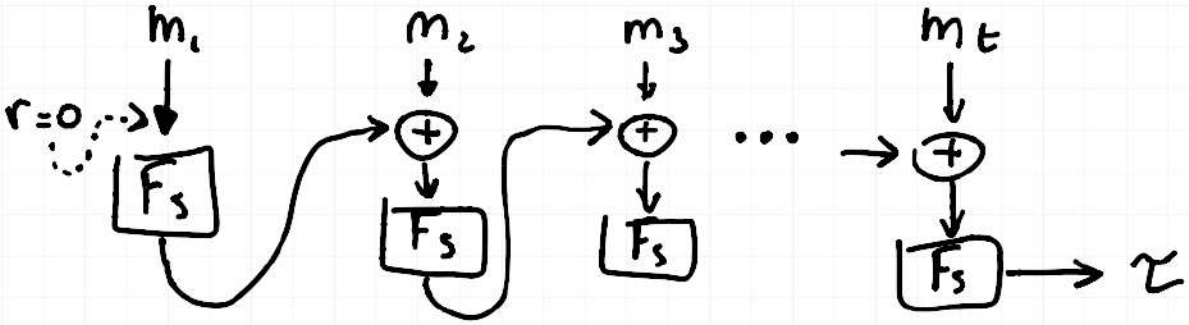
The alternative is COMPUTATIONAL AU out of

$$F = \left\{ F_S : \{0,1\}^n \to \{0,1\}^n \right\}_{S \leftarrow \$ \{0,1\}^\lambda}$$

$$F^* = F(\mathcal{H})$$

This leads to the standardised CBC-MAC in which

$$h_S(m_1, m_2, \ldots m_t) = F_S(m_t \oplus F_S(m_{t-1} \oplus \ldots \oplus F(m_1)) \ldots)$$



THM: Above $\mathcal{H}$ is COMPUTATIONAL AU

$\Rightarrow F(\mathcal{H})$ is a long-input PRF

Even better: $\mathcal{H}$ is directly a FIL MAC!

EXERCISE: Show that CBC-MAC is

1) Not secure as a VIL MAC
2) Not secure as a FIL MAC is $r \neq 0^n$
3) Not secure as a FIL MAC if all blocks are output

XOR-MAC: Different construction

$$Tag(k,m) = (r, F_k(r)) \oplus h_s(m))$$
$$r \leftarrow\$ \{0,1\}^n$$

What property from $\mathcal{H}$? AU is not sufficient...

Given $\tau = (r,z)$, compute

$$r^* = (r, z \oplus a) \text{ for some } a \in \{0,1\}^n.$$

So long as $h_s(m') = h_s(m) \oplus a$ for $m' \neq m$

DEF: $\mathcal{H}$ is ALMOST XOR UNIVERSAL (AXU) if

$$\forall m, m', \forall a: \Pr_{s \leftarrow\$ \{0,1\}^\lambda}[h_s(m) \oplus h_s(m') = a] \leq negl(|s|)$$

THM: XOR-MAC is a long-input MAC if F a PRF and $\mathcal{H}$ is AXU.

COMP. AXU : $h_s(m) = F_s(m_1 \| 1) \oplus \dots \oplus F_s(m_b \| t)$

But what about VIL?

CONST.	FIL	VIL
$\mathcal{F}(\mathcal{H})$	✓	Not in general (depends on $\mathcal{H}$)
XOR-MAC (AXU)	✓	✓
CBC-MAC	✓	X
E-CBC-MAC	✓	✓

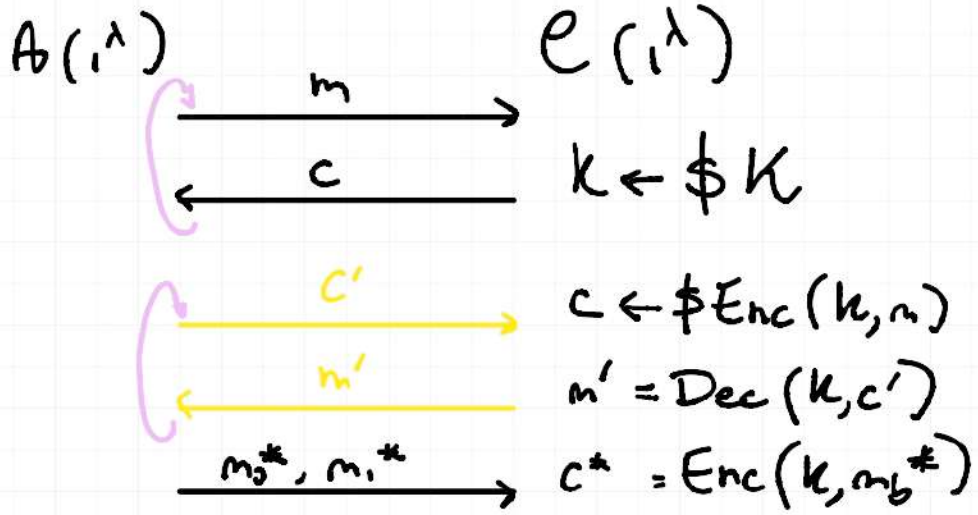
Chosen-ciphertext security

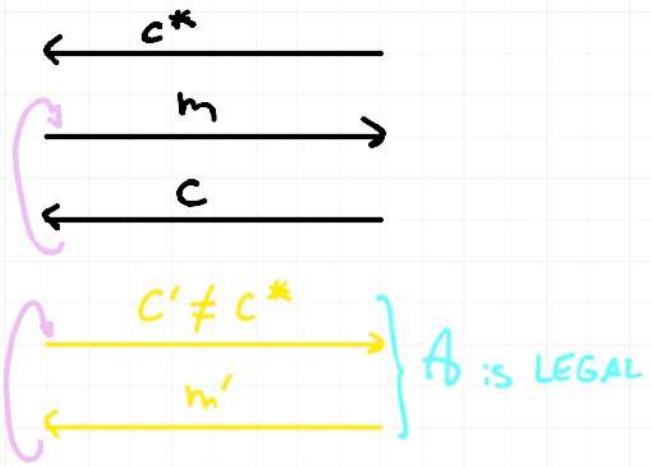
Q: CPA security there are no DECRYPTION QUERIES

Real Threat: Attack on TLS

↳ CHOSEN-CIPHERTEST ATTACK

$GAME^{cca}_{\Pi,A}(\lambda,b)$





DEF: SKE Π is CCA secure if $\forall$ PPT LEGAL A

$$\text{GAME}^{CCA}_{\Pi,A}(\lambda,0) \approx_c \text{GAME}^{CCA}_{\Pi,A}(\lambda,1)$$

RECALL: CPA SKE $Enc(k,m) = (r, F_k(r) \oplus m)$

Not CCA secure!

Problem: this SKE is malleable

$A(1^\lambda)$		$C(1^\lambda)$
	$\xrightarrow{m_0^* = 0^n,\ m_1^* = 1^n}$	$k \leftarrow \$ \{0,1\}^\lambda$
	$\xleftarrow{c^* = (r^*, s^*)}$	$c^* = (r^*, F_k(r^*) \oplus m_b^*)$
$s' = s^* \oplus 1 \| 0^{n-1}$	$\xrightarrow{c' = (r^*, s')}$	$r^* \leftarrow \$ \{0,1\}^n$
	$\xleftarrow{m'}$	

IF $m' = 1 \| 0^{n-1}$
output $b' = 0$
IF $m' = 0 \| 1^{n-1}$
output $b' = 1$

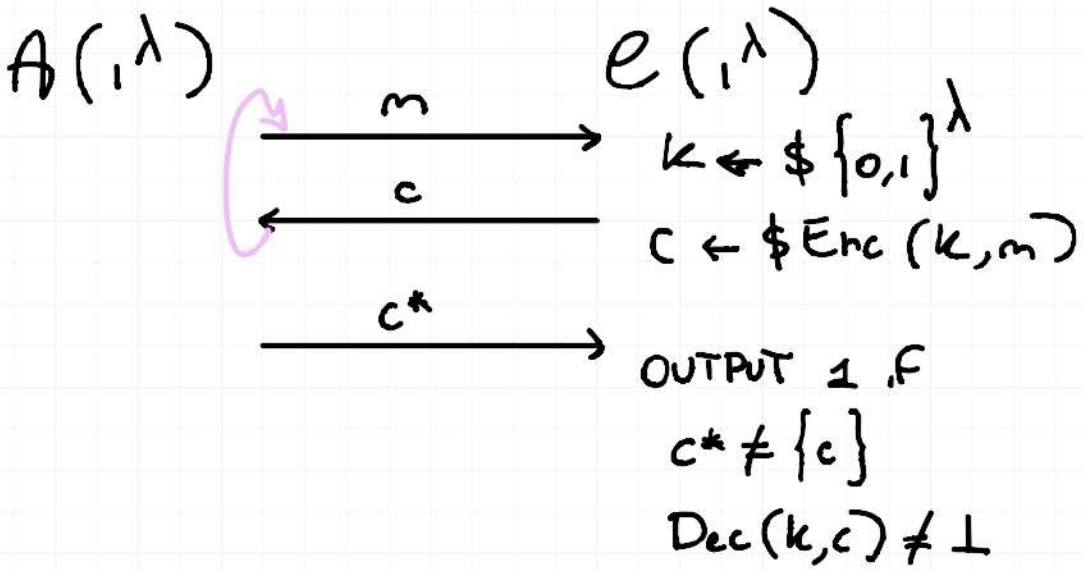
The attacker can distinguish succesfully the encryption of m_0^* from m_1^*

Recipe for CCA security. Seek for AUTHENTICITY
should be hard to produce VALID cyphertexts without knowing the key

VALID : $Dec(k, c) = m \neq \perp$ ← invalid symbol

if t outputs $\perp$, c is invalid

$GAME^{auth}_{\Pi, A}(\lambda)$



DEF: SKE Π is AUTH if $\forall$ PPT A

$Pr[GAME^{auth}_{\Pi,A}(\lambda) = 1] \leq negl(\lambda)$ ← the prob. that A wins the GAME

THM Assume Π satisfies CPA + AUTH
Then Π is CCA secure

Constructions: The main idea is to combine CPA + MAC

① Encrypt-and-MAC:

$$c \leftarrow \$\, Enc(k_1, m)$$
$$\tau = Tag(k_2, m)$$
$$c^* = (c, \tau)$$

This is not ALWAYS CCA secure

Because: if you take any Tag UF-CMA and consider $Tag^{BAD}(k, m) = m[1] \| Tag(k, m)$

$= \tau$

It still is UF-CMA (?)

② Authenticate-then-encrypt

$$\tau = Tag(k_1, m)$$

$$c \leftarrow \$\, Enc(k_2, m \| \tau) \quad c^* = c$$

This also is not always CCA secure!

(yet this is used in TLS) for a specifically chosen set of Enc functions that can be proved secure

③ Encrypt-then-authenticate

$$c \leftarrow \$\, Enc(k_1, m)$$

$$\tau = \text{Tag}(k_2, c) \qquad c^* = (c, \tau)$$

This construction is ALWAYS CCA secure!