CSCI 1510

- CBC-MAC (continued)
- CCA-Security & Unforgeability
- Authenticated Encryption

Message Authentication Code (MAC)

- **Syntax:**

  A message authentication code (MAC) scheme is defined by PPT algorithms (Gen, Mac, Vrfy):

  $$k \leftarrow Gen(1^n)$$

  $$t \leftarrow Mac_k(m) \qquad m \in \{0,1\}^*$$

  $$0/1 := Vrfy_k(m,t)$$

- **Correctness:** $\forall n$, $\forall k$ output by $Gen(1^n)$, $\forall m \in \{0,1\}^*$

  $$Vrfy_k(m, Mac_k(m)) = 1$$

- **Canonical Verification:**

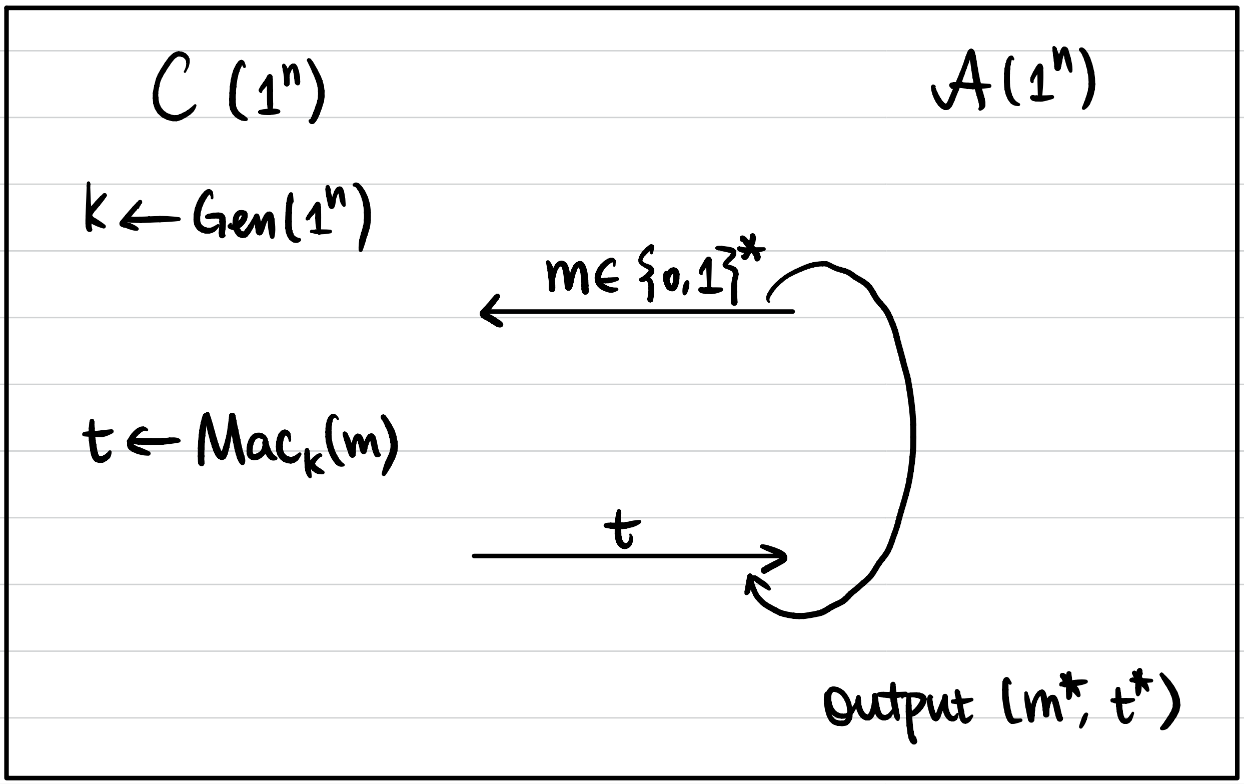
  If $Mac_k(m)$ is derterministic, then $Vrfy_k(m,t)$ is straightforward.

  $$Mac_k(m) \stackrel{?}{=} t$$

Message Authentication Code (MAC)

Def 1 A message authentication code (MAC) scheme $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ is existentially unforgeable under adaptive chosen message attack, or EU-CMA-secure, or secure, if $\forall$ PPT $\mathcal{A}$, $\exists$ negligible function $\varepsilon(\cdot)$ s.t.

$$\Pr[\mathsf{MacForge}_{\mathcal{A},\Pi} = 1] \leq \varepsilon(n).$$



$Q := \{m \mid m \text{ queried by } \mathcal{A}\}$

$\mathsf{MacForge}_{\mathcal{A},\Pi} = 1$ ($\mathcal{A}$ succeeds) if

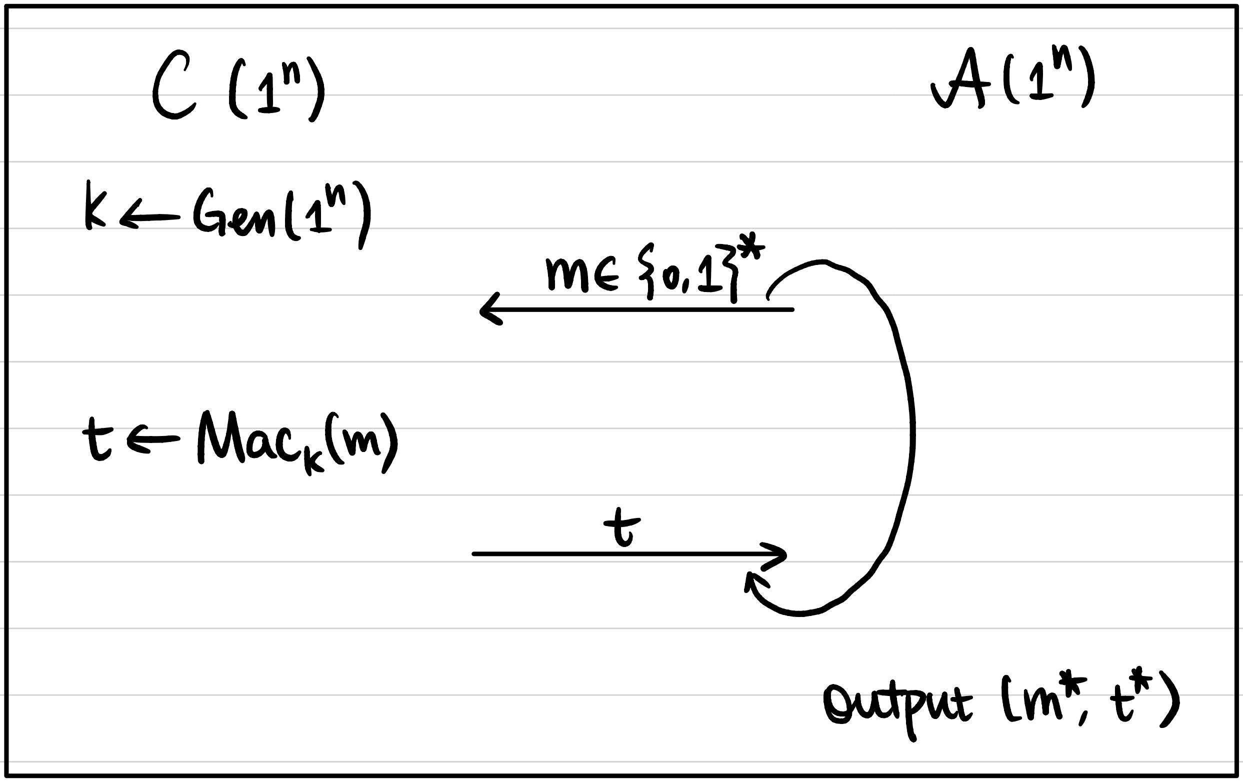
① $m^* \notin Q$, and

② $\mathsf{Vrfy}_k(m^*, t^*) = 1$.

Message Authentication Code (MAC)

Def 2 A message authentication code (MAC) scheme $\Pi = (\text{Gen}, \text{Mac}, \text{Vrfy})$ is strongly secure if $\forall$ PPT $\mathcal{A}$, $\exists$ negligible function $\varepsilon(\cdot)$ s.t.

$$\Pr[\text{MacForge}^{s}_{\mathcal{A},\Pi} = 1] \le \varepsilon(n).$$



$Q := \{(m,t) \mid m$ queried by $\mathcal{A}$, t is the response$\}$

$\text{MacForge}^{s}_{\mathcal{A},\Pi} = 1$ ($\mathcal{A}$ succeeds) if

① $(m^*, t^*) \notin Q$, and

② $\text{Vrfy}_k(m^*, t^*) = 1$.

Thm If $\Pi = (\text{Gen}, \text{Mac}, \text{Vrfy})$ is a secure MAC with canonical verification (Mac is a deterministic algorithm), then Π is also strongly secure.

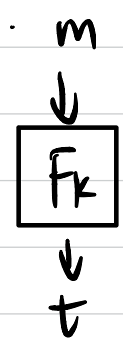
$m^* \neq m$

Fixed-Length MAC

Let $F: \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a PRF.

Construct a MAC scheme:

- $Gen(1^n)$: sample $k \xleftarrow{\$} \{0,1\}^n$, output k.
- $Mac_k(m)$: $m \in \{0,1\}^n$
  output $t := F_k(m)$
- $Vrfy_k(m,t)$: $F_k(m) \stackrel{?}{=} t$



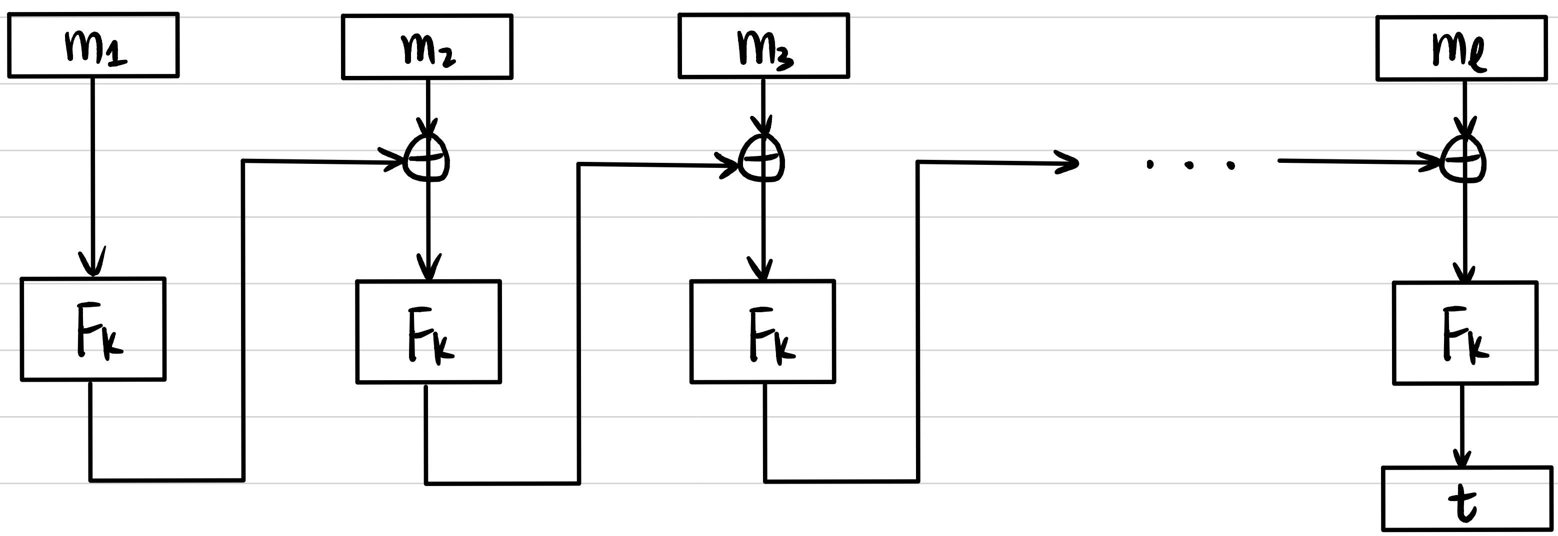
<u>Thm</u> If F is a PRF, then $\Pi = (Gen, Mac, Vrfy)$ is a secure MAC scheme for fixed-length messages of length n.

CBC-MAC (for fixed-length messages)

Let $F: \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be a PRF.

Construct a MAC scheme for messages of length $\ell(n) \cdot n$:

- $Gen(1^n)$: sample $k \xleftarrow{\$} \{0,1\}^n$, output k.
- $Mac_k(m)$: $m \in \{0,1\}^{\ell(n)\cdot n}$ $\quad m = m_1 \| m_2 \| \cdots \| m_\ell \quad m_i \in \{0,1\}^n$



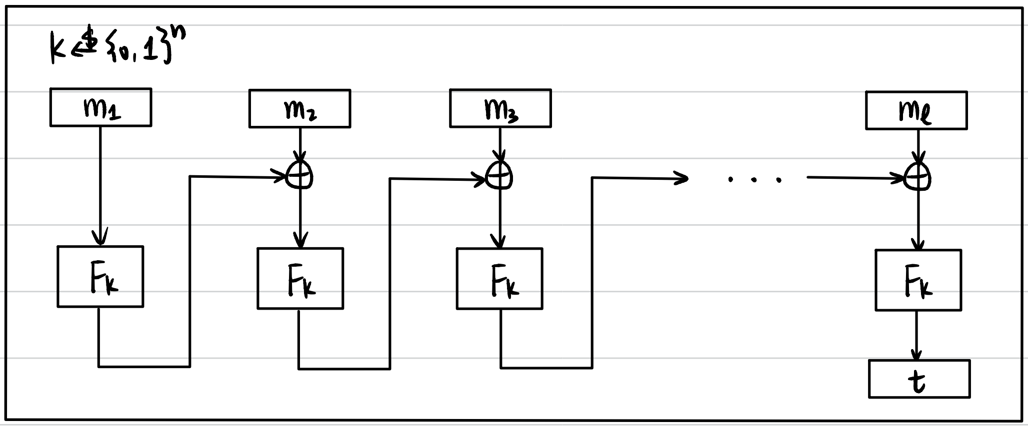
- $Vrfy_k(m,t)$: $Mac_k(m) \stackrel{?}{=} t$

<u>Thm</u> If F is a PRF, then CBC-MAC is a secure MAC scheme for fixed-length messages of length $\ell(n) \cdot n$.

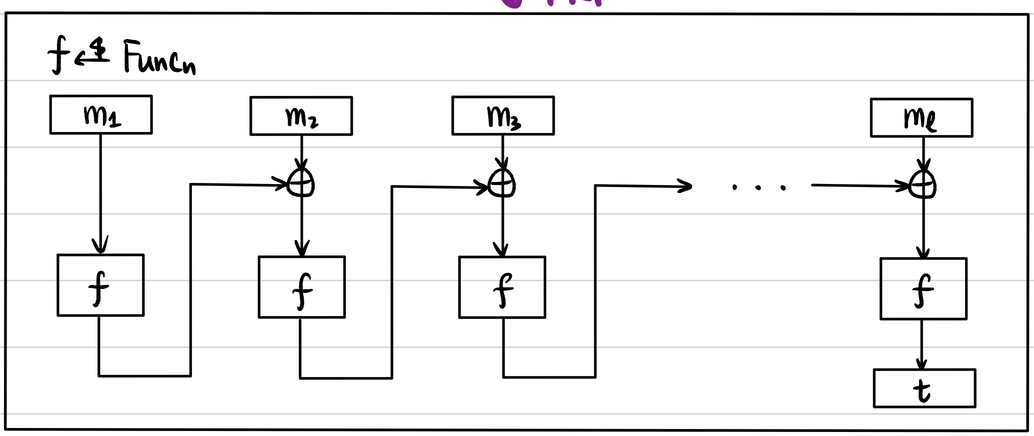
Thm If F is a PRF, then CBC-MAC is a secure MAC scheme for fixed-length messages of length $\ell(n)\cdot n$.

Proof Sketch $\quad Mac: \{0,1\}^n \times \{0,1\}^{\ell(n)\cdot n} \to \{0,1\}^n$

Suffices to show Mac is a PRF.



↕ PRF



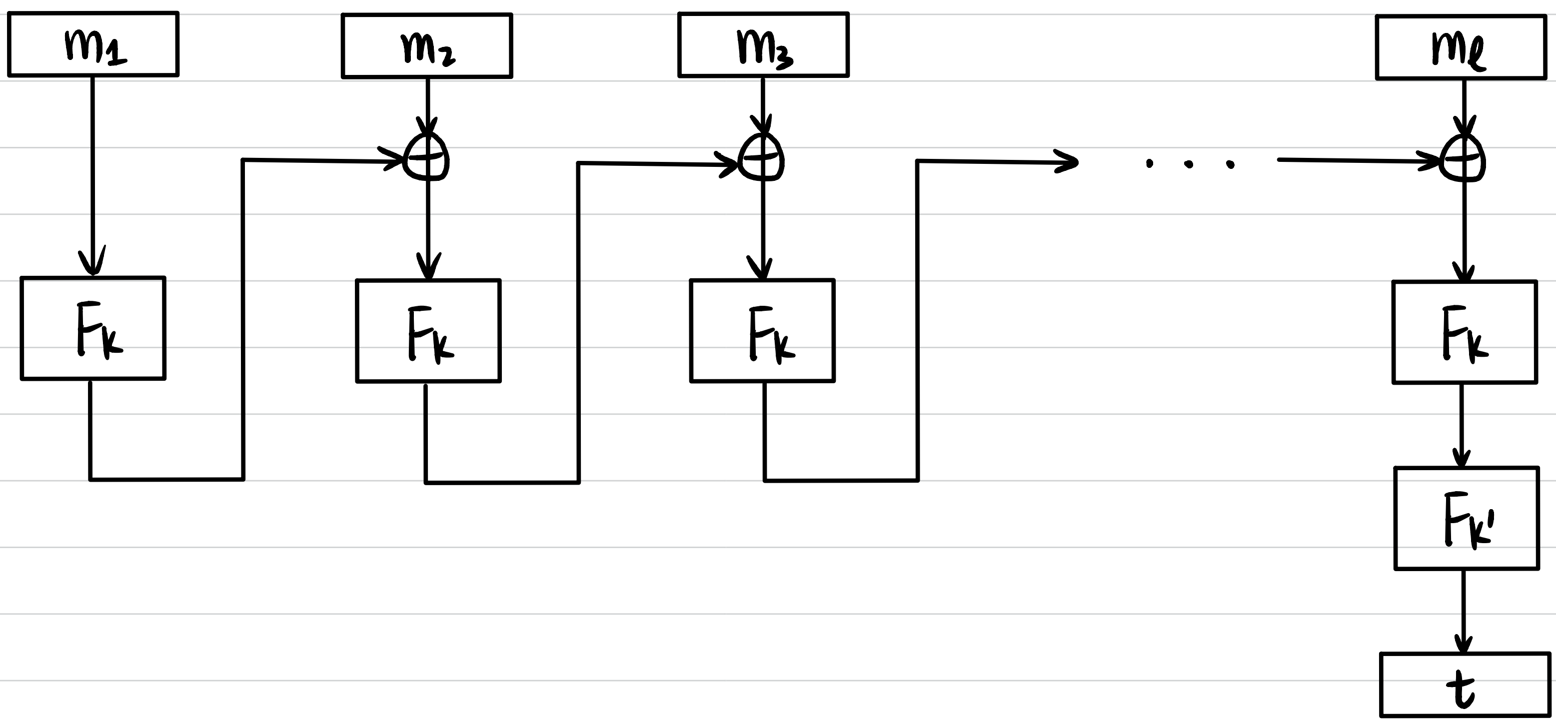
↕ statistical

$g \xleftarrow{\$} \{h \mid h: \{0,1\}^{\ell(n)\cdot n} \to \{0,1\}^n\}$

$t := g(m_1 \| \cdots \| m_\ell)$

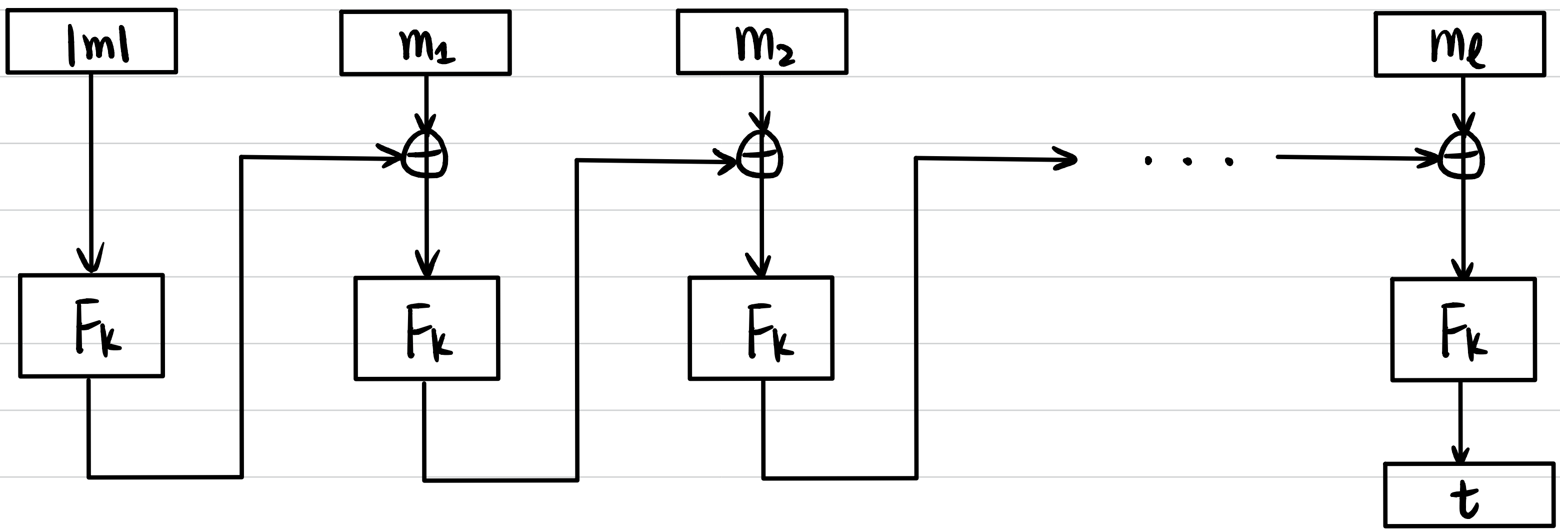
MAC for messages of arbitrary length (multiple of n)

Approach 1: MAC of CBC-MAC

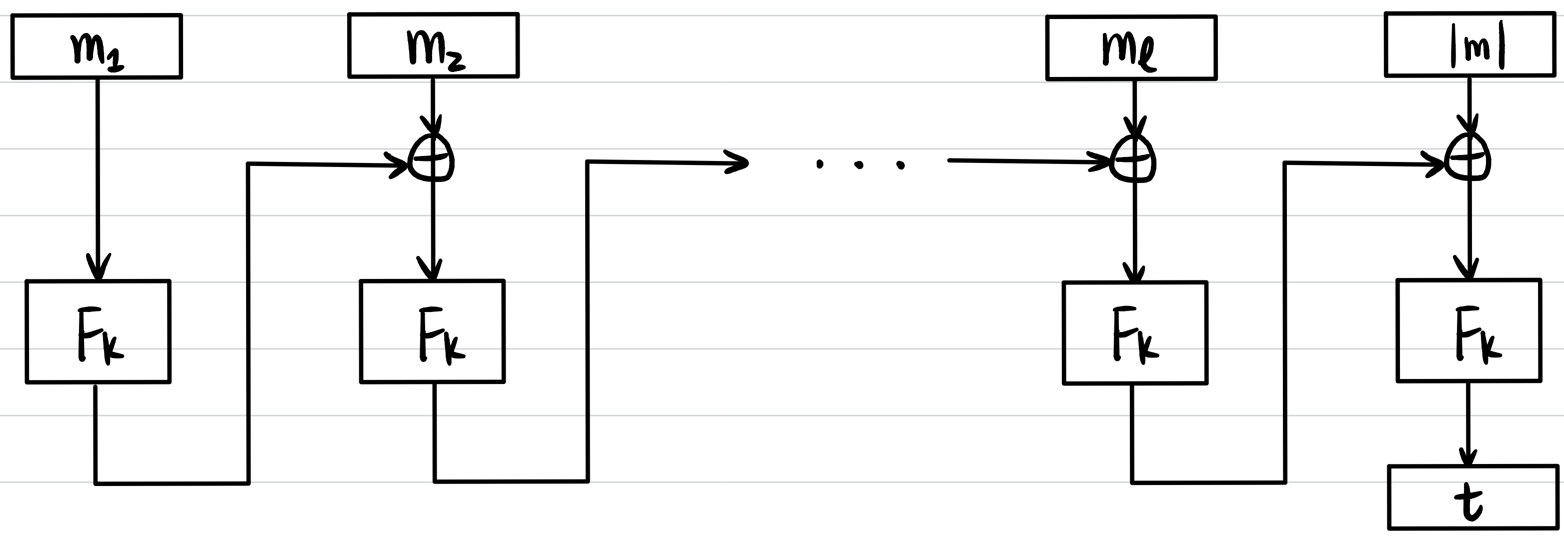


MAC for messages of arbitrary length (multiple of n)

Approach 2: CBC-MAC on $|m| \| m$

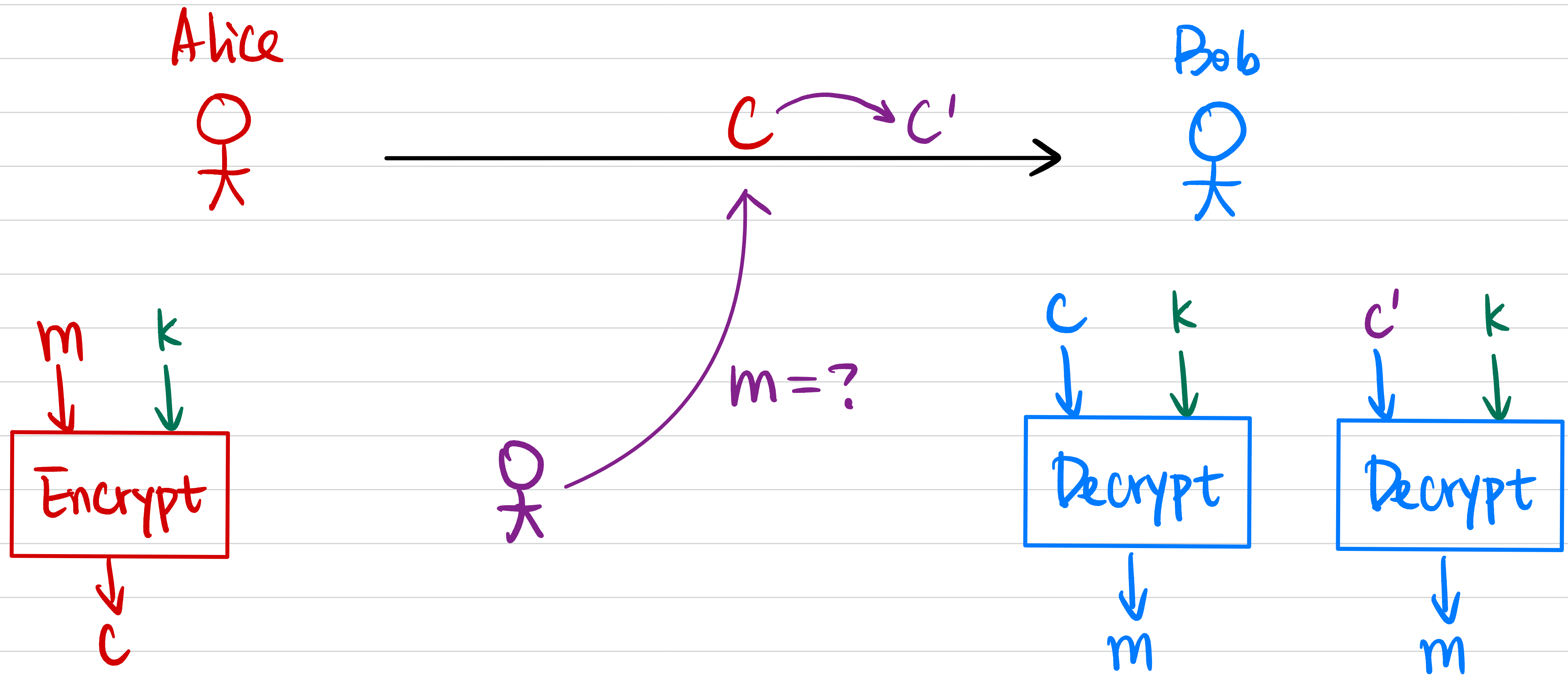


Exercises



Show this is not a secure MAC for messages of arbitrary length (multiple of n).

Authenticated Encryption

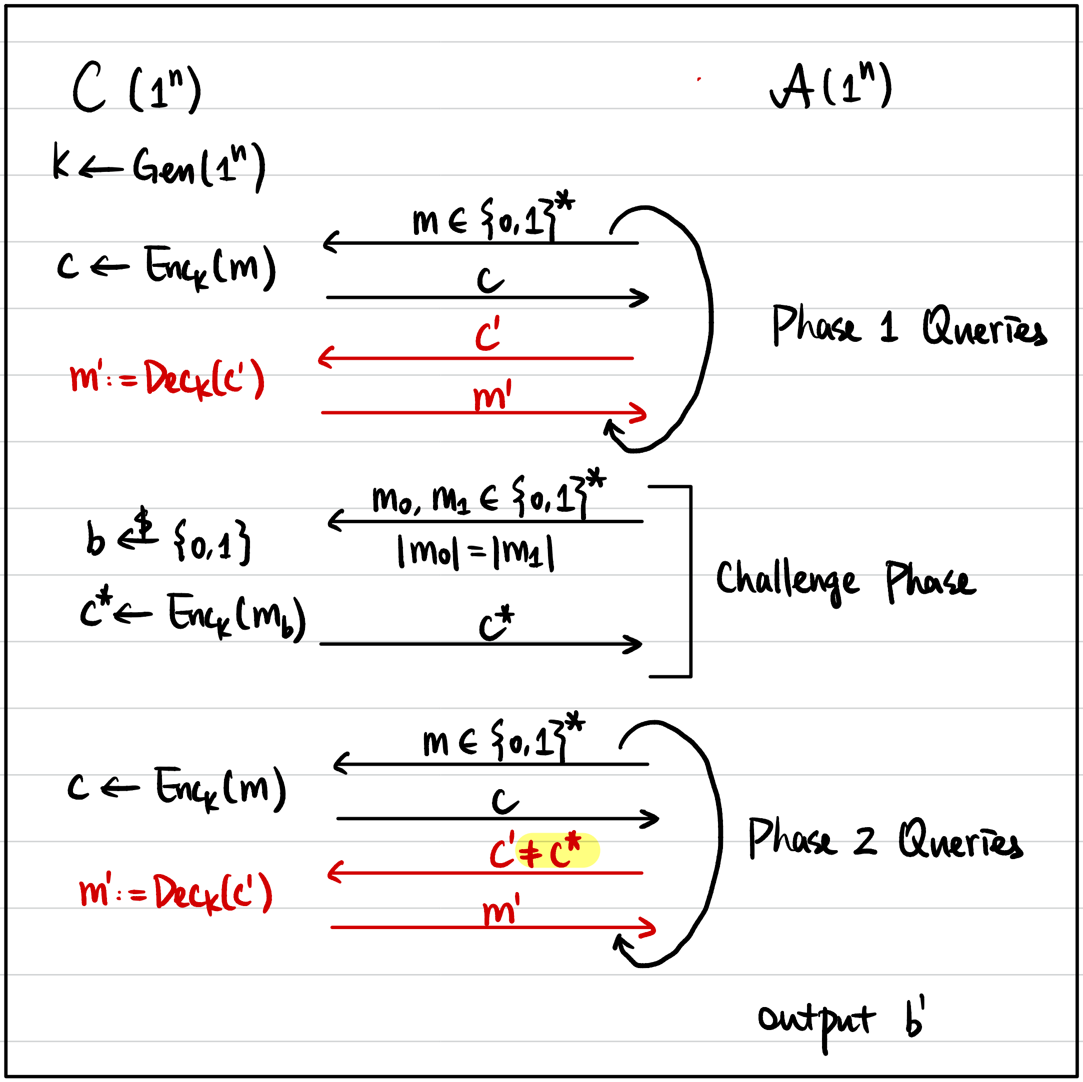


Security Guarantees:

- Message Secrecy: CCA Security
- Message Integrity: Unforgeability

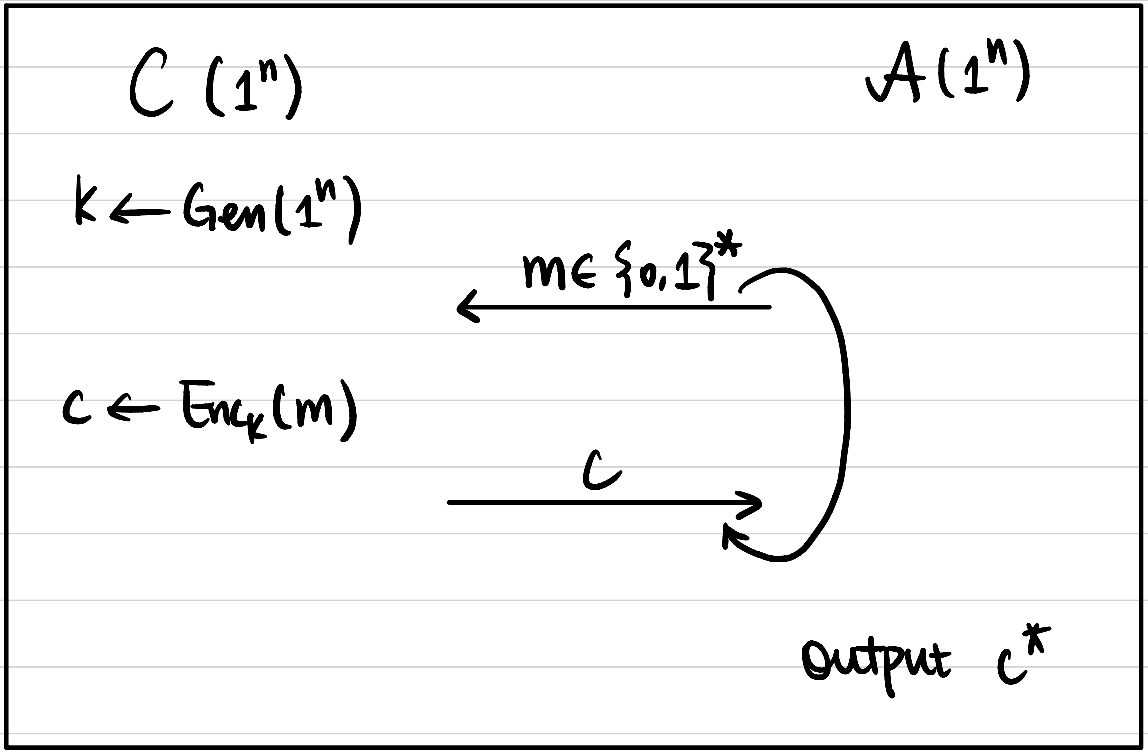
Chosen Ciphertext Attack (CCA) Security

Def A symmetric-key encryption scheme (Gen, Enc, Dec) is **secure against chosen ciphertext attacks**, or **CCA-secure**, if $\forall$ PPT $\mathcal{A}$, $\exists$ negligible function $\varepsilon(\cdot)$ s.t. $\Pr[b = b'] \le \frac{1}{2} + \varepsilon(n)$



Unforgeability

Def A symmetric-key encryption scheme $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is unforgeable if $\forall$ PPT $\mathcal{A}$, $\exists$ negligible function $\varepsilon(\cdot)$ s.t. $\Pr[\mathsf{EncForge}_{\mathcal{A},\Pi} = 1] \le \varepsilon(n)$.



$Q := \{m \mid m \text{ queried by } \mathcal{A}\}$

$m^* := \mathsf{Dec}_k(c^*)$

$\mathsf{EncForge}_{\mathcal{A},\Pi} = 1$ ($\mathcal{A}$ succeeds) if

① $m^* \notin Q$, and

② $m^* \neq \perp$

Def A symmetric-key encryption scheme is authenticated encryption if it is CCA-secure and unforgeable.

Exercises

Is the CPA-secure encryption from PRF CCA-secure? Unforgeable?

$Enc_k(m)$: $m \in \{0,1\}^n$

$r \xleftarrow{\$} \{0,1\}^n$

output $c := \langle r, F_k(r) \oplus m \rangle$

Intuitions

Can we have an encryption scheme that is unforgeable but not CCA-secure?

Can we have an encryption scheme that is CCA-secure but not unforgeable?

Generic Constructions

Let $\Pi^E = (Gen^E, Enc^E, Dec^E)$ be a CPA-secure encryption scheme.
Let $\Pi^M = (Gen^M, Mac^M, Vrfy^M)$ be a strongly secure MAC scheme.

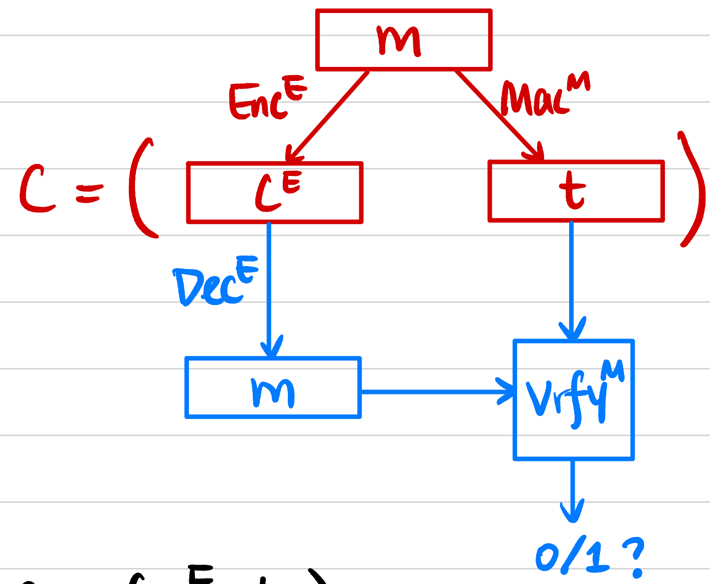
How to construct an authenticated encryption scheme?

① Encrypt-and-Authenticate

② Authenticate-then-Encrypt

③ Encrypt-then-Authenticate

Encrypt-and-Authenticate



$Gen(1^n)$:

$k^E \leftarrow Gen^E(1^n)$

$k^M \leftarrow Gen^M(1^n)$

Output $k = (k^E, k^M)$

$Enc_k(m)$:

$c^E \leftarrow Enc^E(k^E, m)$

$t \leftarrow Mac^M(k^M, m)$

output $c = (c^E, t)$

$Dec_k(c)$: $c = (c^E, t_c)$

$m := Dec^E(k^E, c^E)$

$b := Vrfy^M(k^M, (m, t))$

If $b = 1$, output m

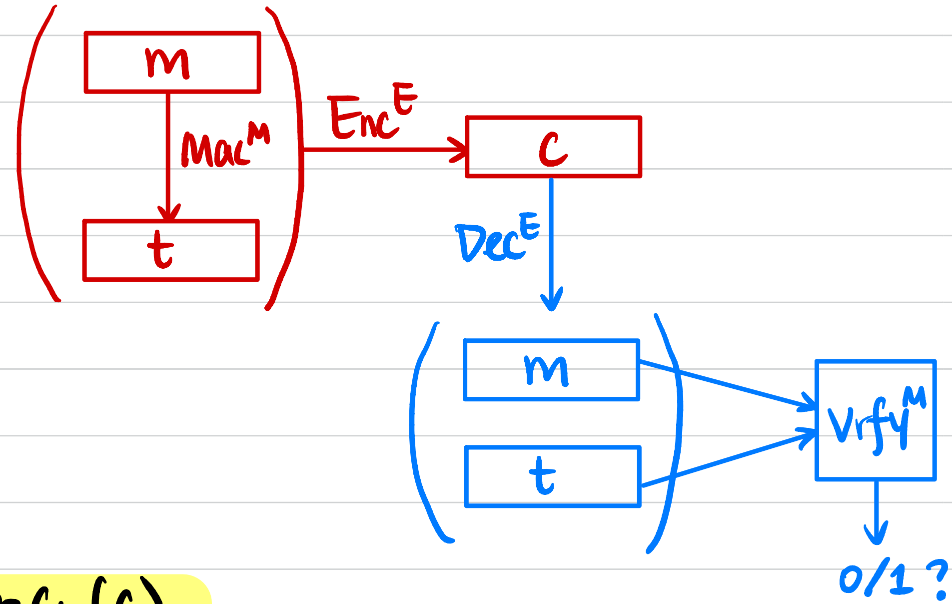
Otherwise output $\perp$

Q1: Is it CPA-secure?

Q2: Is it CCA-secure?

Q3: Is it unforgeable?

Authenticate-then-Encrypt



Gen(1^n):

$k^E \leftarrow Gen^E(1^n)$

$k^M \leftarrow Gen^M(1^n)$

Output $k = (k^E, k^M)$

Enc$_k$(m):

$t \leftarrow Mac^M(k^M, m)$

$c \leftarrow Enc^E(k^E, m\|t)$

Output c

Dec$_k$(c):

$m\|t := Dec^E(k^E, c)$

$b := Vrfy^M(k^M, (m, t))$

If $b = 1$, output m

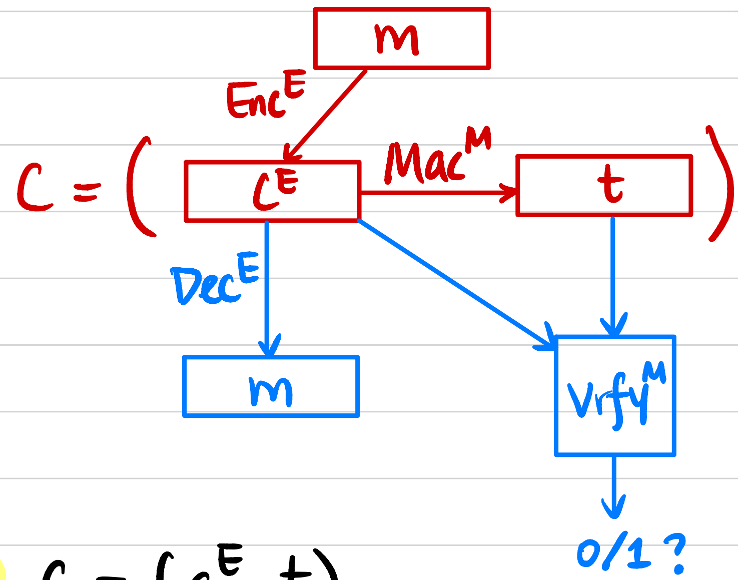
Otherwise output $\perp$

Q1: Is it CPA-secure?

Q2: Is it CCA-secure?

Q3: Is it unforgeable?

Encrypt-then-Authenticate



Gen(1^n):

$k^E \leftarrow Gen^E(1^n)$

$k^M \leftarrow Gen^M(1^n)$

Output $k = (k^E, k^M)$

Enc$_k$(m):

$c^E \leftarrow Enc^E(k^E, m)$

$t \leftarrow Mac^M(k^M, c^E)$

output $c = (c^E, t)$

Dec$_k$(c): $c = (c^E, t)$

$m := Dec^E(k^E, c^E)$

$b := Vrfy^M(k^M, (c^E, t))$

If $b=1$, output m

Otherwise output $\perp$

Q1: Is it CPA-secure?

Q2: Is it CCA-secure?

Q3: Is it unforgeable?