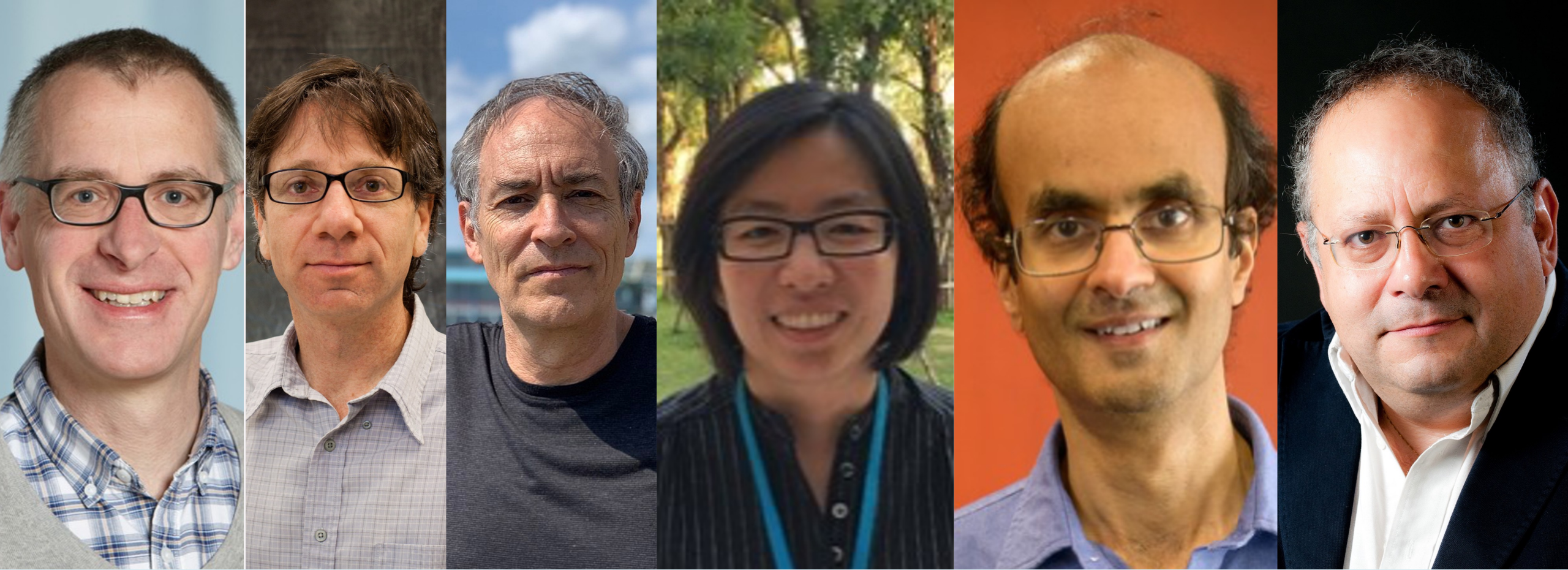


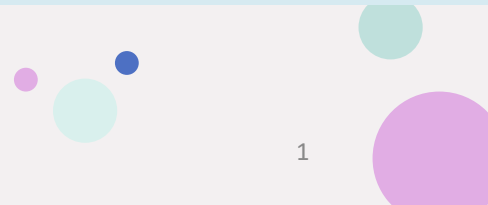


Lecture 10

Authenticated Encryption, Domain Extension of MACs

CS409/6011: Introduction to Cryptography

Sruthi Sekar



Recap

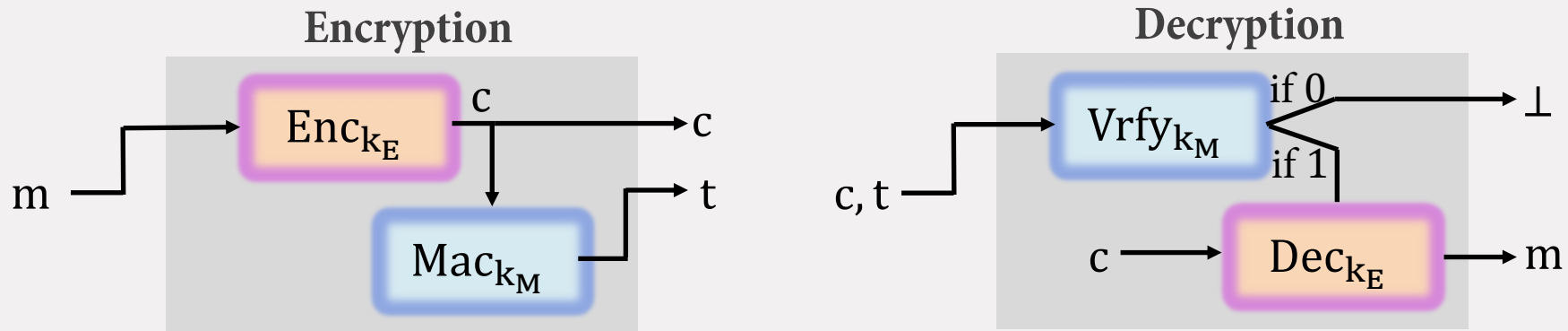
- **(Chosen Ciphertext Attack) CCA- Security:** **Mallory** gets both encryption and **decryption oracle** and still cannot distinguish between the (unqueried) ciphertexts corresponding to two messages of her choice.
- **Message Authentication Codes (MACs):** New building block that helps in authentication: EU-CMA security guarantees that a **Mallory** with access to a signing oracle cannot come up with a forgery on a unqueried message.
- **Authenticated Encryption: CCA Secure + Ciphertext Integrity**
 1. **CCA Security** allows **Mallory** to query the decryption oracle on a ciphertext related to the challenge ciphertext to get a message in response! The security guarantees that still she doesn't win.
 2. **Ciphertext integrity** says that **Mallory** cannot come up with valid ciphertexts without querying the encryption oracle.

RECALL

Attempt III: Encrypt then Authenticate

$\Pi_E = (\text{Gen}, \text{Enc}, \text{Dec})$
CPA Secure Encryption

$\Pi_M = (\text{Gen}, \text{Mac}, \text{Vrfy})$
EU-CMA secure MAC



k_E, k_M are independently generated

Is this secure?



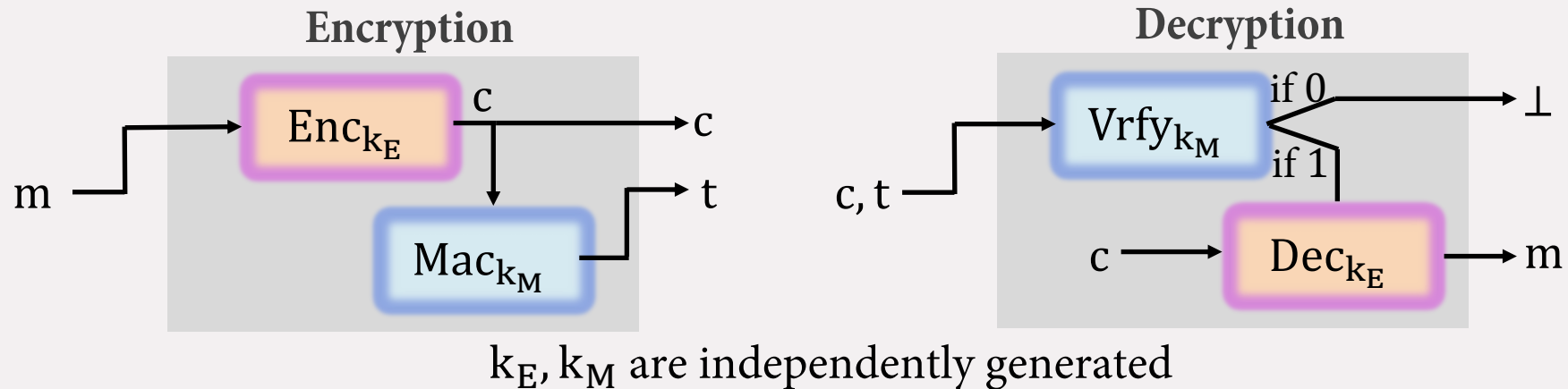
Not with the CMA security of MAC!
(Mallory can generate a new valid tag on an already queried c)

we need a stronger MAC security to prevent this!

Attempt III: Encrypt then Authenticate

$\Pi_E = (\text{Gen}, \text{Enc}, \text{Dec})$
CPA Secure Encryption

$\Pi_M = (\text{Gen}, \text{Mac}, \text{Vrfy})$
EU-SCMA secure MAC



Used in IPSec! Is this secure? ✓

Secure if you use CPA with
strong-CMA secure MAC

Today



Strong CMA Security for
MACs

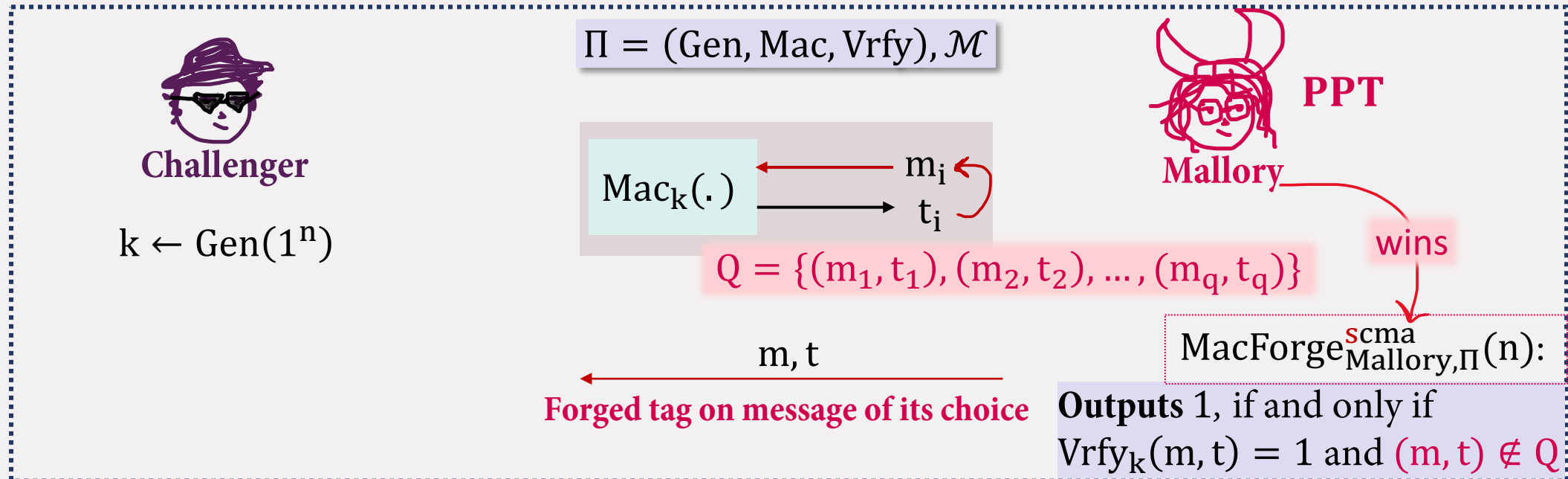


Constructing an
Authenticated Encryption



Domain Extension
of MACs

EU-SCMA Security for MAC



Definition 1

Π is **existential unforgeable under a strong chosen-message attack (EU-SCMA)**, if, for all PPT adversary **Mallory**, there exists a negligible function negl such that, $\forall n$

$$\Pr[\text{MacForge}_{\text{Mallory}, \Pi}^{\text{scma}}(n) = 1] \leq \text{negl}(n)$$

An EU-SCMA Secure MAC

- EU-CMA allows Mallory to forge a different tag t on some $m \in Q$, which EU-SCMA does not allow.
- Every EU-CMA MAC (**Gen, Mac, Vrfy**) with **canonical verification** (meaning that Vrfy_k just runs the Mac_k and checks if the tag output is the same as the tag it received) is **EU-SCMA secure**.
- Every **deterministic MAC** has canonical verification.

Example: The **PRF-based MAC** below (which we saw last time) is deterministic and EU-CMA secure $\Rightarrow$ it is EU-SCMA secure!

$\text{Gen}(1^n): k \in_R \{0,1\}^n$ $\text{Mac}(k, m): \text{output } t := F_k(m)$ $\text{Vrfy}(k, m, t): \text{output } 1, \text{ if } t := F_k(m), \text{ and } 0, \text{ o. w.}$

[[Lecture 9, Construction 2](#)] ($\mathcal{M} = \{0,1\}^n$, F is a PRF)

Today



Strong CMA Security for
MACs



Constructing an
Authenticated Encryption



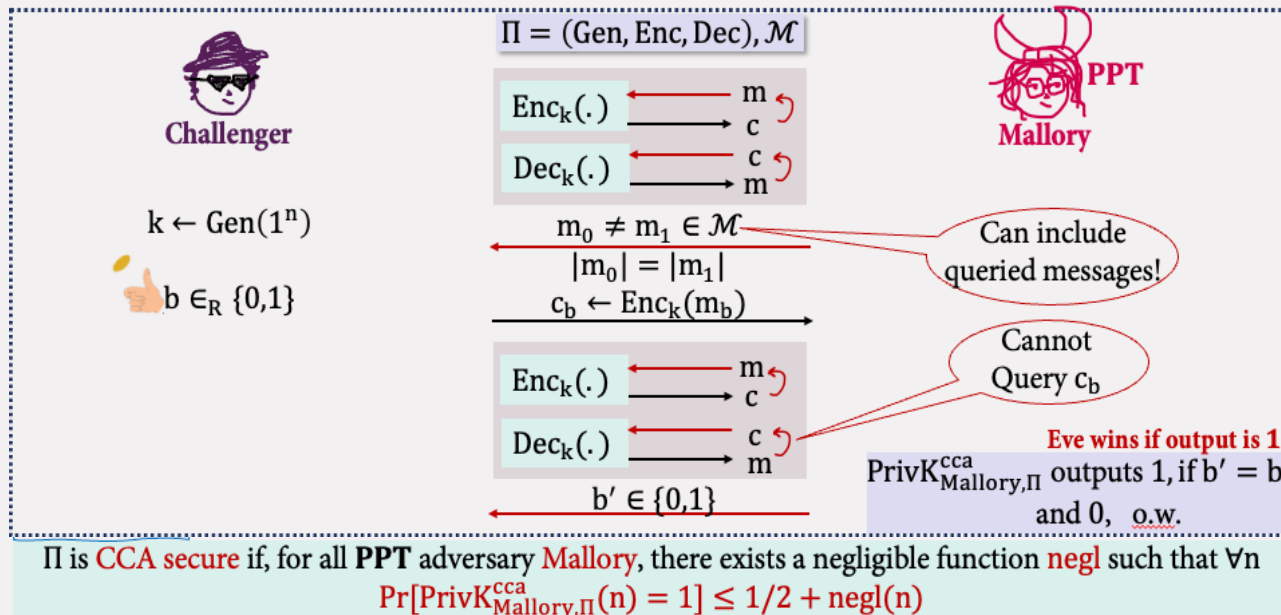
Domain Extension
of MACs

RECALL

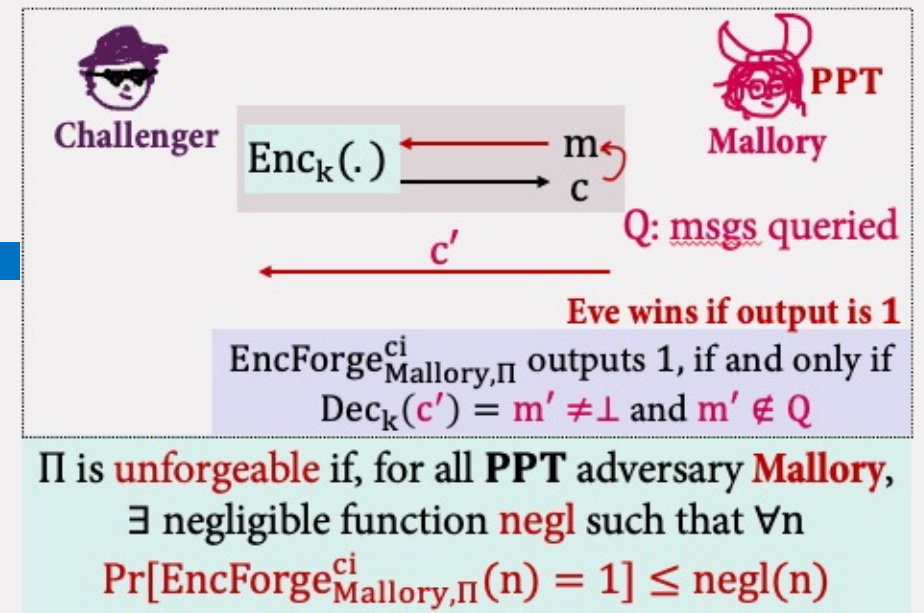
Authenticated Encryption (AE)

$$\Pi = (\text{Gen}, \text{Enc}, \text{Dec}), \mathcal{M}$$

SECRECY



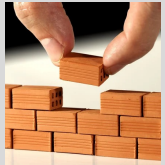
CIPHERTEXT INTEGRITY



Definition 2

$\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ is an **authenticated encryption scheme** for $\mathcal{M}$ if it is **CCA secure** and has **ciphertext integrity**!

AE: Encrypt then Authenticate

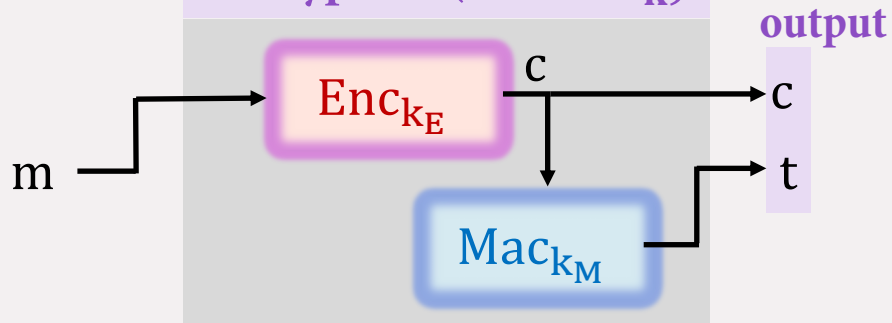


$\Pi_E = (\text{Gen}_E, \text{Enc}, \text{Dec})$
CPA Secure Encryption

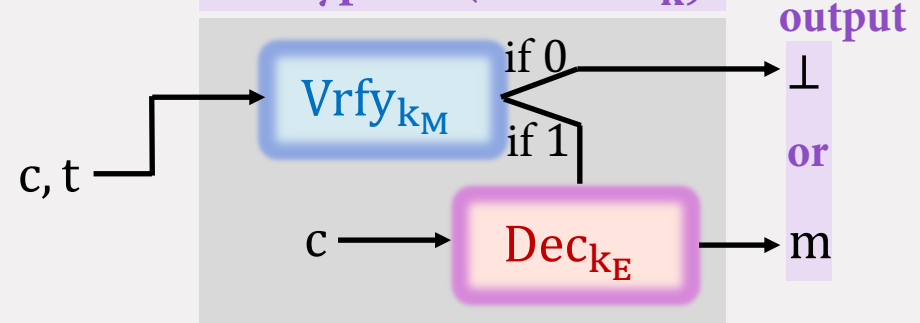
$\Pi_M = (\text{Gen}_M, \text{Mac}, \text{Vrfy})$
EU-SCMA secure MAC

(Key Generation) $\text{AE. Gen}(1^n)$: Pick $k_E \leftarrow \text{Gen}_E(1^n)$ and $k_M \leftarrow \text{Gen}_M(1^n)$ output $k = (k_E, k_M)$

Encryption (AE. Enc_k)



Decryption (AE. Dec_k)



Construction 1: $\Pi_{AE} = (\text{AE. Gen}, \text{AE. Enc}, \text{AE. Dec})$

Theorem 1

If $\Pi_E = (\text{Gen}_E, \text{Enc}, \text{Dec})$ is a CPA secure encryption and $\Pi_M = (\text{Gen}_M, \text{Mac}, \text{Vrfy})$ is an EU-SCMA secure MAC, then Π_{AE} is an authenticated encryption scheme.

AE: Encrypt then Authenticate

Theorem 1

If $\Pi_E = (\text{Gen}_E, \text{Enc}, \text{Dec})$ is a CPA secure encryption and $\Pi_M = (\text{Gen}_M, \text{Mac}, \text{Vrfy})$ is an EU-SCMA secure MAC, then Π_{AE} is an authenticated encryption scheme.

Proof Sketch:

STEP I

EU-SCMA security of Π_M

$\Rightarrow$

Any new decryption query by Mallory (something Mallory didn't get from encryption query) will be invalid!

Ciphertext Integrity

STEP II

Ciphertext Integrity*

$\Rightarrow$

CCA security essentially reduces to CPA security of Π_E , as the decryption oracle either outputs $\perp$ or uses the Enc queries

CCA Security

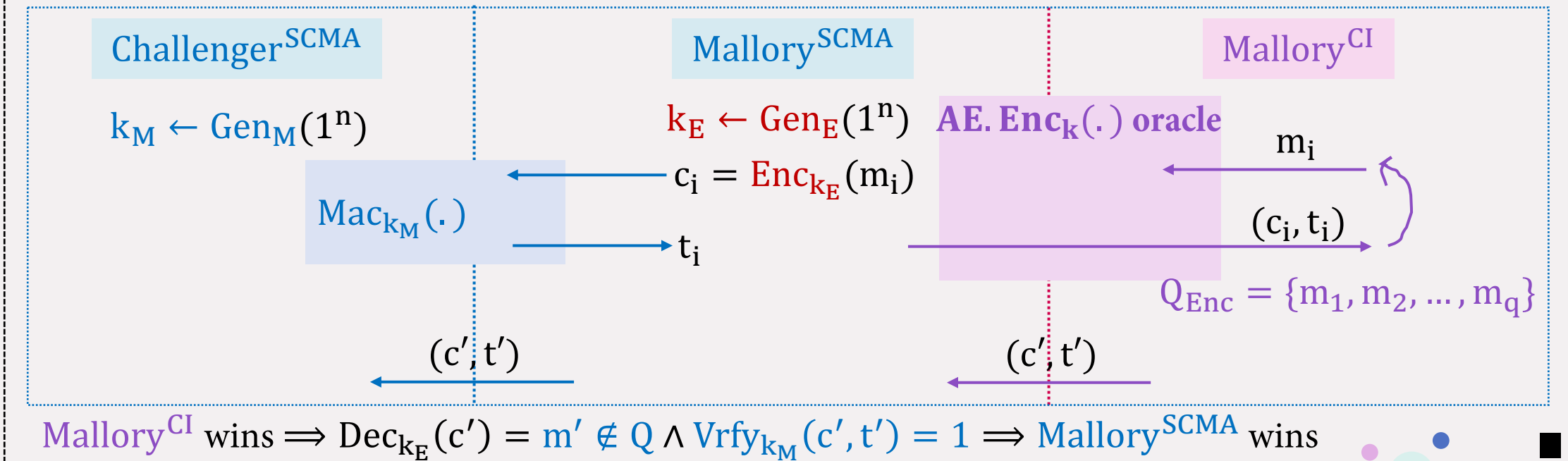
AE: Encrypt then Authenticate

Theorem 1A

If $\Pi_M = (\text{Gen}_M, \text{Mac}, \text{Vrfy})$ is an EU-SCMA secure MAC, then Π_{AE} has ciphertext integrity.

Ciphertext Integrity Proof: Assume $\exists \text{Mallory}^{CI}$ that breaks ciphertext integrity of Π_{AE}

To show: $\exists \text{Mallory}^{SCMA}$ breaking EU-SCMA security of Π_M



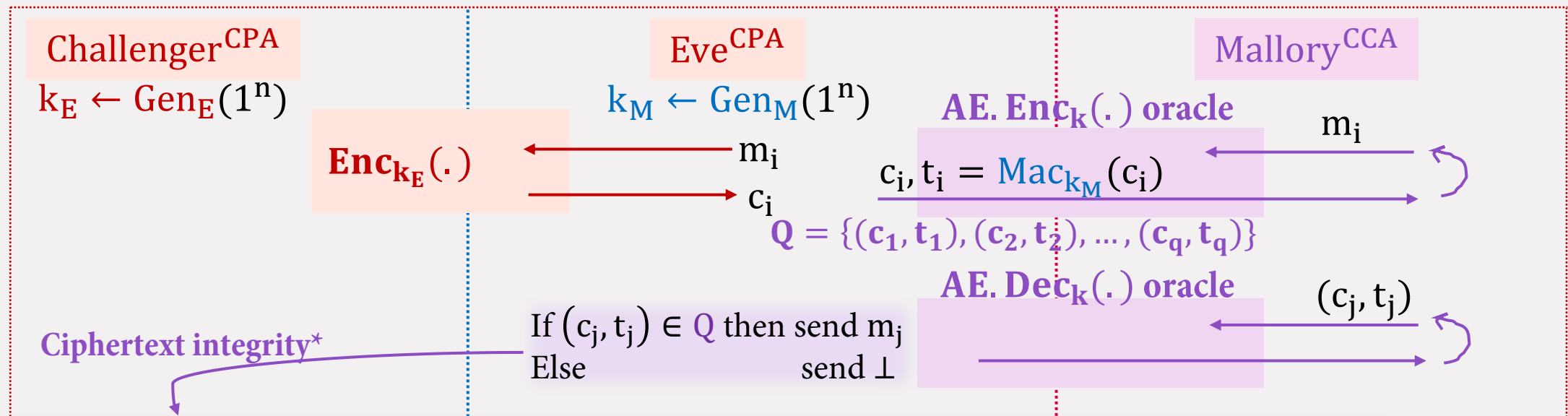
AE: Encrypt then Authenticate

Theorem 1

If $\Pi_E = (\text{Gen}_E, \text{Enc}, \text{Dec})$ is a CPA secure encryption and $\Pi_M = (\text{Gen}_M, \text{Mac}, \text{Vrfy})$ is an EU-SCMA secure MAC, then Π_{AE} is an authenticated encryption scheme.

CCA Security Proof: Assume $\exists \text{Mallory}^{\text{CCA}}$ that breaks ciphertext integrity of Π_{AE}

To show: $\exists \text{Eve}^{\text{CPA}}$ breaking CPA security of Π_E

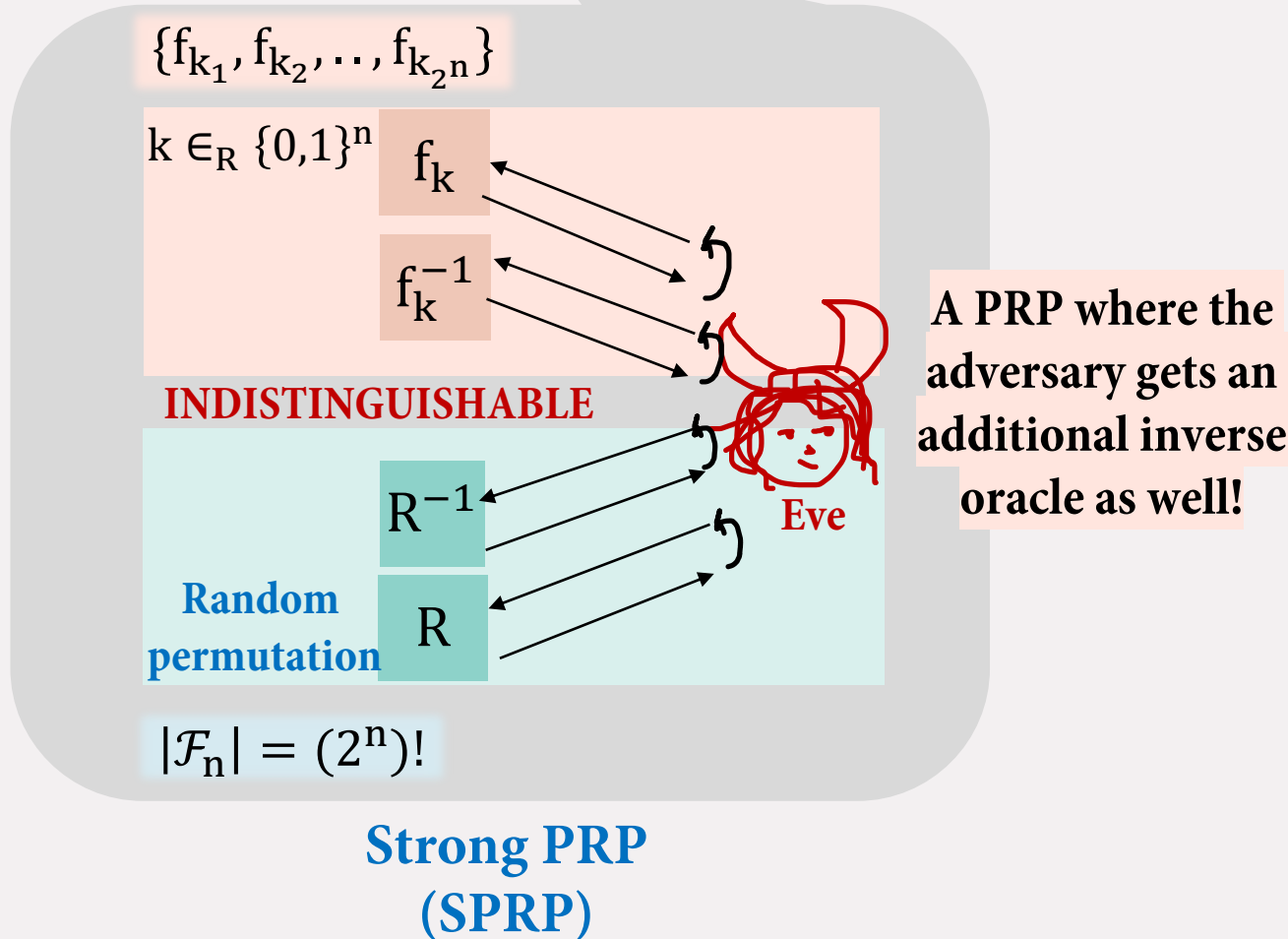


Argue that EU-SCMA security of $\Pi_M \Rightarrow \Pr[(c_j, t_j) \notin Q \wedge \text{Vrfy}_{k_M}(c_j, t_j) = 1] \leq \text{negl}(n)$

Discuss rest of the reduction on board

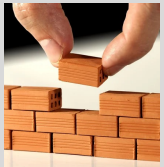
Need for independent k_E and k_M

F be a strong-PRP



Need for independent k_E and k_M

F be a strong-PRP $\Rightarrow F^{-1}$ is also a strong-PRP



$\Pi_E = (\text{Gen}_E, \text{Enc}, \text{Dec})$
 $k \leftarrow \text{Gen}_E$
 $\text{Enc}_k(m) := F_k(m||r), \text{ for } r \in_R \{0,1\}^{n/2}$

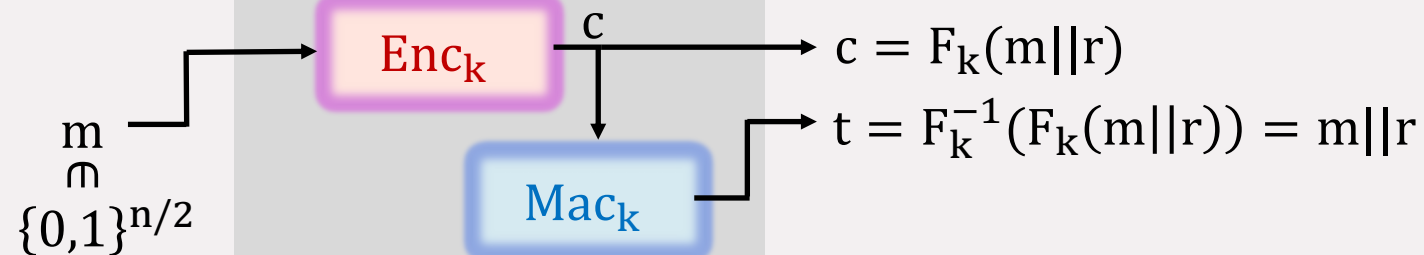
$\Pi_M = (\text{Gen}_M, \text{Mac}, \text{Vrfy})$
 $k \leftarrow \text{Gen}_M$
 $\text{Mac}_k(m) = F_k^{-1}(m)$

(Key Generation) AE. $\text{Gen}(1^n)$: $k \in_R \{0,1\}^n$ Use same key!

(Hard) HW1:

In fact, Π_E is **CCA secure**,
(but it does not satisfy
ciphertext integrity).

Encryption (AE. Enc_k)



**It fits the encrypt-then-authenticate paradigm, uses a CPA secure encryption and EU-SCMA MAC,
but is clearly an insecure authenticated encryption scheme!**

Today



Strong CMA Security for
MACs



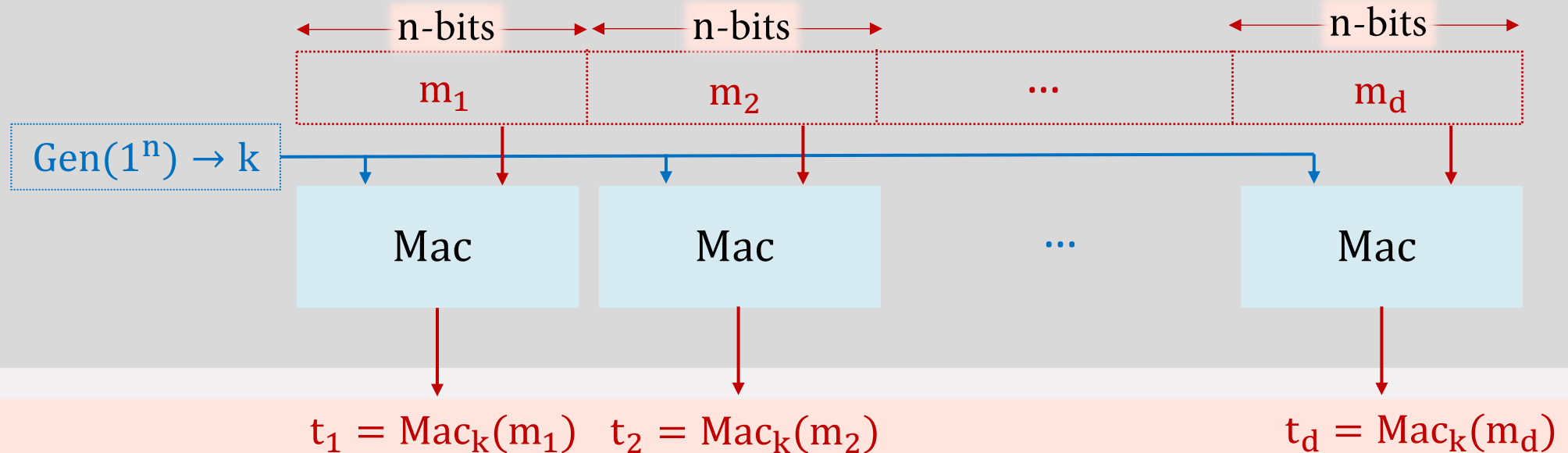
Constructing an
Authenticated Encryption



Domain Extension
of MACs

Domain Extension: Attempt I

$\Pi = (\text{Gen}, \text{Mac}, \text{Vrfy})$, $\mathcal{M} = \{0,1\}^n$ be an EU-CMA secure MAC.



SECURE OR NOT?

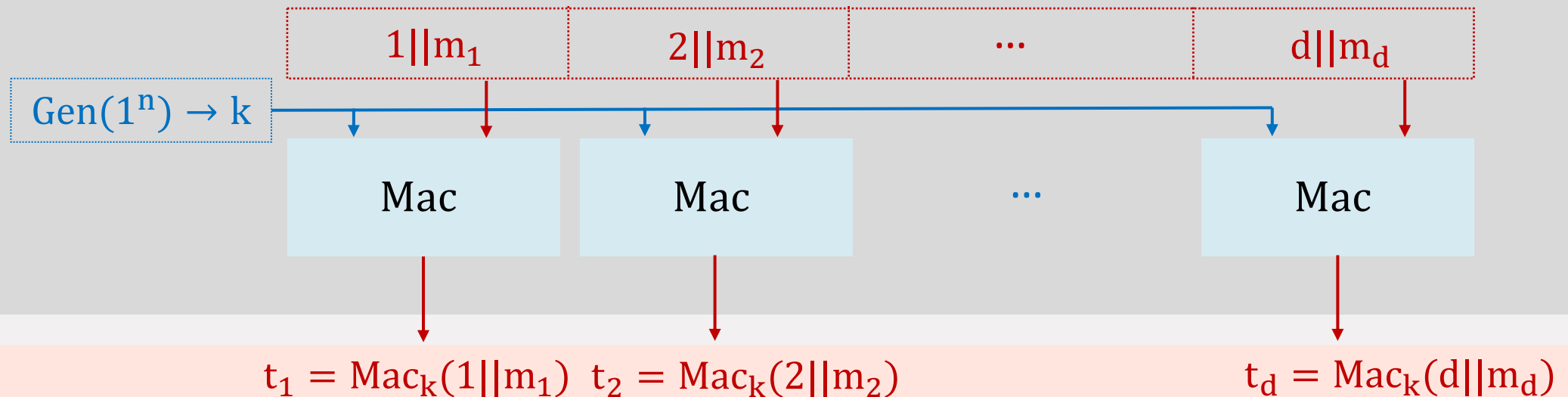
Block Reordering Attack

Get tag (t_1, t_2) on (m_1, m_2) from signing oracle

Forgery: $((m_2, m_1), (t_2, t_1))$

Domain Extension: Attempt II

$\Pi = (\text{Gen}, \text{Mac}, \text{Vrfy})$, $\mathcal{M} = \{0,1\}^n$ be an EU-CMA secure MAC.



SECURE OR NOT?

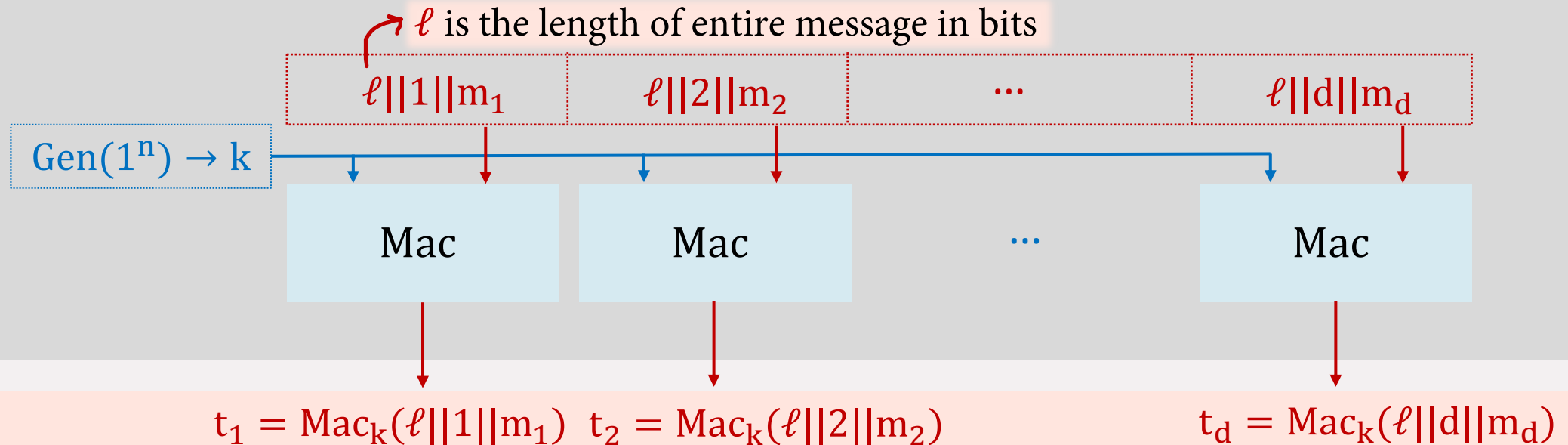
Truncation Attack

Get tag (t_1, t_2) on $(1||m_1, 2||m_2)$ from signing oracle

Forgery: $(1||m_1, t_1)$

Domain Extension: Attempt III

$\Pi = (\text{Gen}, \text{Mac}, \text{Vrfy})$, $\mathcal{M} = \{0,1\}^n$ be an EU-CMA secure MAC.



SECURE OR NOT?

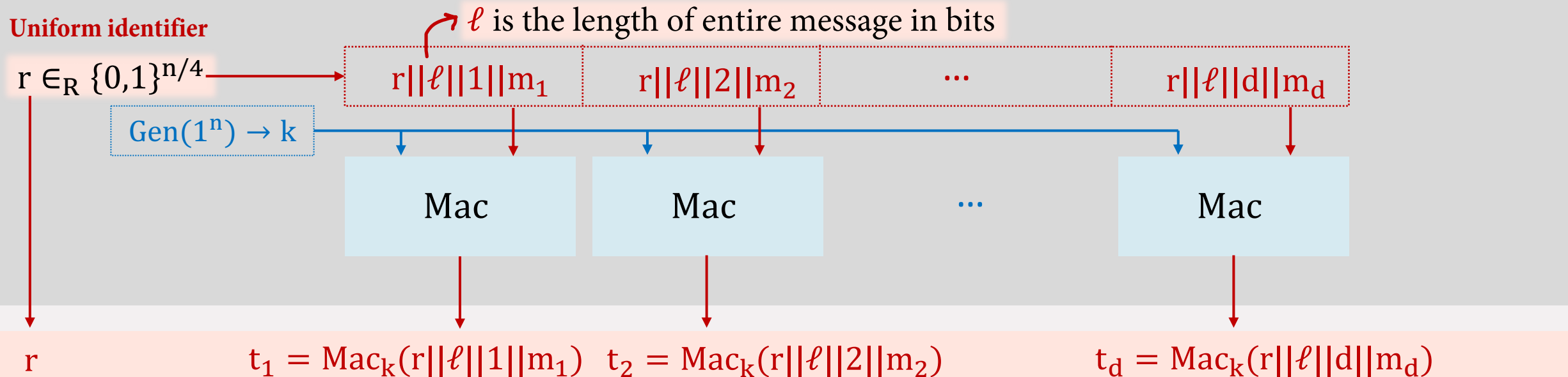
Mix and match Attack

Get tags (t_1, t_2) on $(\ell||1||m_1, \ell||2||m_2)$ and (t'_1, t'_2) on $(\ell||1||m'_1, \ell||2||m'_2)$ from signing oracle
Forgery: $(\ell||1||m_1, \ell||2||m'_2), (t_1, t'_2)$

Domain Extension of MAC

(Theoretical Construction)

$\Pi = (\text{Gen}, \text{Mac}, \text{Vrfy})$, $\mathcal{M} = \{0,1\}^n$ be an EU-CMA secure MAC.



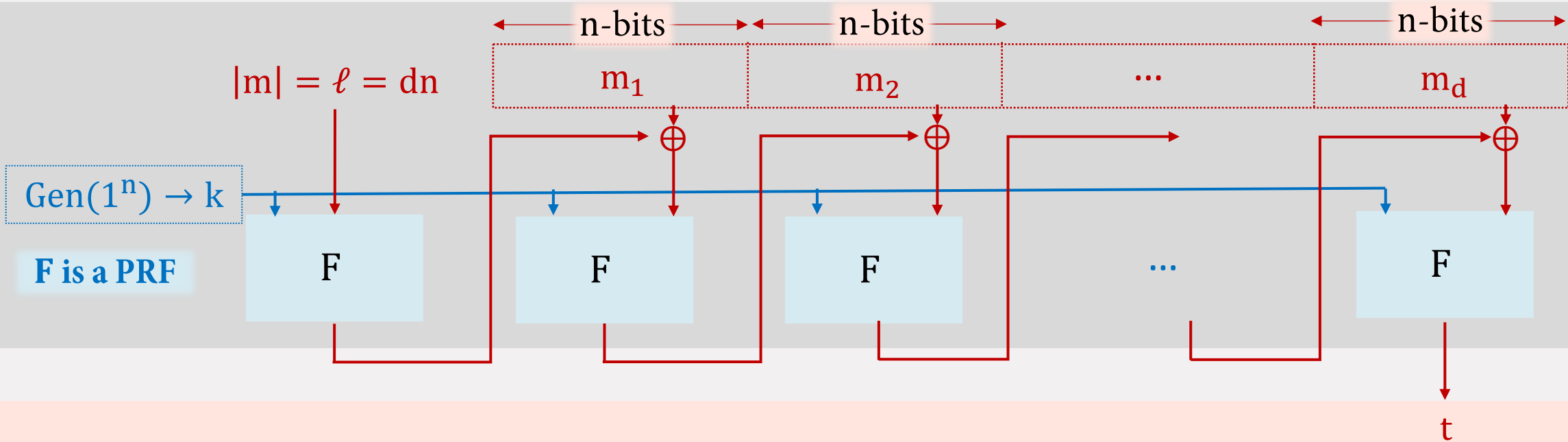
Construction 2

$|msg| = dn/4\text{-bits}; \quad |tag| = dn\text{-bits}$

Theorem 2

If $\Pi = (\text{Gen}, \text{Mac}, \text{Vrfy})$ is an EU-CMA secure MAC, then **Construction 2** is an EU-CMA secure MAC.

CBC-MAC



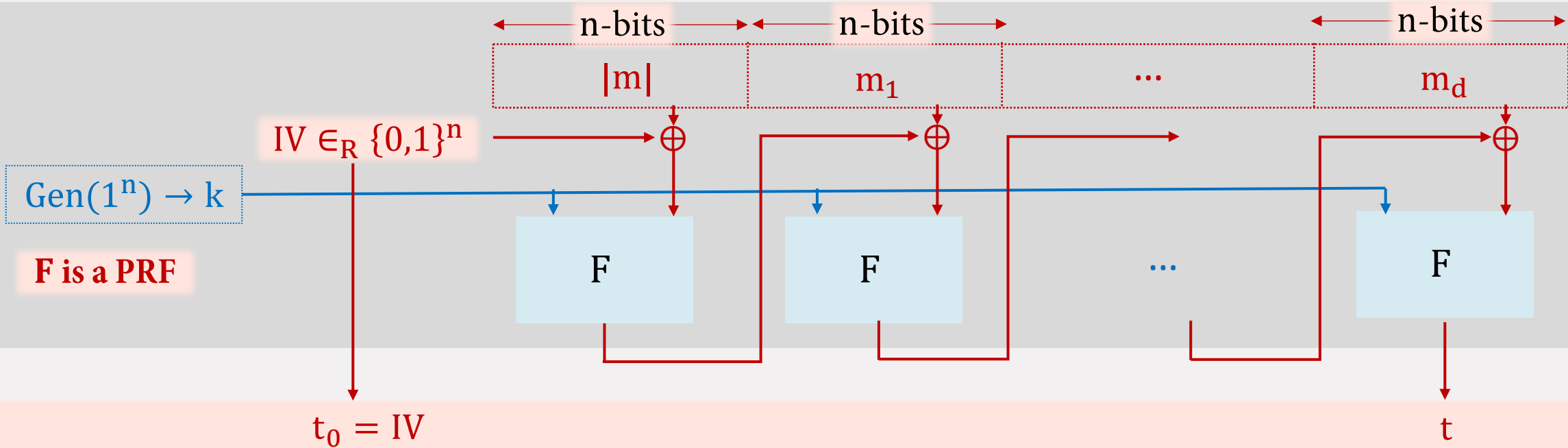
Construction 3

$|msg| = dn\text{-bits}; \quad |tag| = n\text{-bits}$

SECURE OR NOT?

Yes! If F is a PRF then, **Construction 3** is an EU-CMA secure MAC for messages of arbitrary length!

CBC-MAC Variants



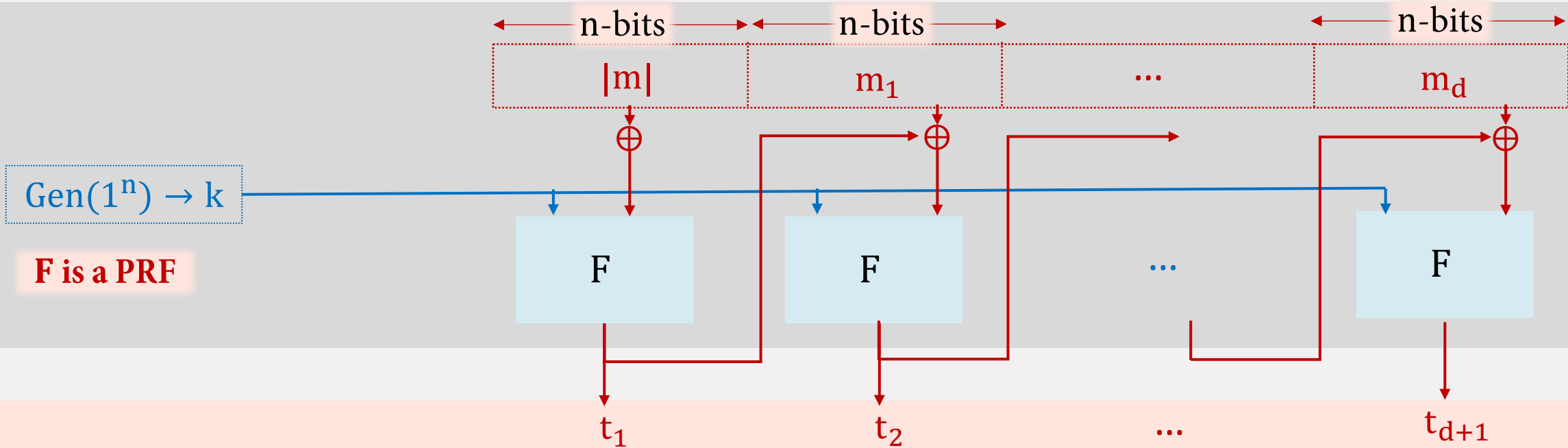
CBC MAC vs CBC-mode Encryption

What if I use a **random IV** (instead of $IV = 0^n$) like CBC-mode encryption?

HW 2

Insecure (show an EU-CMA attack)!

CBC-MAC Variants



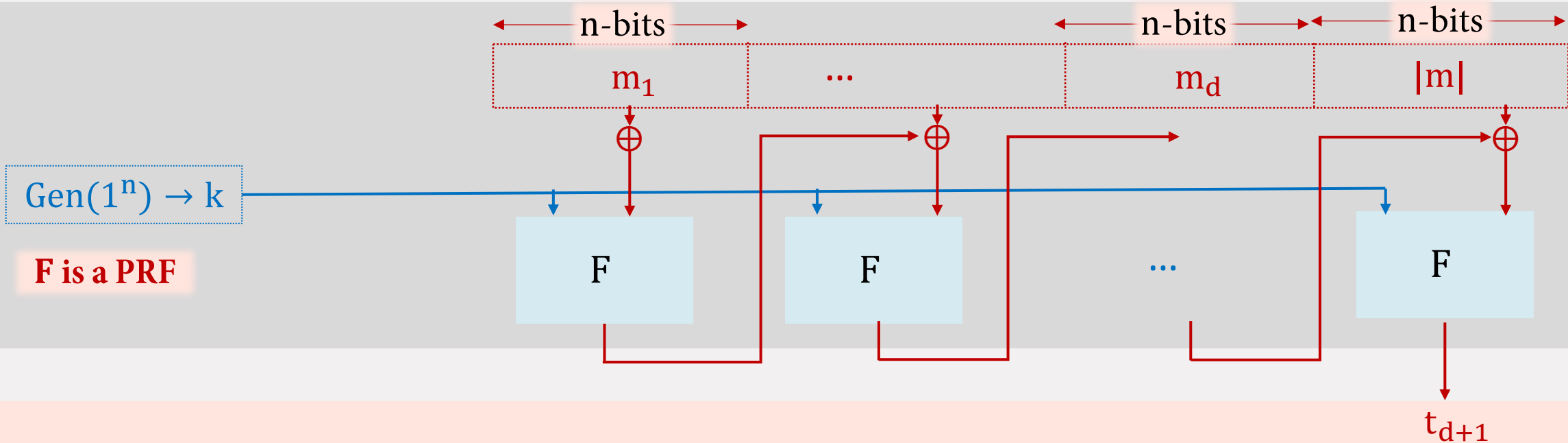
CBC MAC vs CBC-mode Encryption

What if I output all intermediate outputs of F as in CBC encryption?

HW 3

Insecure (show an EU-CMA attack)!

CBC-MAC Variants

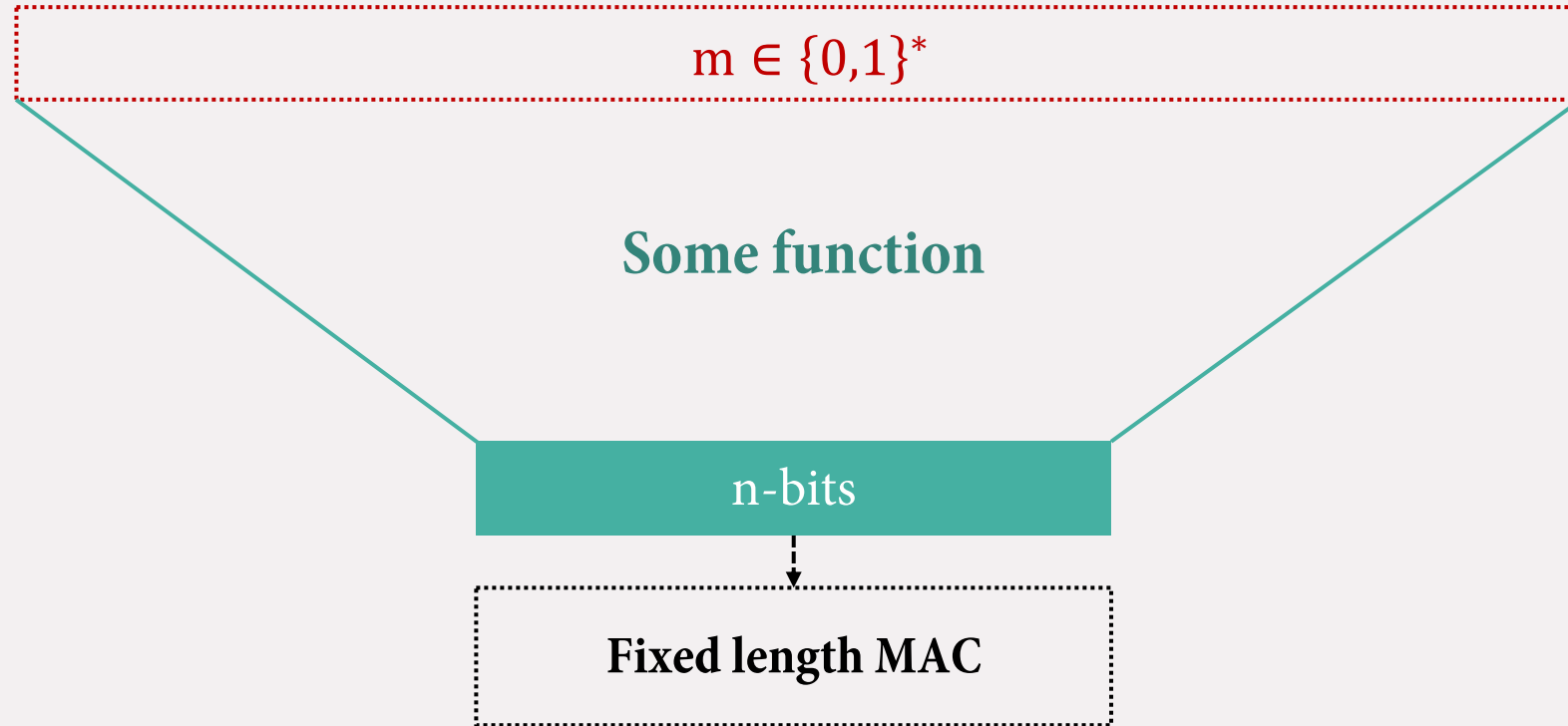


What if I append $|m|$ to the end instead of the beginning?

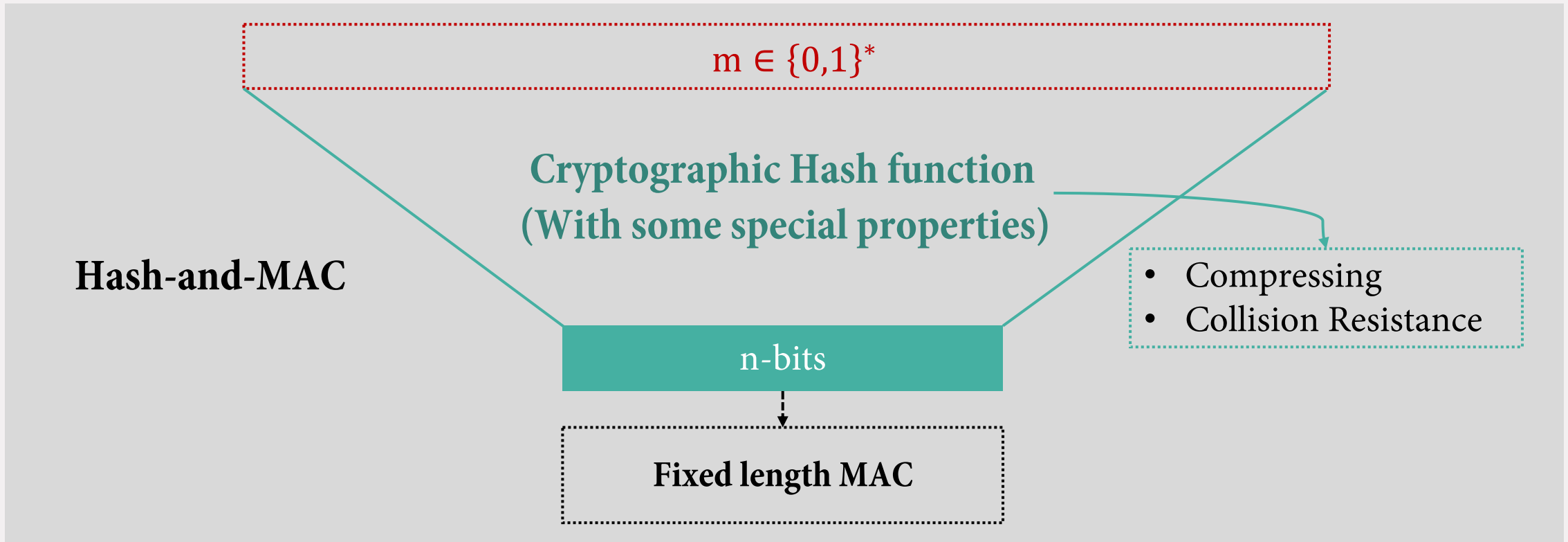
HW 4

Insecure (show an EU-CMA attack)!

Other Domain Extension Methods



Other Domain Extension Methods



HMAC

In fact, we can use this new primitive to build a MAC for messages of arbitrary length without using a PRF!



THANK YOU