Lecture 4 – Message authentication, UF-CMA, CBC-MAC, CMAC

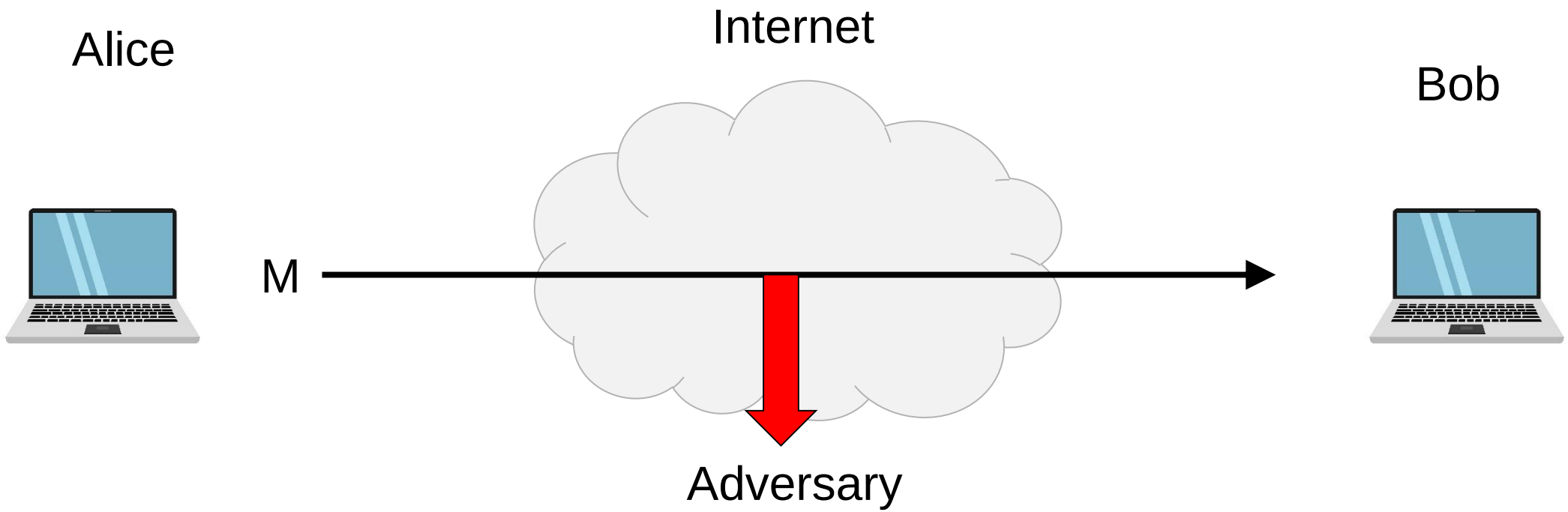
TEK4500

15.09.2020

Håkon Jacobsen

hakon.jacobsen@its.uio.no

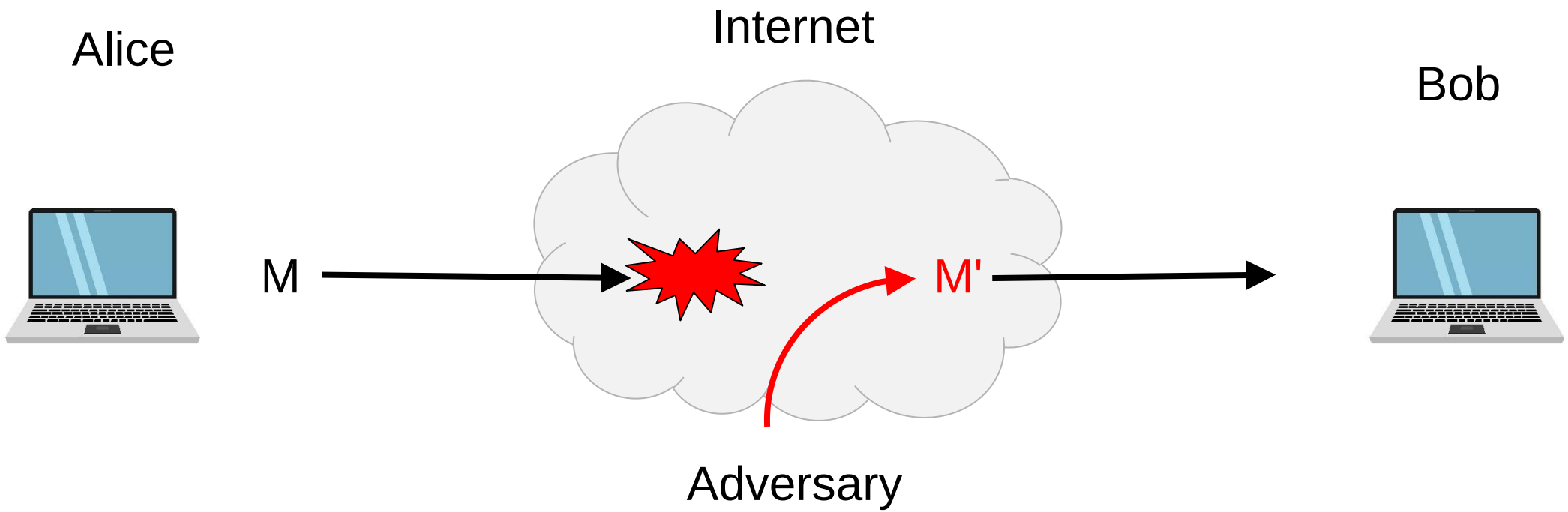
What is cryptography?



Security goals:

- **Data privacy:** adversary should not be able to read message M

What is cryptography?



Security goals:

- **Data privacy:** adversary should not be able to read message M
- **Data integrity:** adversary should not be able to modify message M
- **Data authenticity:** message M really originated from Alice

Basic goals of cryptography

	Message privacy	**Message integrity / authentication**
Symmetric keys	Symmetric encryption	Message authentication codes (MAC)
Asymmetric keys	Asymmetric encryption (a.k.a. public-key encryption)	Digital signatures

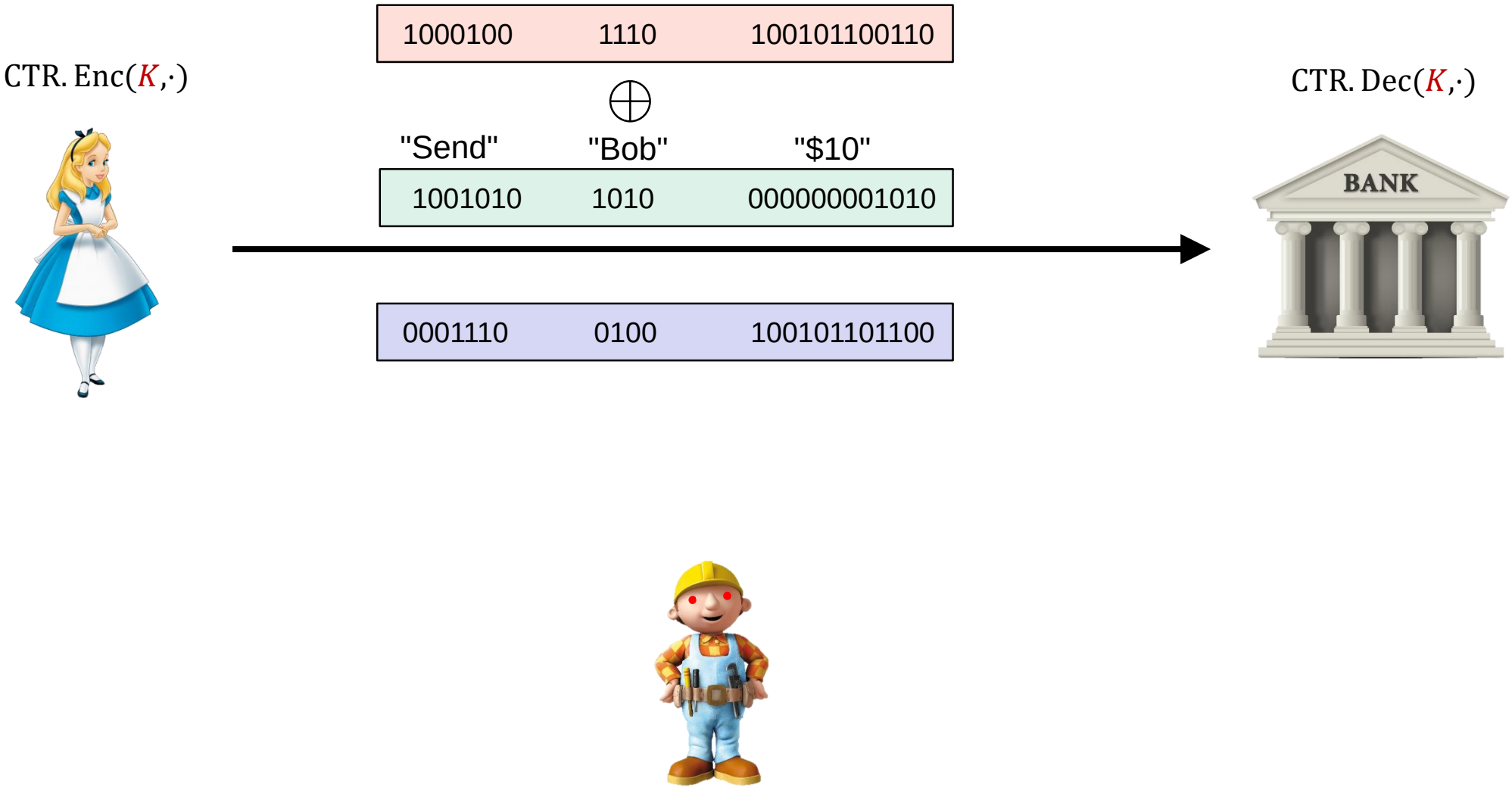
Motivation

- Goal: *integrity*, but not privacy
- Examples:
  - Protecting OS system files against tampering
  - Browser cookies stored by web servers
  - Control signals in network management

Encryption ≠ integrity

"Send Bob $10"

Encryption ≠ integrity



Encryption ≠ integrity

$\mathrm{CTR.Enc}(K,\cdot)$

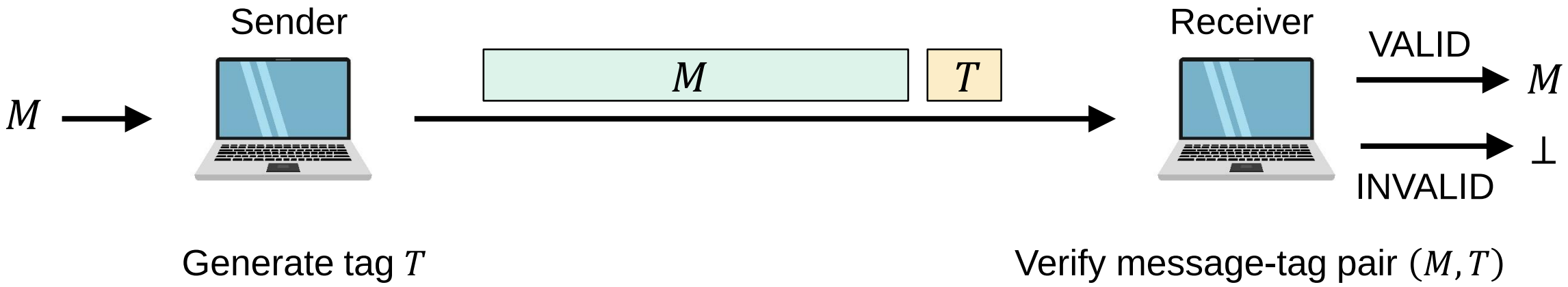
1000100	1110	100101100110
⊕		
"Send"	"Bob"	"$10"
1001010	1010	000000001010

~~0001110~~	~~0100~~	~~100101101100~~
0001110	0100	**0110**01101100

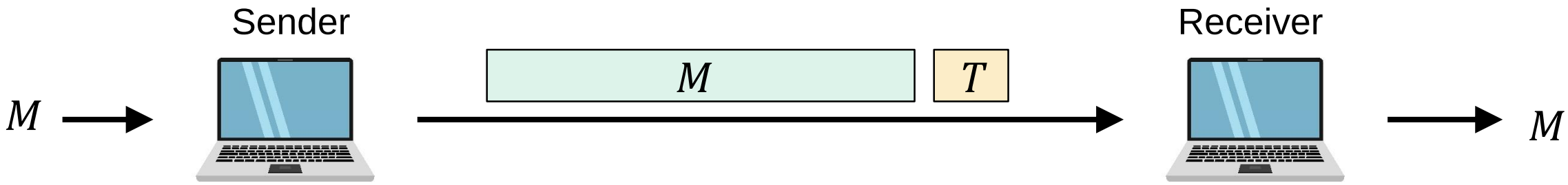
$\mathrm{CTR.Dec}(K,\cdot)$

1001010	1010	**111**00001010
"Send"	"Bob"	"**$3850**"

Message authentication – idea



Authentication from error-checking codes

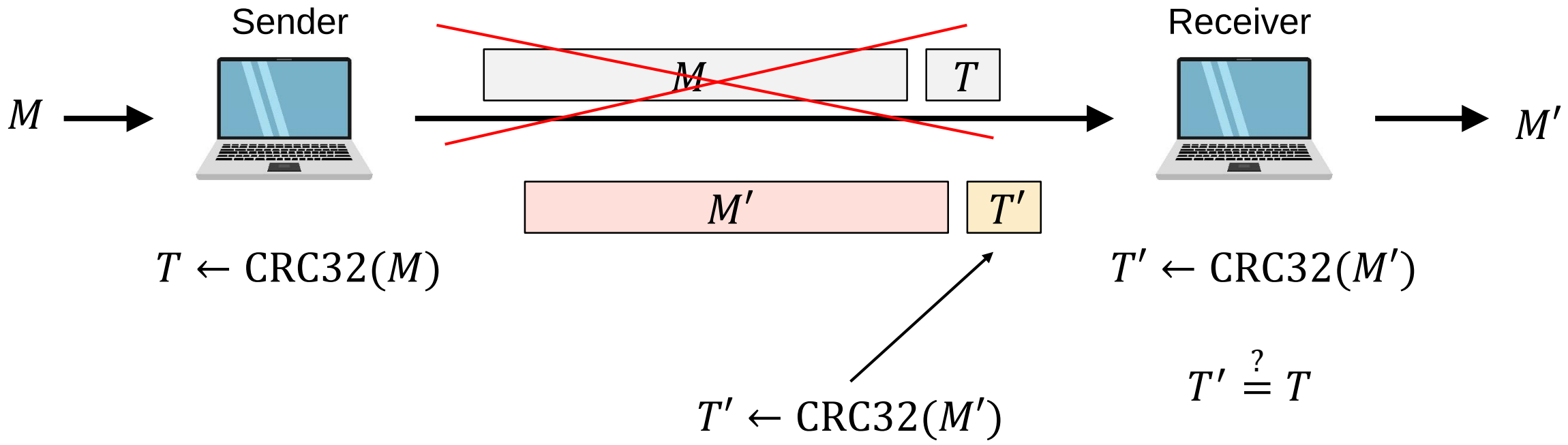


$T \leftarrow \mathrm{CRC32}(M)$

$T' \leftarrow \mathrm{CRC32}(M)$

$T' \stackrel{?}{=} T$

Keyless message integrity doesn't work

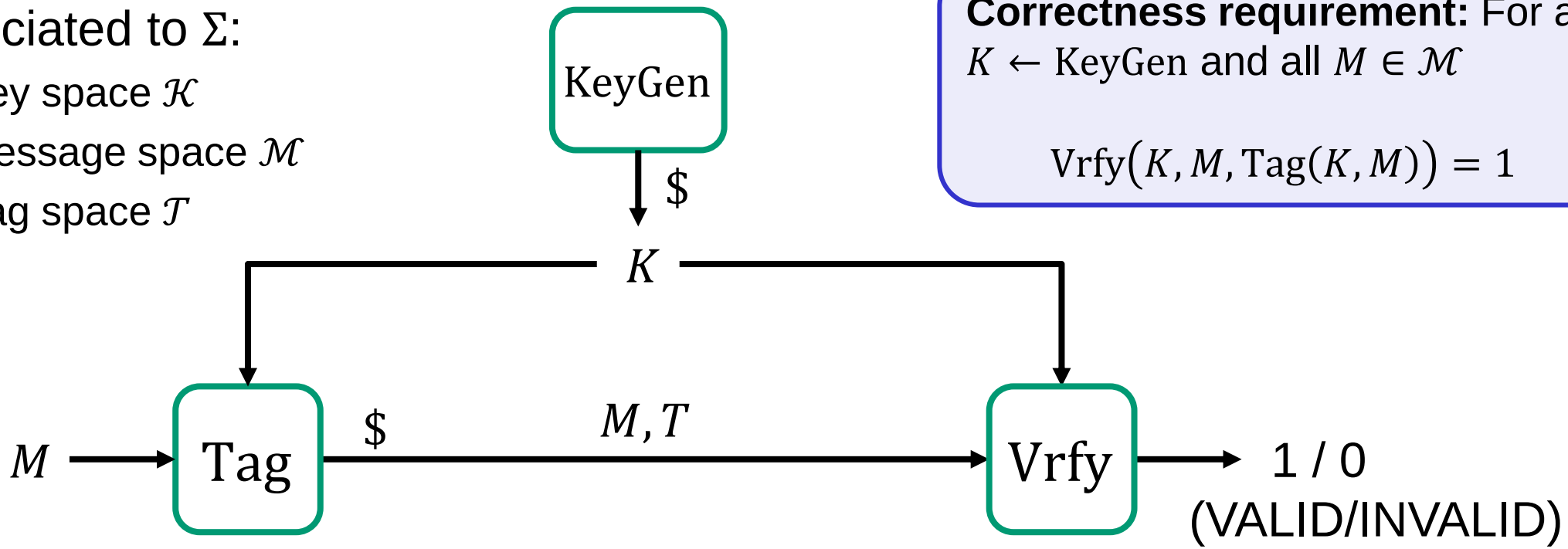


Message authentication schemes – syntax

- A **message authentication scheme** $\Sigma = (\mathrm{KeyGen}, \mathrm{Tag}, \mathrm{Vrfy})$ consists of three algorithms:
- Associated to Σ:
  - Key space $\mathcal{K}$
  - Message space $\mathcal{M}$
  - Tag space $\mathcal{T}$

Correctness requirement: For all $K \leftarrow \mathrm{KeyGen}$ and all $M \in \mathcal{M}$

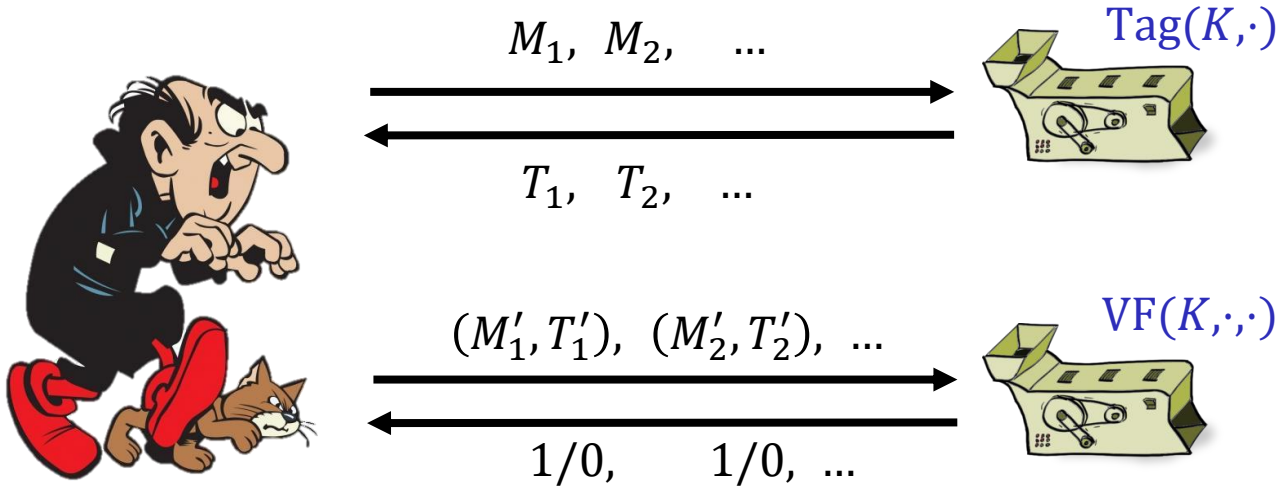
$$\mathrm{Vrfy}\big(K, M, \mathrm{Tag}(K, M)\big) = 1$$



- KeyGen and Tag may be randomized ($), but Vrfy must be deterministic

UF-CMA – Unforgability against chosen-message attacks

Challenger



Adversary *wins* if a pair (M_i', T_i') is valid,
and was not among the pairs $(M_1, T_1), (M_2, T_2), \ldots$

UF-CMA – Unforgability against chosen-message attacks

$\mathbf{Exp}_{\Sigma}^{\text{uf-cma}}(A)$

1. $K \xleftarrow{\$} \Sigma.\text{KeyGen}$
2. $S \leftarrow [\,]$
3. $won \leftarrow 0$
4. $A^{\text{Tag}_K(\cdot),\, \text{VF}_K(\cdot,\cdot)}$
5. **return** won

$\text{Tag}_K(M)$

---

1. $T \leftarrow \Sigma.\text{Tag}(K, M)$
2. $S.\text{add}\big((M, T)\big)$
3. **return** T

$\text{VF}_K(M, T)$

---

1. $d \leftarrow \Sigma.\text{Vrfy}(K, M, T)$
2. **if** $d = 1$ and $(M, T) \notin S$ **then:**
3. $won \leftarrow 1$
4. **return** d

Challenger

$M_1,\ M_2,\ \ldots$ → $\text{Tag}(K,\cdot)$

$T_1,\ T_2,\ \ldots$ ←

$(M_1', T_1'),\ (M_2', T_2'),\ \ldots$ → $\text{VF}(K,\cdot,\cdot)$

$1/0,\quad 1/0,\ \ldots$ ←

Definition: The **UF-CMA-advantage** of an adversary A is

$$\mathbf{Adv}_{\Sigma}^{\text{uf-cma}}(A) = \Pr\left[\mathbf{Exp}_{\Sigma}^{\text{uf-cma}}(A) \Rightarrow 1\right]$$

UF-CMA – Unforgability against chosen-message attacks

$\mathbf{Exp}_{\Sigma}^{\text{uf-cma}}(A)$

1. $K \xleftarrow{\$} \Sigma.\text{KeyGen}$
2. $S \leftarrow [\,]$
3. $won \leftarrow 0$
4. $A_K^{\text{Tag}_K(\cdot),\,\text{VF}_K(\cdot,\cdot)}$
5. **return** won

$\text{Tag}_K(M)$

---

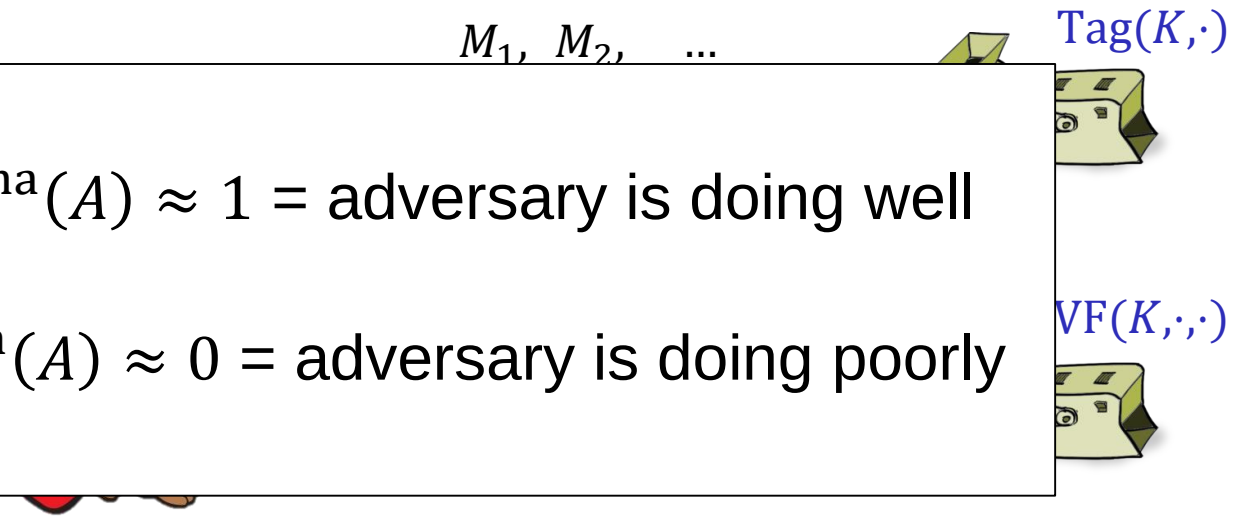
1. $T \leftarrow \Sigma.\text{Tag}(K, M)$
2. $S.\text{add}\big((M, T)\big)$
3. **return** T

$\text{VF}_K(M, T)$

---

1. $d \leftarrow \Sigma.\text{Vrfy}(K, M, T)$
2. **if** $d = 1$ and $(M, T) \notin S$ **then:**
3. $won \leftarrow 1$
4. **return** d

Challenger



$\mathbf{Adv}_{\Sigma}^{\text{uf-cma}}(A) \approx 1$ = adversary is doing well

$\mathbf{Adv}_{\Sigma}^{\text{uf-cma}}(A) \approx 0$ = adversary is doing poorly

Definition: The **UF-CMA-advantage** of an adversary A is

$$\mathbf{Adv}_{\Sigma}^{\text{uf-cma}}(A) = \Pr\left[\mathbf{Exp}_{\Sigma}^{\text{uf-cma}}(A) \Rightarrow 1\right]$$

Properties of UF-CMA definition

- UF-CMA security implies:
  - Hard to recover K
  - Hard to do *selective* forgery: forge a tag on a *specific* message chosen by the adversary
  - Hard to forge a new *tag* on an old message (and tag)
    - If each message has a *unique* tag $\Longrightarrow$ forgery must be on a new message
  - Tag lengths must be *long enough*!
    - Suppose $|T| = 10$; then adversary who simply guesses T has advantage $\mathbf{Adv}_{\Sigma}^{\text{uf-cma}}(A) = \frac{1}{2^{10}} \approx \frac{1}{1000}$

- Does *not* give protection against **replay attacks**
  - Message counters
  - Nonces
  - Timestamps

$\mathbf{Exp}_{\Sigma}^{\text{uf-cma}}(A)$

1. $K \xleftarrow{\$} \Sigma.\text{KeyGen}$
2. $S \leftarrow [\,];\ \ won \leftarrow 0$
3. $A_K^{\text{Tag}_K(\cdot),\,\text{VF}_K(\cdot,\cdot)}$
4. **return** won

$\text{Tag}_K(M)$

1. $T \leftarrow \Sigma.\text{Tag}(K, M)$
2. $S.\text{add}\big((M, T)\big)$
3. **return** T

$\text{VF}_K(M, T)$

1. $d \leftarrow \Sigma.\text{Vrfy}(K, M, T)$
2. **if** $d = 1$ and $(M, T) \notin S$ **then:**
3. $\quad won \leftarrow 1$
4. **return** d

$$\mathbf{Adv}_{\Sigma}^{\text{uf-cma}}(A) = \Pr\left[\mathbf{Exp}_{\Sigma}^{\text{uf-cma}}(A) \Rightarrow 1\right]$$

Message authentication *codes* (MACs)

- Special class of message authentication schemes:

$\Sigma.\mathrm{Tag} : \mathcal{K} \times \mathcal{M} \to \mathcal{T}$ is a *deterministic function*

$\Sigma.\mathrm{Vrfy} : \mathcal{K} \times \mathcal{M} \times \mathcal{T} \to \{0,1\}$ simply recomputes tag and compares:

$$\Sigma.\mathrm{Vrfy}(K, M, T) \stackrel{\text{def}}{=} \begin{cases} 1, & \text{if } T = \Sigma.\mathrm{Tag}(K, M) \\ 0, & \text{otherwise} \end{cases}$$

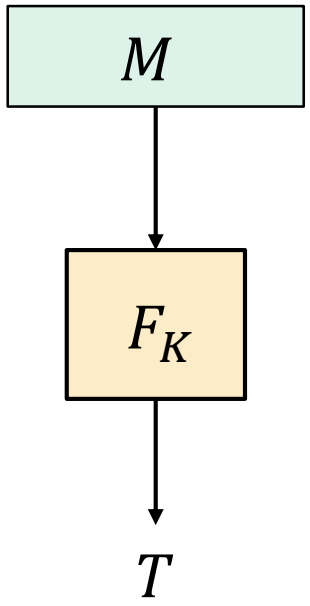
- Most common in practice

- MAC $\Longrightarrow$ unique tags $\Longrightarrow$ forgeries must be on new *messages*

PRFs are good MACs

$F : \{0,1\}^k \times \{0,1\}^{in} \to \{0,1\}^{out}$

PRF



Alg $\Sigma_{\text{PRF}}.\text{Tag}(K, M)$

---

1. **if** $M \notin \{0,1\}^{in}$ **then**
2. **return** $\perp$
3. **return** $F_K(M)$

Alg $\Sigma_{\text{PRF}}.\text{Vrfy}(K, M, T)$

---

1. $T' \leftarrow F_K(M)$
2. **return** $T' \stackrel{?}{=} T$

Theorem: If F is a secure PRF then Σ_{PRF} is UF-CMA secure for *fixed-length* messages $M \in \{0,1\}^{in}$

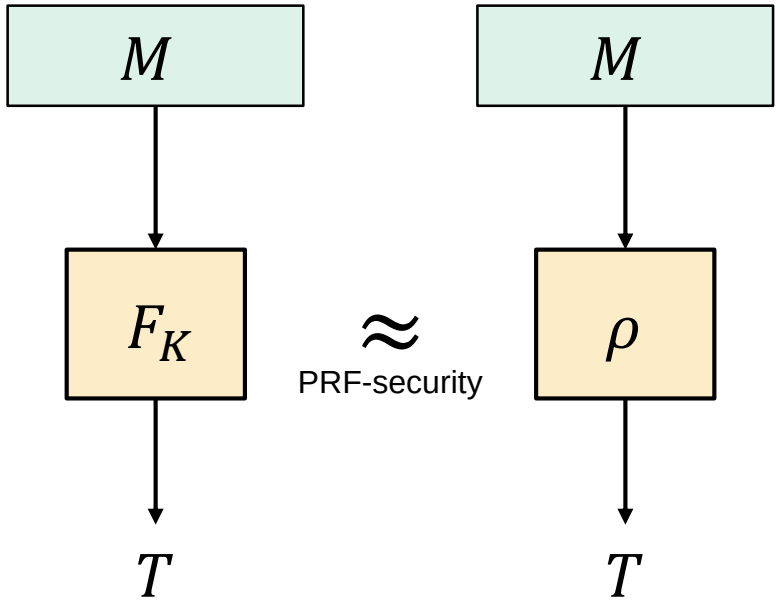
Detailed: for *any* UF-CMA adversary A against Σ_{PRF} asking v Vrfy queries, there *is* a PRF-adversary B against F such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\Sigma_{\text{PRF}}}(A) \leq \mathbf{Adv}^{\text{prf}}_{F}(B) + \frac{v}{2^{out}}$$

PRFs are good MACs – proof sketch

Theorem: For *any* UF-CMA adversy A, asking v Vrfy queries, there *is* a PRF-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\Sigma_{\text{PRF}}}(A) \leq \mathbf{Adv}^{\text{prf}}_{F}(B) + \frac{v}{2^{out}}$$

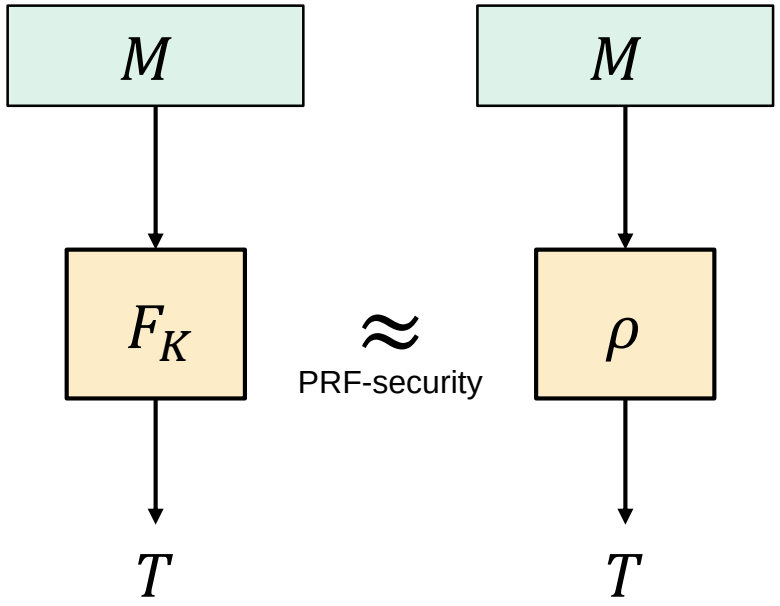


$$\rho \xleftarrow{\$} \text{Func}[in, out]$$

PRFs are good MACs – proof sketch

Theorem: For *any* UF-CMA adversy A, asking v Vrfy queries, there *is* a PRF-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\Sigma_{\text{PRF}}}(A) \leq \mathbf{Adv}^{\text{prf}}_{F}(B) + \frac{v}{2^{out}}$$

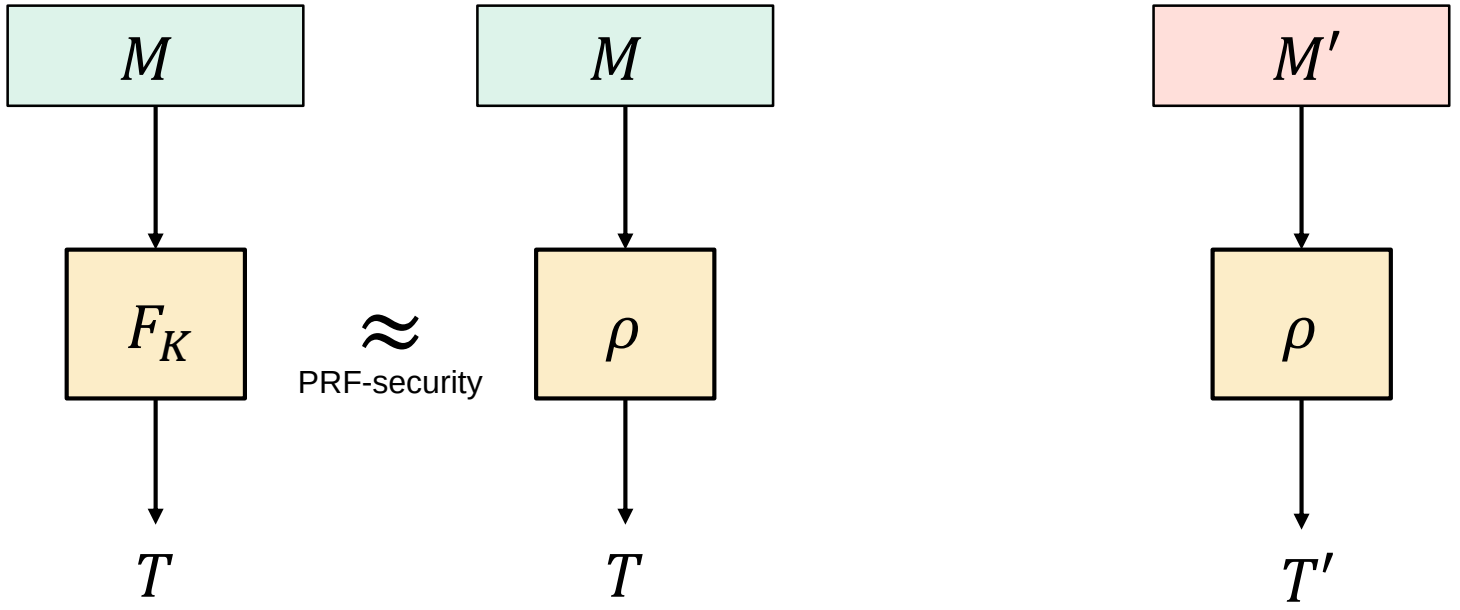


$$\rho \xleftarrow{\$} \text{Func}[in, out]$$

PRFs are good MACs – proof sketch

Theorem: For *any* UF-CMA adversy A, asking v Vrfy queries, there *is* a PRF-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\Sigma_{\text{PRF}}}(A) \leq \mathbf{Adv}^{\text{prf}}_{F}(B) + \frac{v}{2^{out}}$$

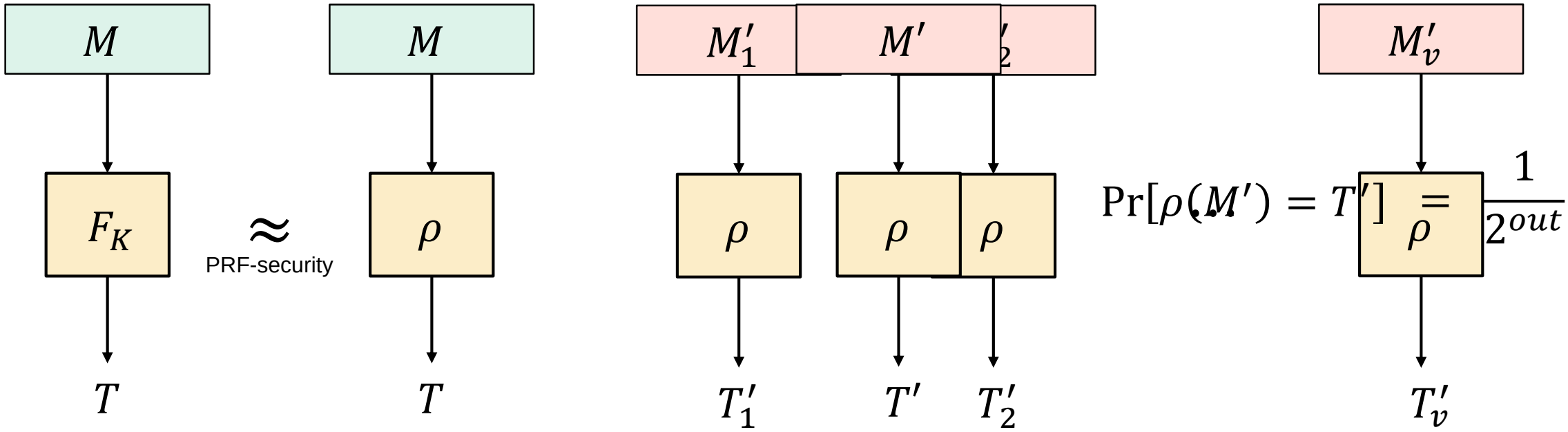


$$\Pr[\rho(M') = T'] = \frac{1}{2^{out}}$$

PRFs are good MACs – proof sketch

Theorem: For *any* UF-CMA adversy A, asking v Vrfy queries, there *is* a PRF-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\Sigma_{\text{PRF}}}(A) \le \mathbf{Adv}^{\text{prf}}_{F}(B) + \frac{v}{2^{out}}$$

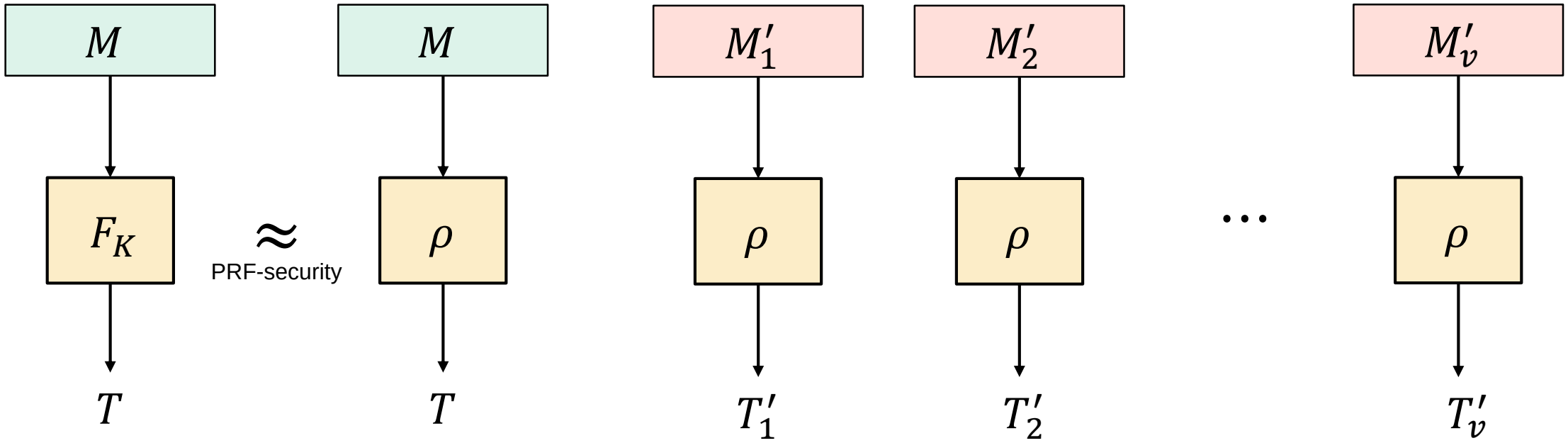


$$\Pr[\rho(M'_1) = T'_1 \ \vee \ \cdots \vee \ \rho(M'_v) = T'_v] \quad \le \quad \sum_{i=1}^{v} \Pr[\rho(M'_i) = T'_i] \quad = \quad v \cdot \frac{1}{2^{out}}$$

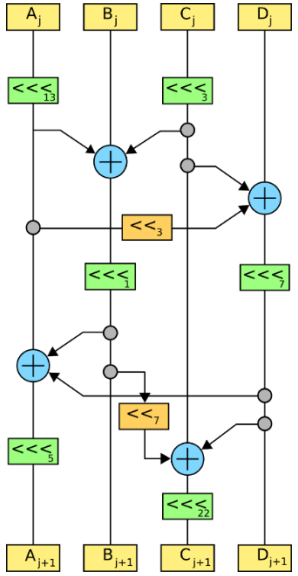
PRFs are good MACs – proof sketch

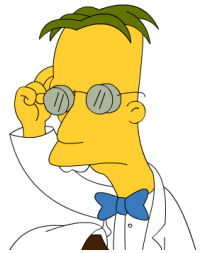
Theorem: For *any* UF-CMA adversy A, asking v Vrfy queries, there *is* a PRF-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\Sigma_{\text{PRF}}}(A) \leq \mathbf{Adv}^{\text{prf}}_{F}(B) + \frac{v}{2^{out}}$$



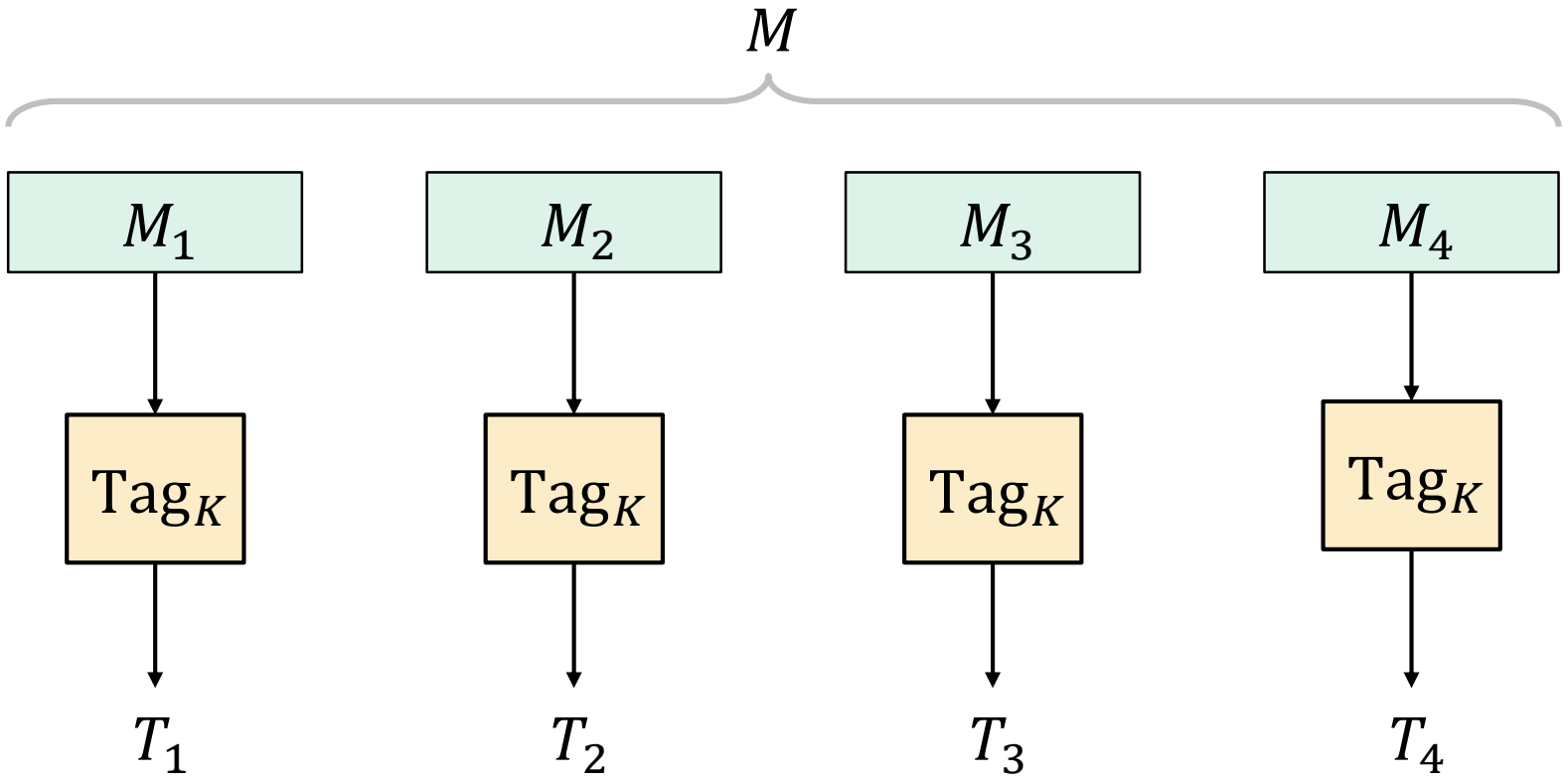
$$\Pr[\rho(M'_1) = T'_1 \ \vee \ \cdots \vee \ \rho(M'_v) = T'_v] \quad \leq \quad \sum_{i=1}^{v} \Pr[\rho(M'_i) = T'_i] \quad = \quad v \cdot \frac{1}{2^{out}}$$





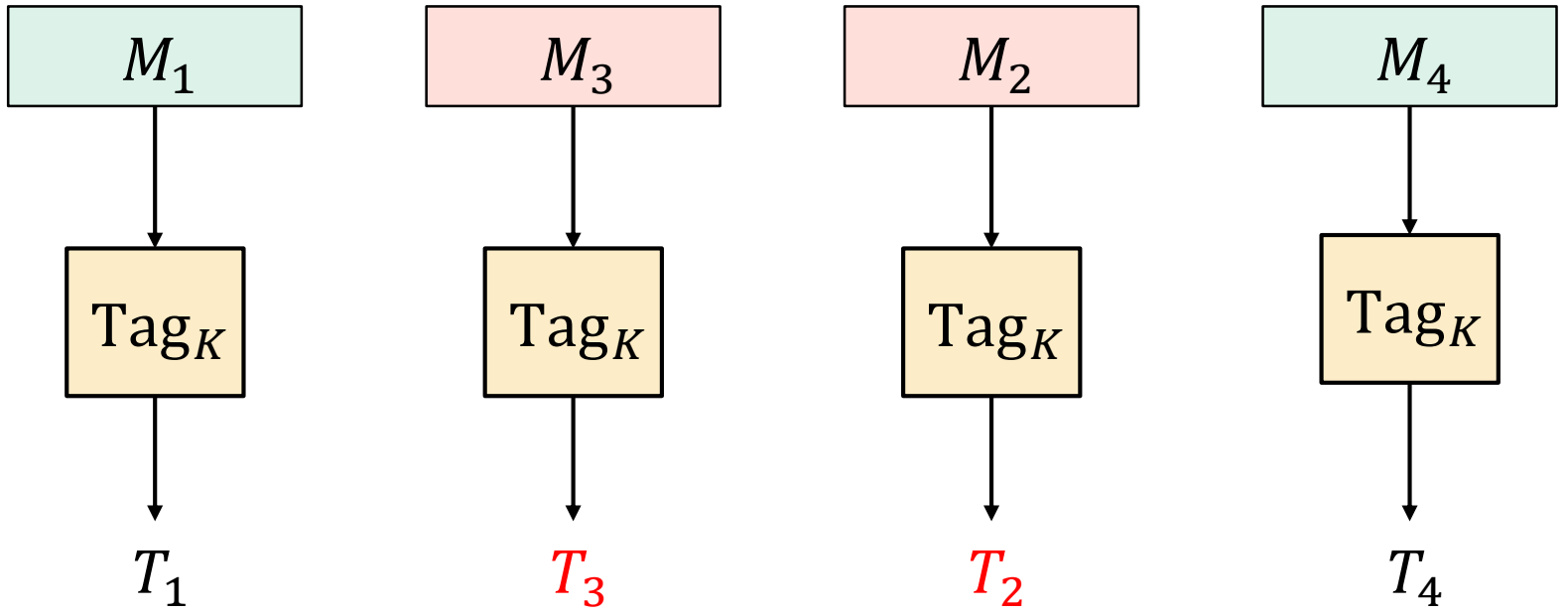
MACs for long messages

Attempt 1



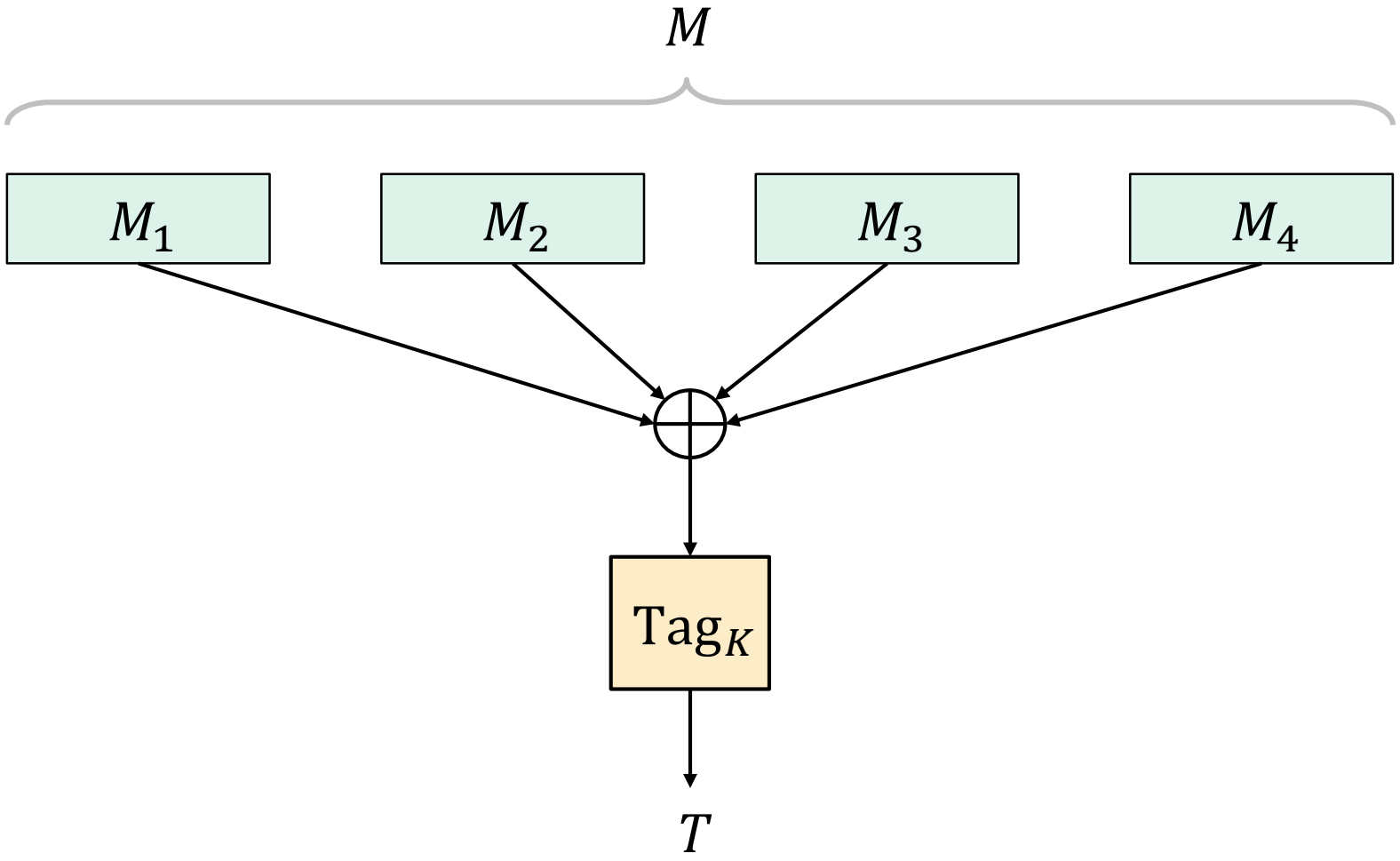
$$T = T_1||T_2||T_3||T_4$$

Attempt 1 – an attack

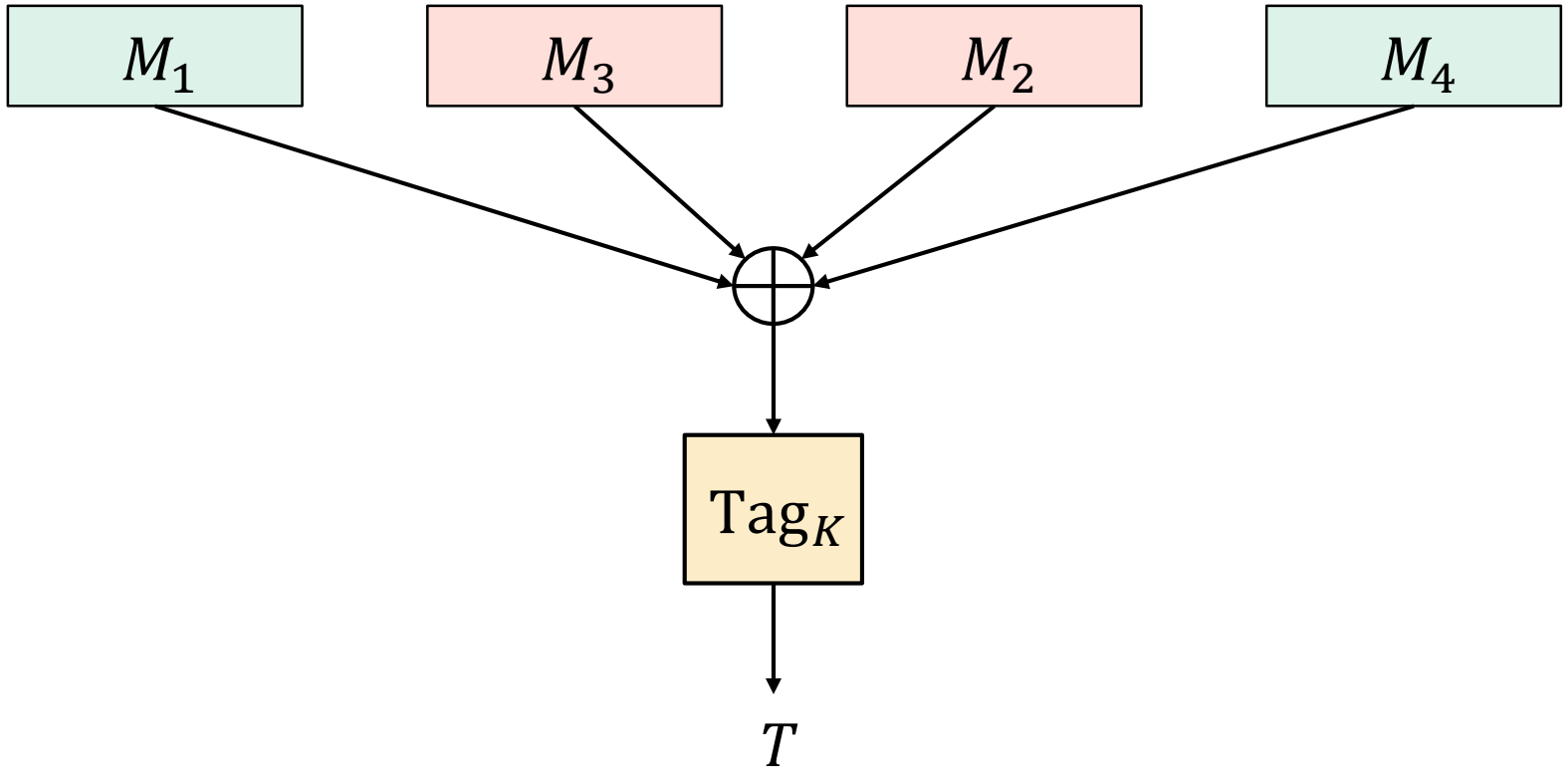


$$T = T_1 || T_3 || T_2 || T_4$$

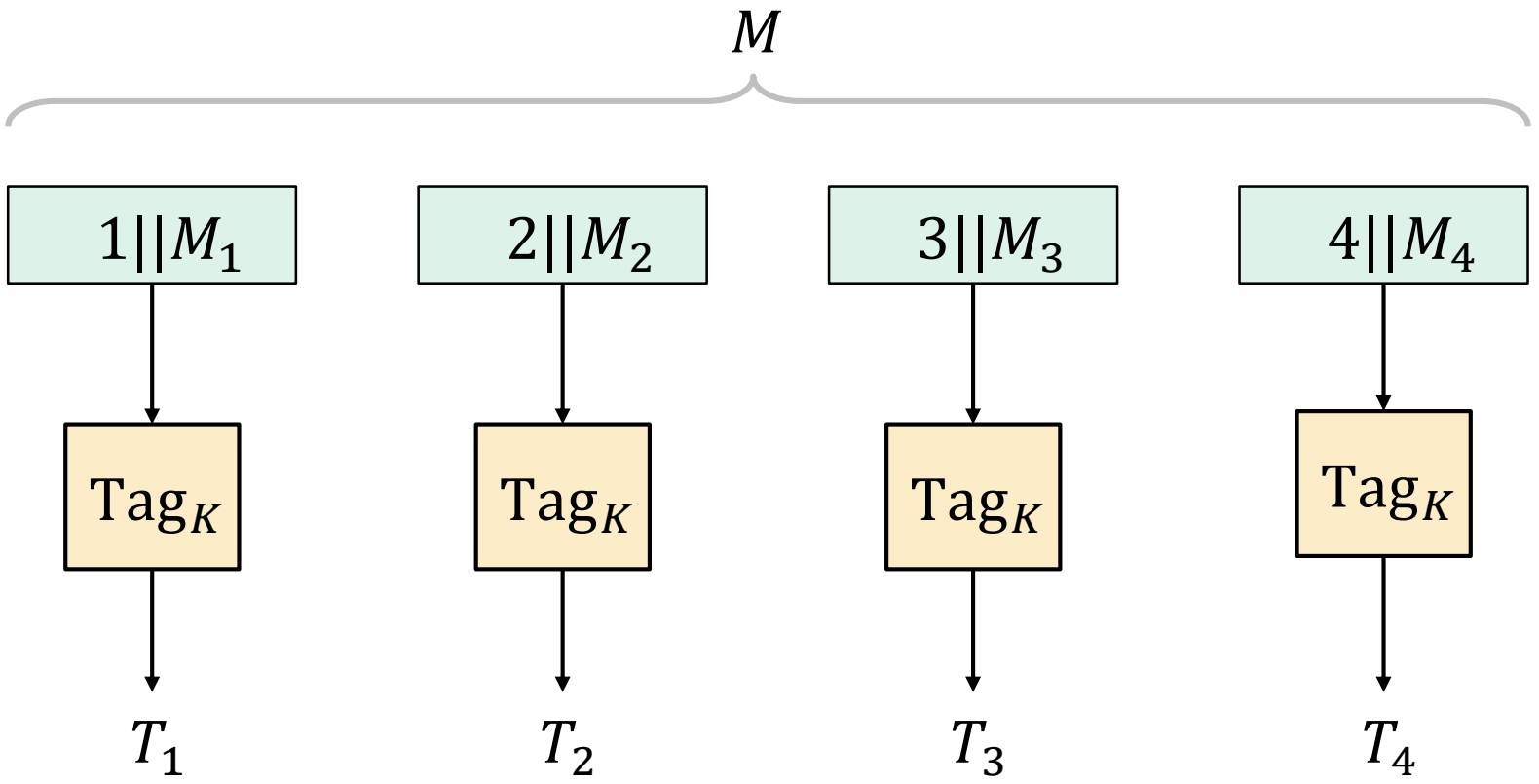
Attempt 2



Attempt 2 – an attack



Attempt 3

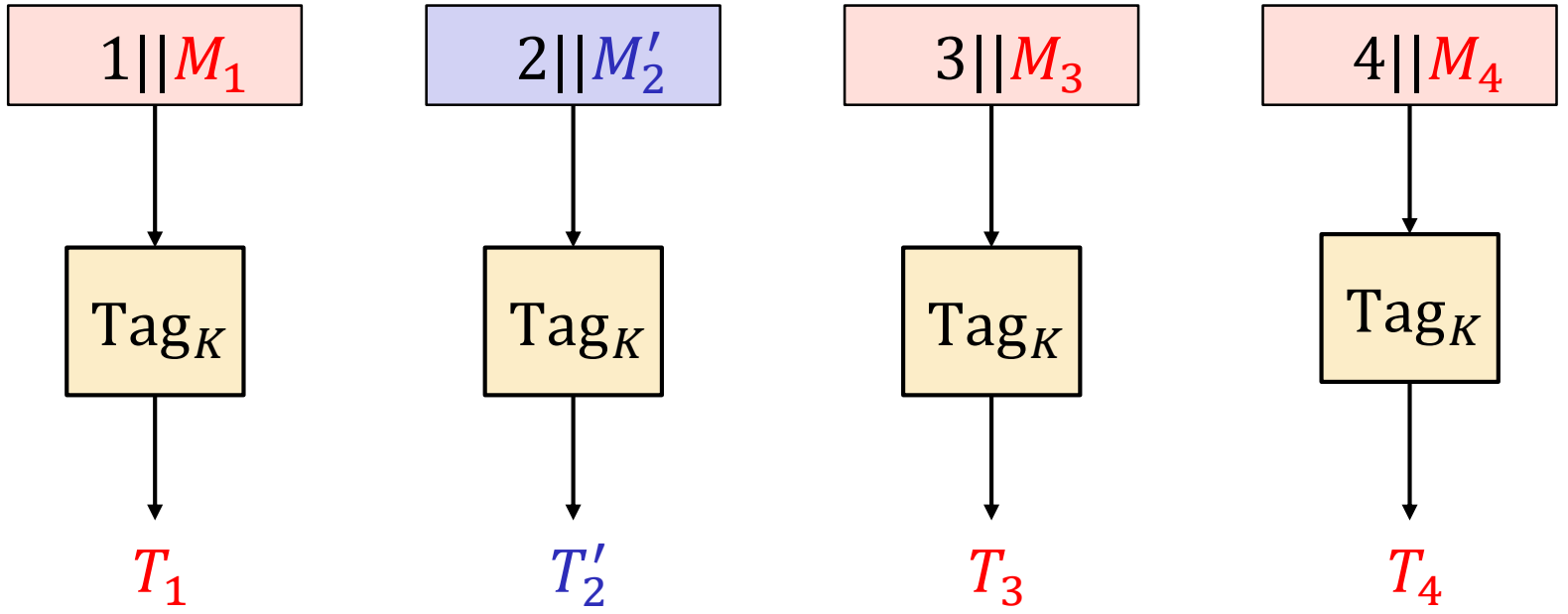


$$T = T_1||T_2||T_3||T_4$$

Attempt 3 – an attack

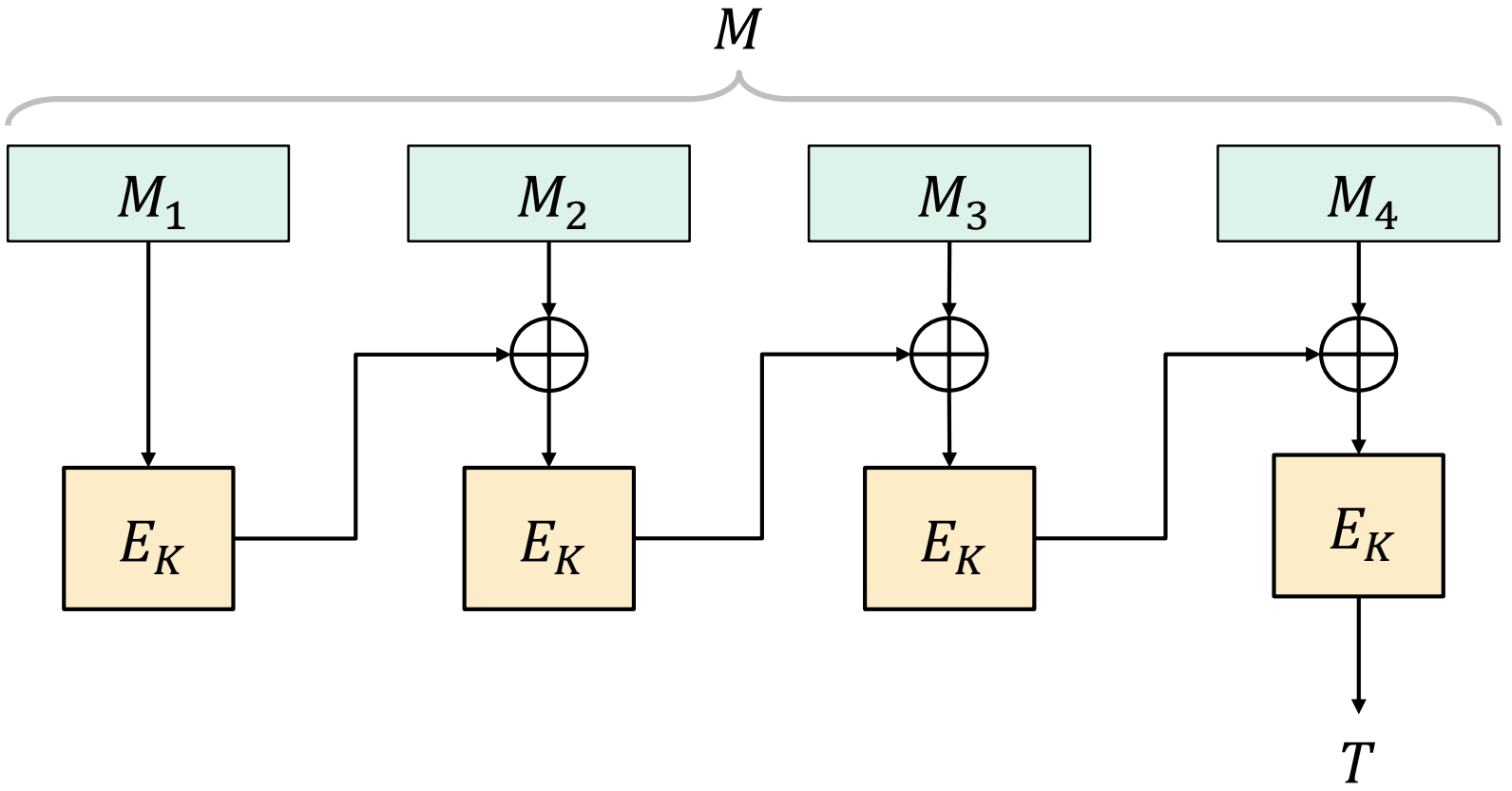
$$M = M_1||M_2||M_3||M_4 \xrightarrow{\text{Tag}_K} T_1||T_2||T_3||T_4$$

$$M' = M'_1||M'_2||M'_3||M'_4 \xrightarrow{\text{Tag}_K} T'_1||T'_2||T'_3||T'_4$$



$$T = T_1||T'_2||T_3||T_4$$

CBC-MAC

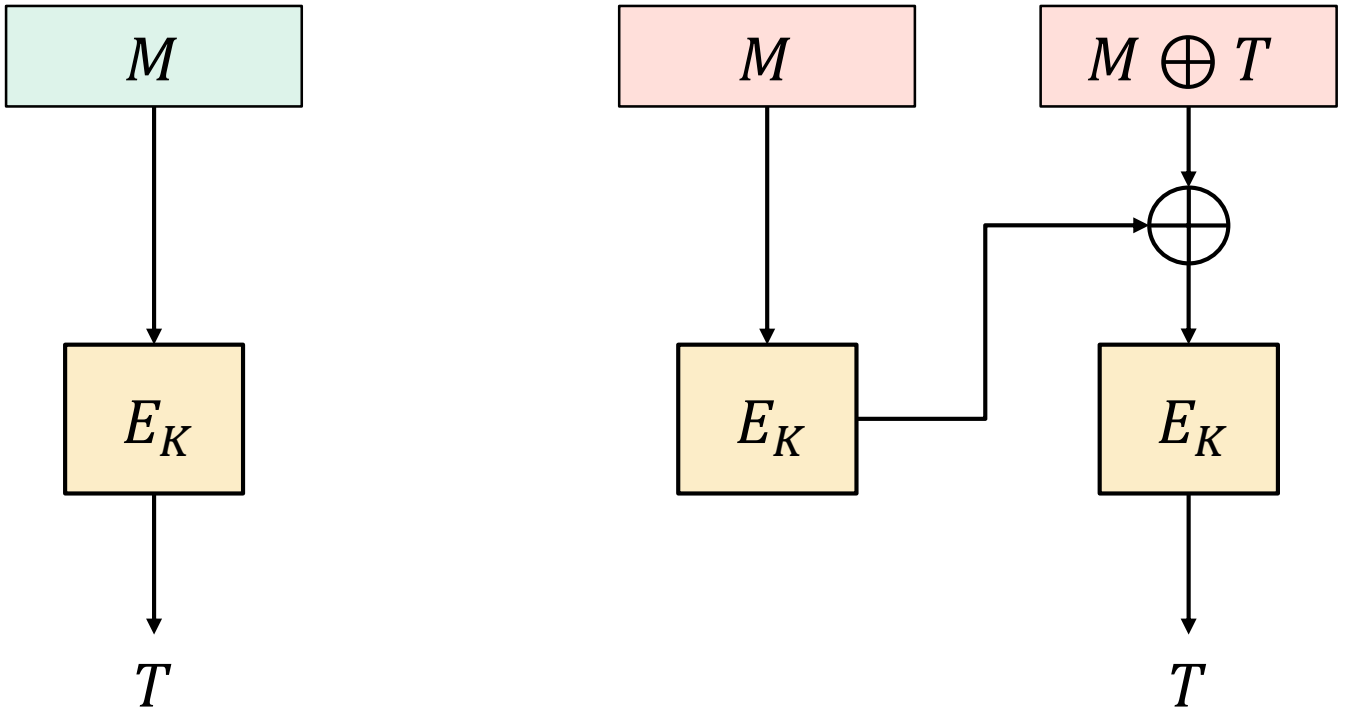


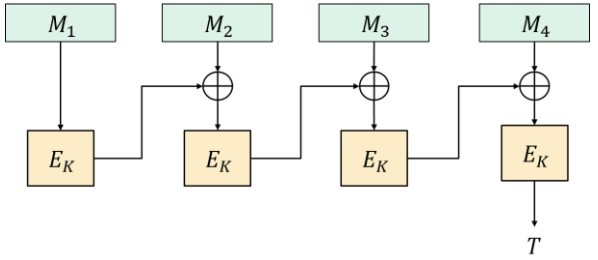
CBC-MAC – security

Theorem: If $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is a secure block cipher, then for any *fixed* ℓ CBC-MAC is UF-CMA secure for messages $M \in \{0,1\}^{\ell \cdot n}$.

- CBC-MAC is secure if you *only* MAC messages of length $4 \cdot n$
- CBC-MAC is secure if you *only* MAC messages of length $23 \cdot n$
- CBC-MAC is **not** secure if you MAC messages of lengths $4 \cdot n$ *and* $23 \cdot n$

CBC-MAC – pitfalls; variable-length messages



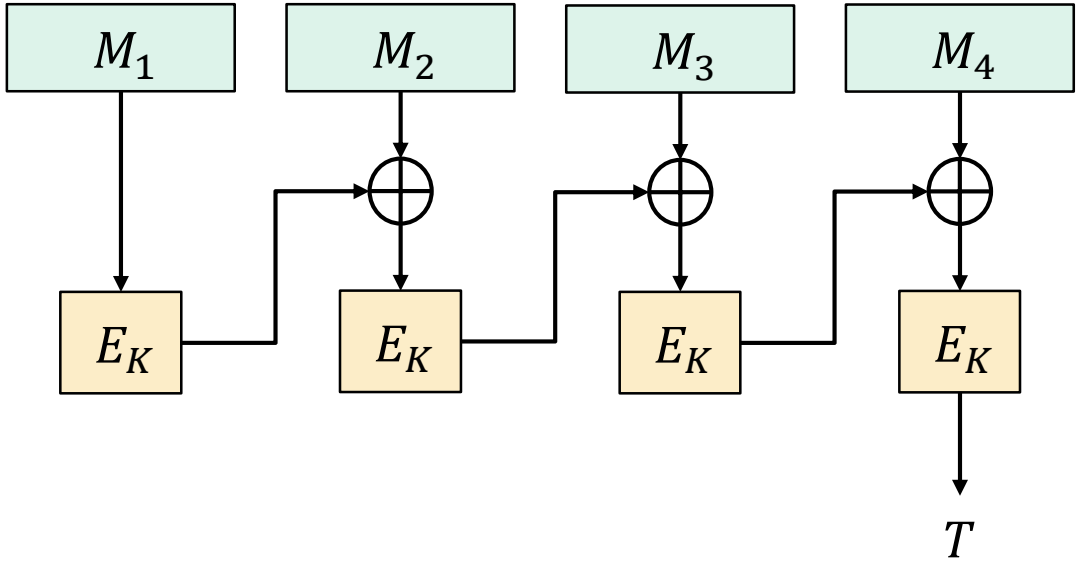


$$\begin{aligned}\text{CBC.Tag}(K, M||M \oplus T) &= E_K\big(E_K(M) \oplus (M \oplus T)\big) \\ &= E_K\left(T \oplus (M \oplus T)\right) \\ &= E_K\left(M\right) \\ &= T\end{aligned}$$

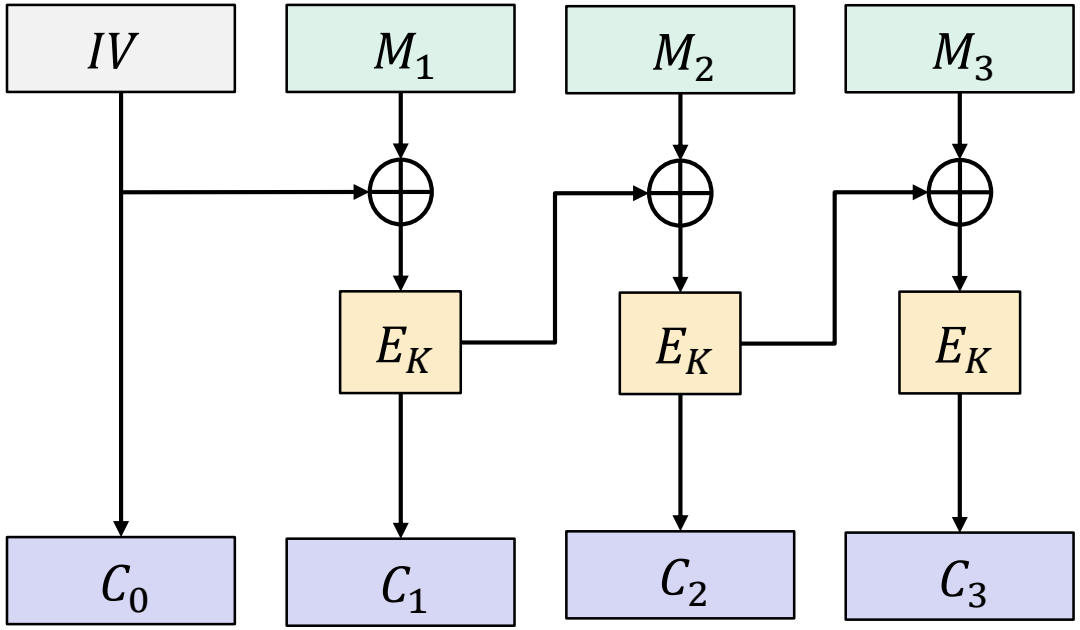
$$\mathbf{Adv}^{\text{uf-cma}}_{\text{CBC-MAC}}(A) = 1$$

CBC-MAC vs. CBC$-encryption

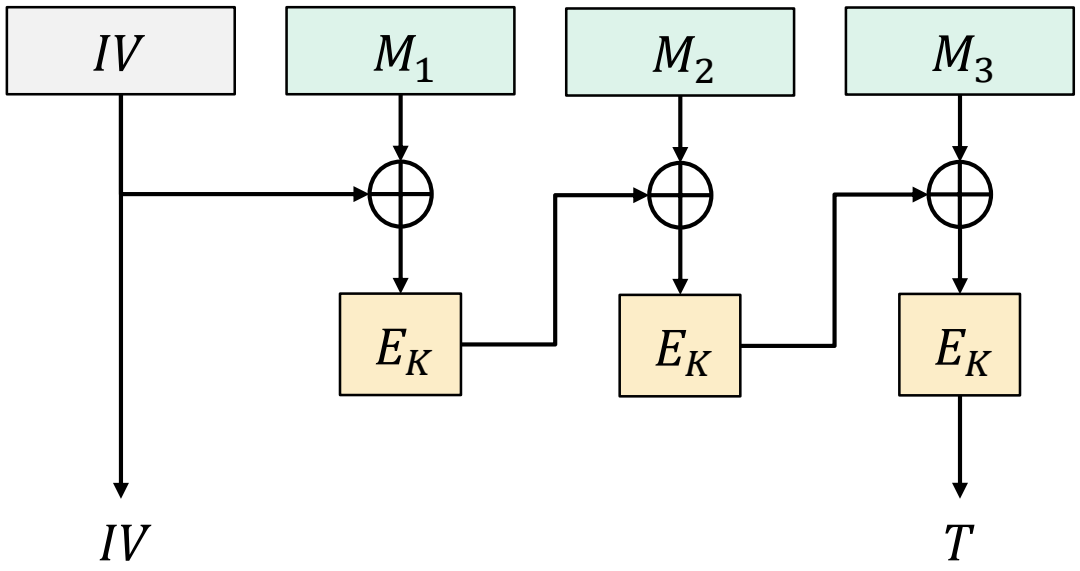
CBC-MAC

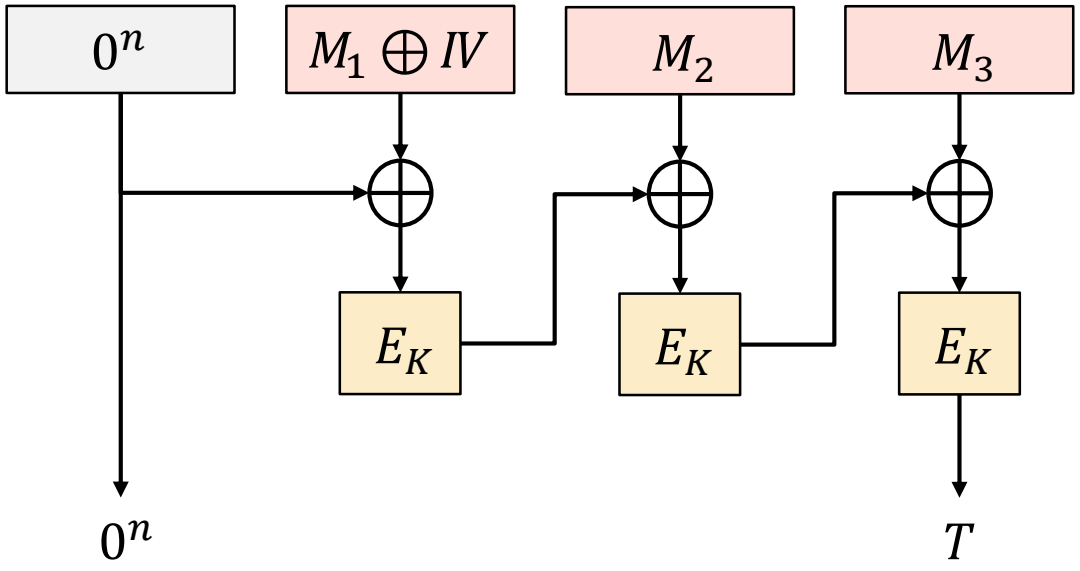


CBC$-encryption



CBC$-MAC – pitfalls; randomized IV

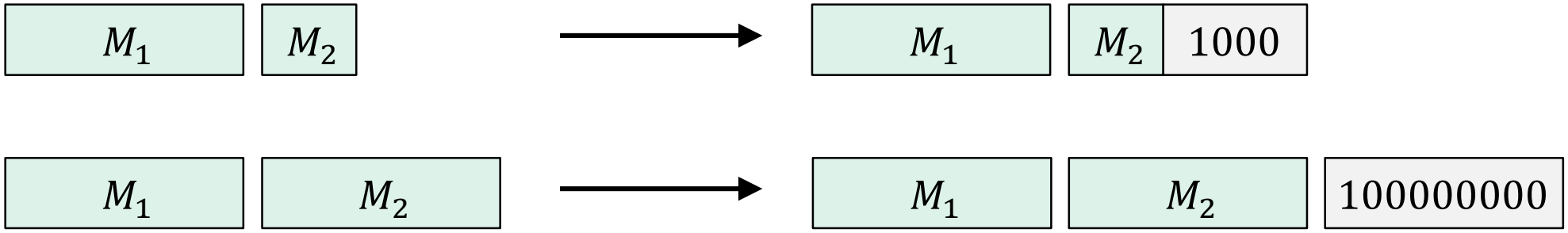




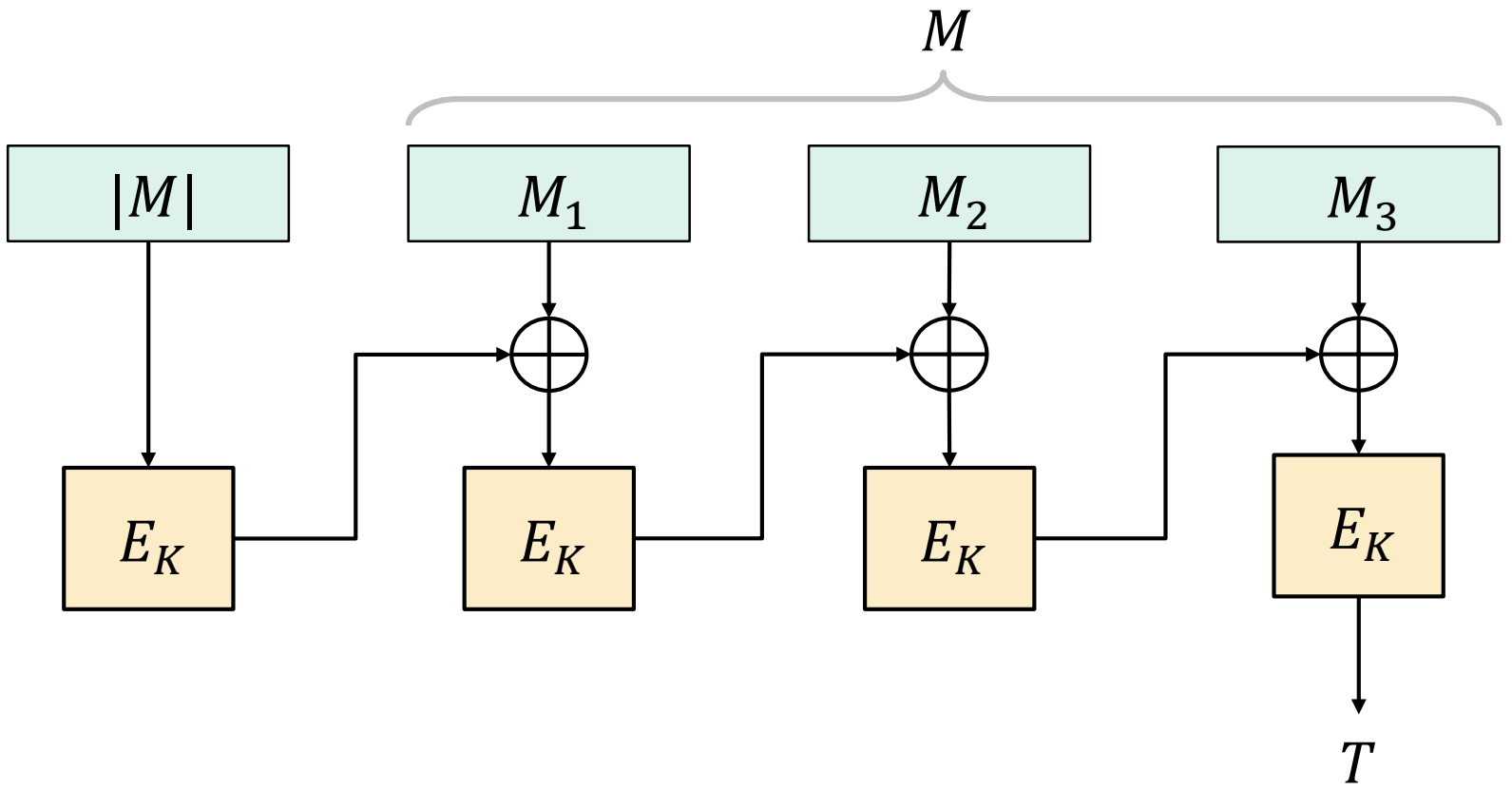
Allowing variable-length messages

- Want to support different-length messages *simultaneously*
  - Want to MAC messages of length $4 \cdot n$ *and* $23 \cdot n$
  - Could use different keys for each length $\Longrightarrow$ not practical

- Want to support message not a multiple of the block length
  - **Padding** needed



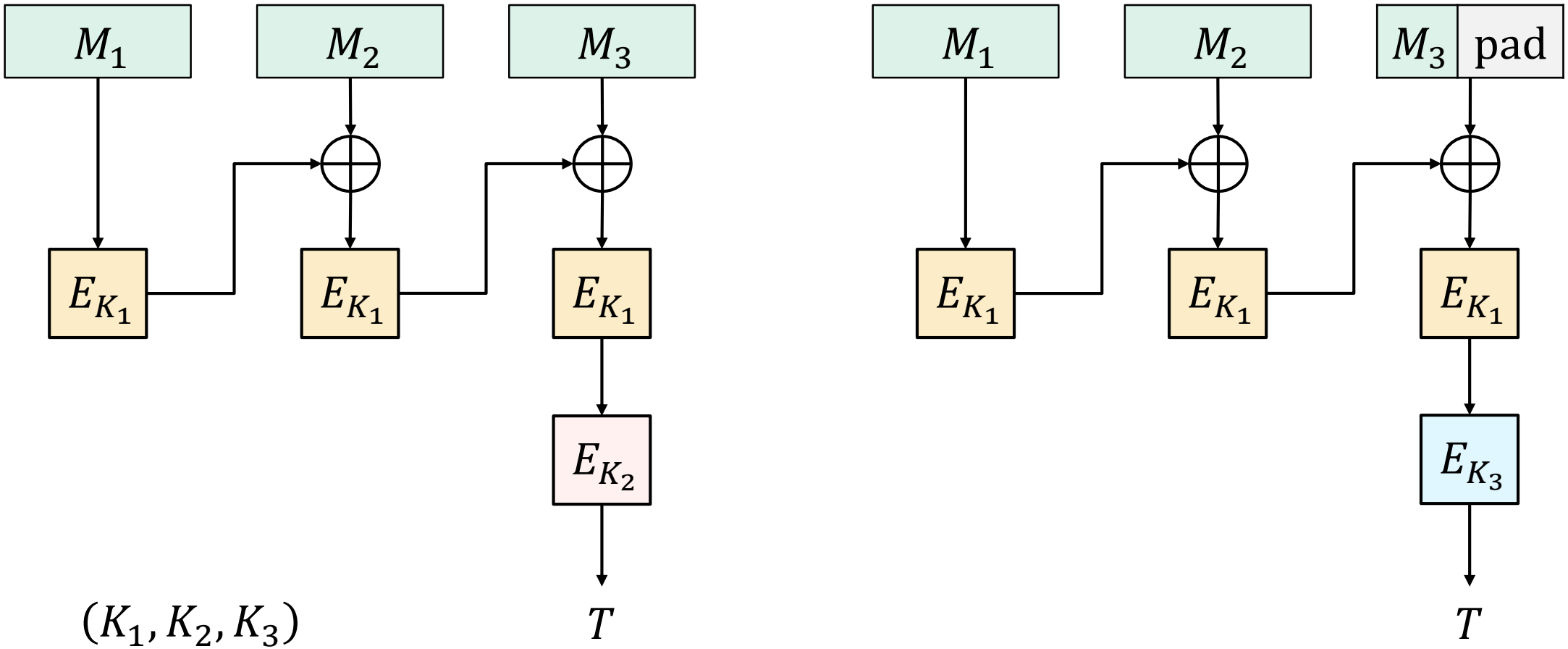
CBC-MAC for variable-length messages



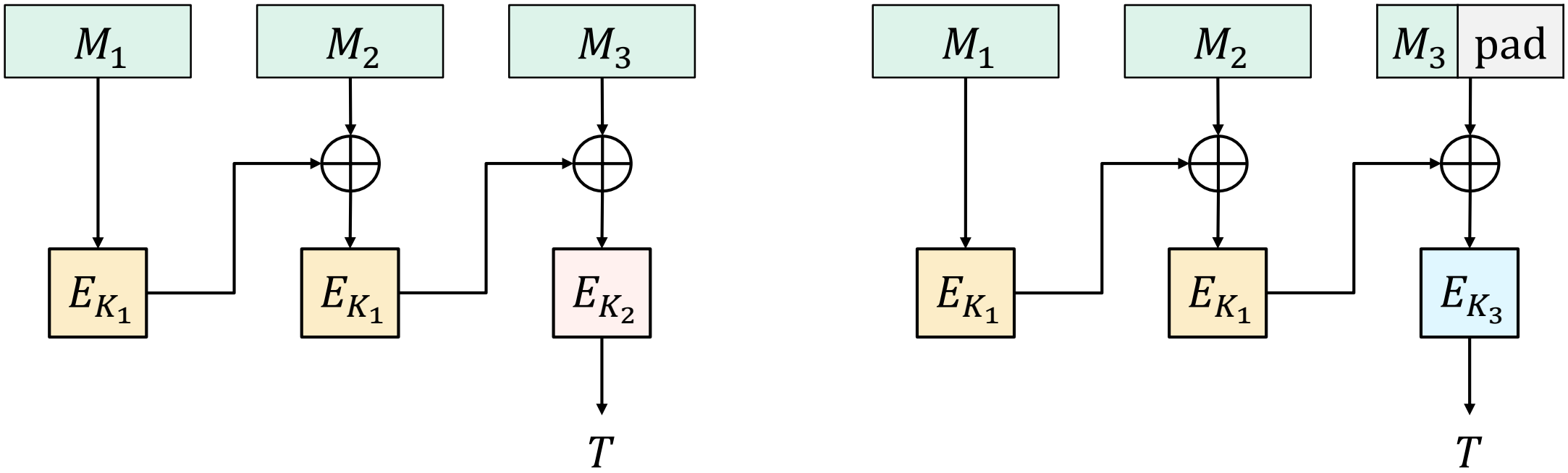
Theorem: If E is a secure PRP then length-prepended CBC-MAC is UF-CMA secure

What if you append the length of M at the *end* rather than at the beginning?

ECBC-MAC

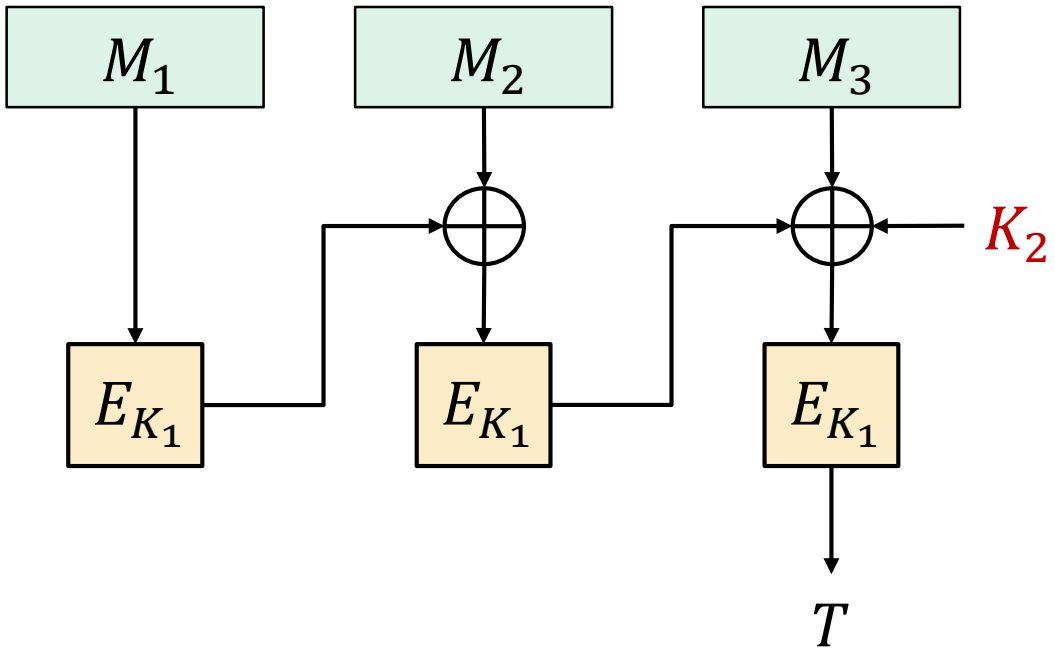


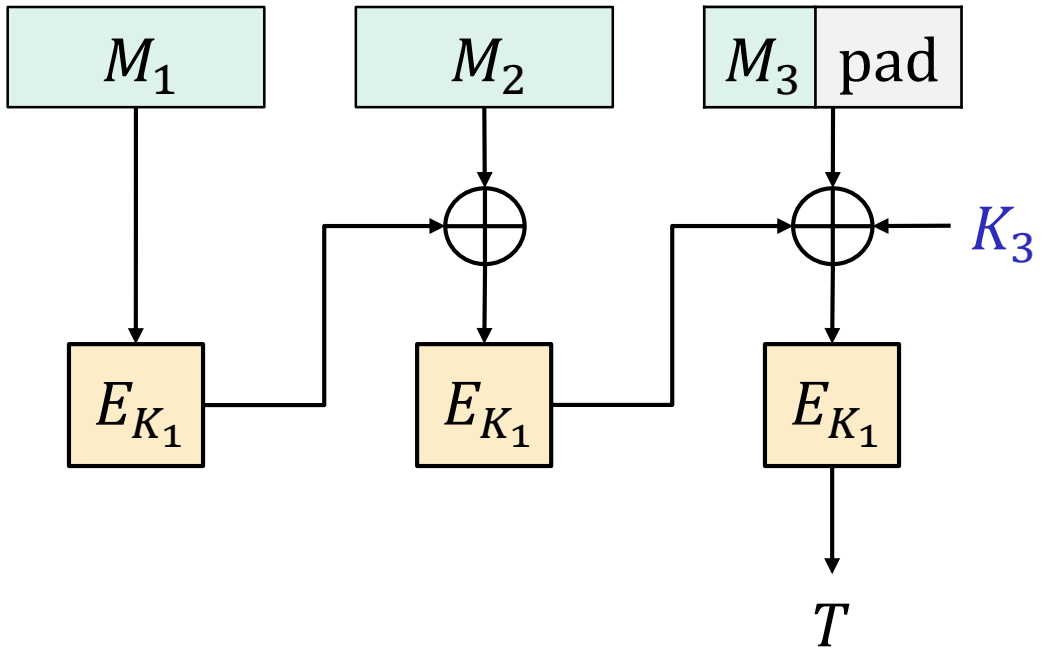
FCBC-MAC



(K_1, K_2, K_3)

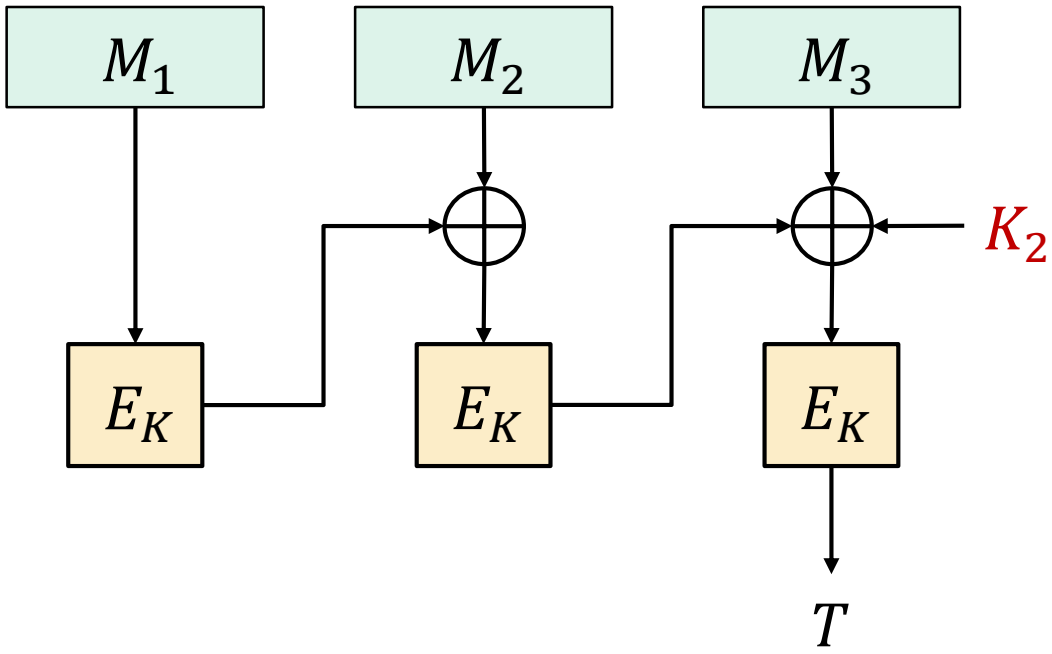
XCBC-MAC

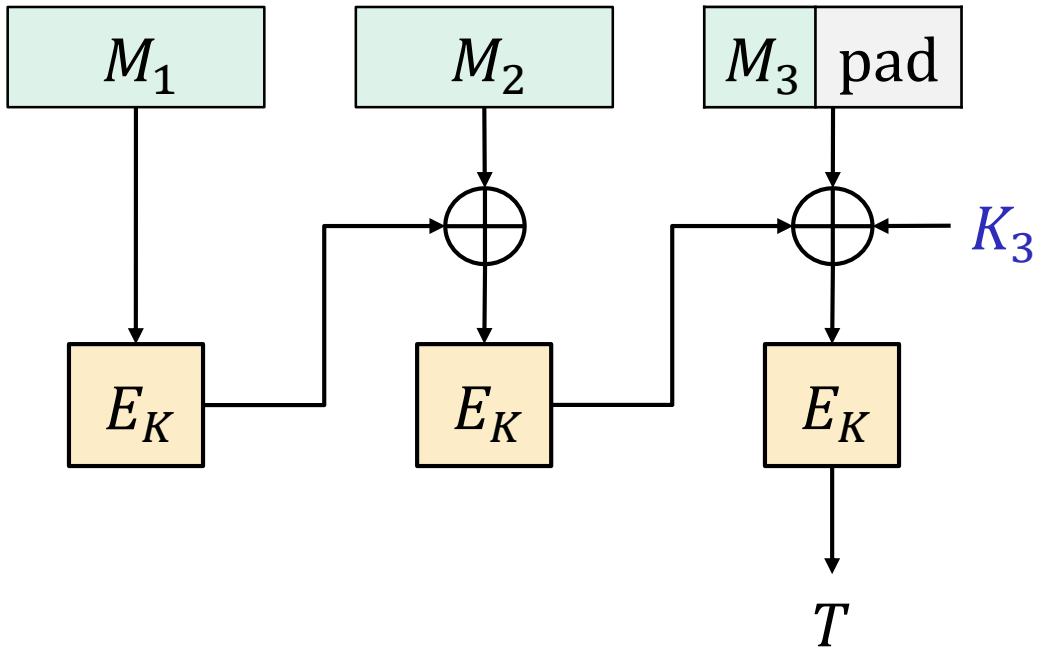




(K_1, K_2, K_3)

CMAC a.k.a. One-key MAC (OMAC)



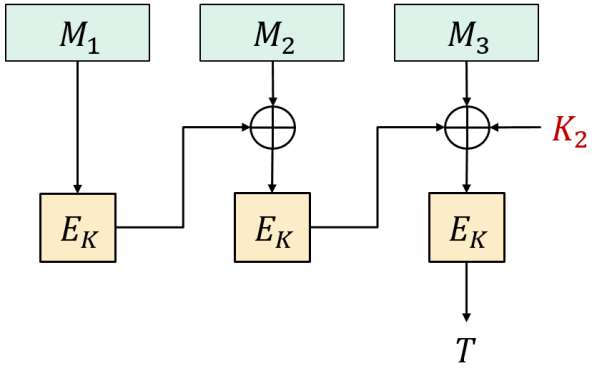


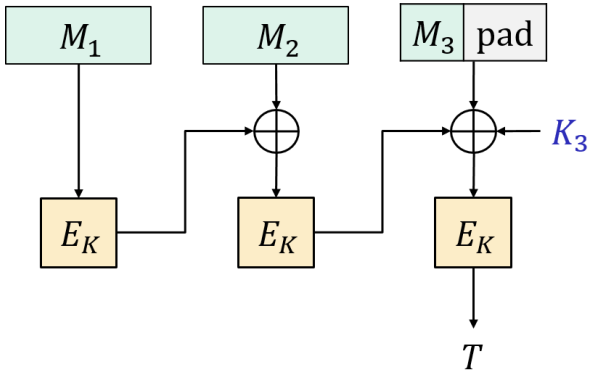
$$K_2 = 2 * E_K(0^n)$$

$$K_3 = 4 * E_K(0^n)$$

CMAC

- 1 E-call per message block + 1 E-call to derive K_2, K_3
- Fully sequential
  - Cannot utilize parallel AES cores
- Only one key
- Standardized by NIST





$K_2 = 2 * F_K(0^n)$

$K_3 = 4 * F_K(0^n)$

CMAC security

Theorem: If $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is a secure PRP then CMAC is UF-CMA secure.

Detailed: (Nandi '09) For *any* UF-CMA adversary making q Tag-queries, each of max length ℓ n-bit blocks, and v Vrfy-queries, there *is* a PRP-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\text{CMAC}}(A) \leq \mathbf{Adv}^{\text{prp}}_{E}(B) + \frac{5 \cdot \ell \cdot q^2}{2^n} + \frac{v}{2^n}$$

- **Example:**
  - $E = \text{AES} \quad (n = 128)$
  - Number of Tag-queries: $q = 2^{40} \approx 1$ trillion queries
  - Max n-bit blocks per message: $\ell = 2^{16} \approx 1$ MB
  - Number of Vrfy-queries: $v = 2^{40}$
  - $\mathbf{Adv}^{\text{prp}}_{E}(B) \approx 0$

$$\mathbf{Adv}^{\text{uf-cma}}_{\text{AES-MAC}}(A) \leq \frac{5 \cdot 2^{16} \cdot (2^{40})^2}{2^{128}} + \frac{2^{40}}{2^{128}} \approx \frac{2^{3+16+80}}{2^{128}} = \frac{1}{2^{29}}$$

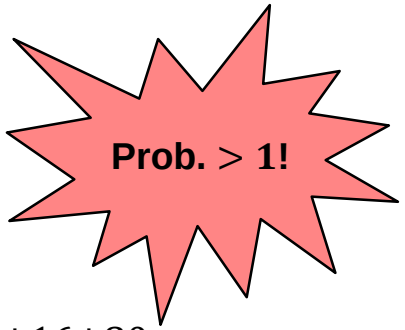
CMAC security

Theorem: If $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is a secure PRP then CMAC is UF-CMA secure.

Detailed: (Nandi '09) For *any* UF-CMA adversary making q Tag-queries, each of max length ℓ n-bit blocks, and v Vrfy-queries, there *is* a PRP-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\text{CMAC}}(A) \leq \mathbf{Adv}^{\text{prp}}_{E}(B) + \frac{5 \cdot \ell \cdot q^2}{2^n} + \frac{v}{2^n}$$

- **Example:**
  - $E = \text{DES} \quad (n = 64)$
  - Number of Tag-queries: $q = 2^{40} \approx 1$ trillion queries
  - Max n-bit blocks per message: $\ell = 2^{16} \approx 1$ MB
  - Number of Vrfy-queries: $v = 2^{40}$
  - $\mathbf{Adv}^{\text{prp}}_{E}(B) \approx 0$



$$\mathbf{Adv}^{\text{uf-cma}}_{\text{DES-CMAC}}(A) \leq \frac{5 \cdot 2^{16} \cdot (2^{40})^2}{2^{64}} + \frac{2^{40}}{2^{64}} \approx \frac{2^{3+16+80}}{2^{64}} = 2^{35}$$

CMAC security

Theorem: If $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is a secure PRP then CMAC is UF-CMA secure.

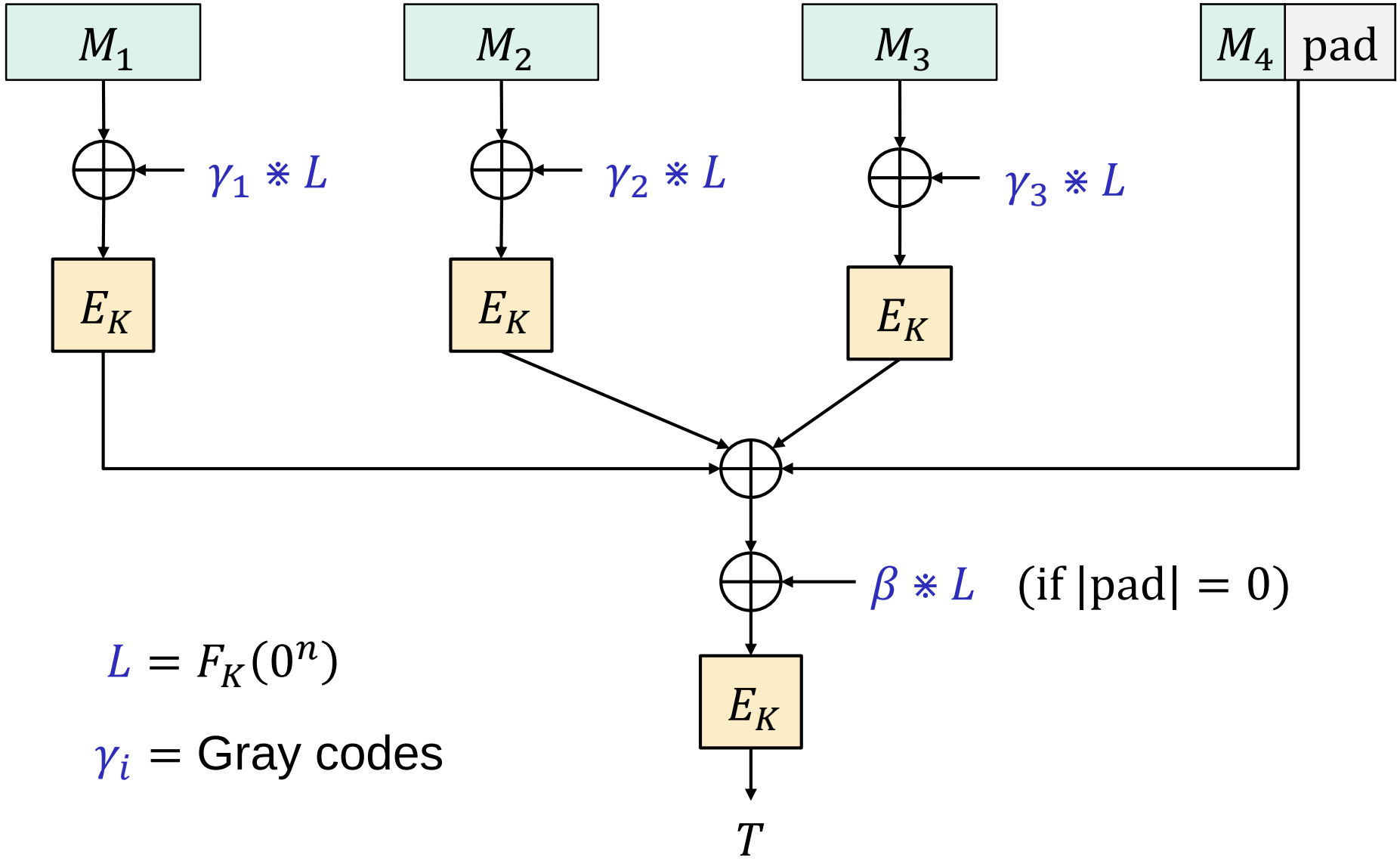
Detailed: (Nandi '09) For *any* UF-CMA adversary making q Tag-queries, each of max length ℓ n-bit blocks, and v Vrfy-queries, there *is* a PRP-adversary B such that

$$\mathbf{Adv}^{\text{uf-cma}}_{\text{CMAC}}(A) \leq \mathbf{Adv}^{\text{prp}}_{E}(B) + \frac{5 \cdot \ell \cdot q^2}{2^n} + \frac{v}{2^n}$$

- **Example:**
  - $E = \text{DES} \quad (n = 64)$
  - Number of Tag-queries: $q = 2^{16} \approx 65{,}000$ queries
  - Max n-bit blocks per message: $\ell = 2^{10} \approx 1$ kB
  - Number of Vrfy-queries: $v = 2^{40}$
  - $\mathbf{Adv}^{\text{prp}}_{E}(B) \approx 0$

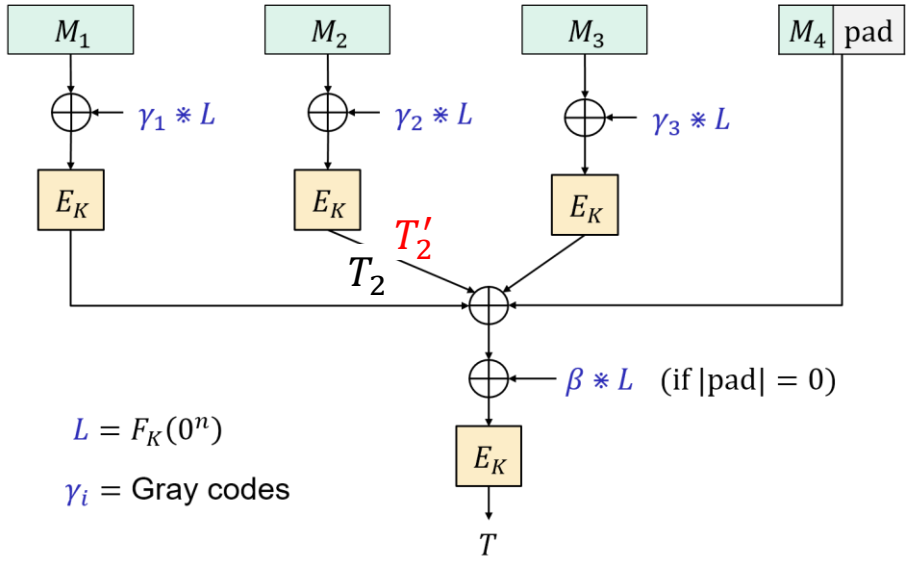
$$\mathbf{Adv}^{\text{uf-cma}}_{\text{DES-CMAC}}(A) \leq \frac{5 \cdot 2^{10} \cdot (2^{16})^2}{2^{64}} + \frac{2^{40}}{2^{64}} \approx \frac{2^{3+10+32}}{2^{64}} = \frac{1}{2^{19}}$$

PMAC


M_1
M_2
M_3
M_4 pad
$\gamma_1 * L$
$\gamma_2 * L$
$\gamma_3 * L$
E_K
E_K
E_K
$\beta * L$ (if |pad| = 0)
E_K
T
$L = F_K(0^n)$
γ_i = Gray codes

PMAC properties

- Fully parallelizable
- Incremental
- One key
- UF-CMA secure
- …not used in practice



$$\text{PMAC}(K, M_1||M_2|| \dots ||M_{1000}) = T$$

$$\text{PMAC}(K, M_1||M_2'|| \dots ||M_{1000}) = E_K(E_K^{-1}(T) \oplus T_2 \oplus T_2')$$

$$T_2 = E_K(M_2 \oplus \gamma_2 * L)$$

$$T_2' = E_K(M_2' \oplus \gamma_2 * L)$$

Summary

- UF-CMA the right security notion for message integrity
  - Does not cover replay attacks
- PRFs are good MACs
  - But usually of short (fixed) input length
- CBC-MAC good MAC for messages of a ***single*** *fixed* length
- CMAC upgrades CBC-MAC to variable-length messages