# Bounding Distinguishing Advantages of Two Functions

PRF

$$Adv_{F_k, f}(D) := \left| Pr_k[D^{F_k}(\cdot) = 1] - Pr_f[D^{f}(\cdot) = 1] \right|$$

$f : \{0,1\}^n \to \{0,1\}^n$

PRP

$$Adv_{F_k, \pi}(D) := \left| Pr_k[D^{F_k}(\cdot) = 1] - Pr[D^{\pi}(\cdot) = 1] \right|$$

$F = f \xleftarrow{\$} Func(n)$

$G = \pi \xleftarrow{\$} Perm(n)$

Bounding distinguishing Adv of $F \not\approx G$

$$Adv_{F,G}(\mathcal{D}) := \left| Pr[D^F = 1] - Pr[D^G = 1] \right|$$

$$\Downarrow$$

$$Adv_{f,\pi}(\mathcal{D}) = \binom{q(n)}{2} \cdot \frac{1}{2^n}$$

PRP-PRF Switching Lemma

RF

$$\Pr\left[f(x_1)=y_1 \mid f(x_2)=y_2,\ f(x_3)=y_3\right] = \frac{1}{2^n}$$

$$\Pr\left[\pi(x_1)=y_1 \mid \pi(x_2)=y_2,\ \pi(x_3)=y_3\right] = \frac{1}{2^n-2}$$

RP

$\mathcal{D} \rightarrow x_1, x_2, \ldots, x_q$

$\downarrow$

$y_1, y_2, \ldots, y_q$

$\Pr[y_i = y_j] = \frac{1}{2^n}$

$\Pr[\exists i,j ;\ y_i = y_j] = \binom{q}{2} \cdot \frac{1}{2^n}$

$\mathcal{O}$ $\quad$ $\mathcal{D}$ $\quad$ (Adaptive)

$x_1$ ←

$y_1$ →

$x_2$ ←

$y_2$ →

$\vdots$

$x_q$ ←

$y_q$ →

$b$ ←

$d_r$

$x_1 = \phi_1(d_r)$

$x_2 = \phi_2(d_r, y_1)$

$\vdots$

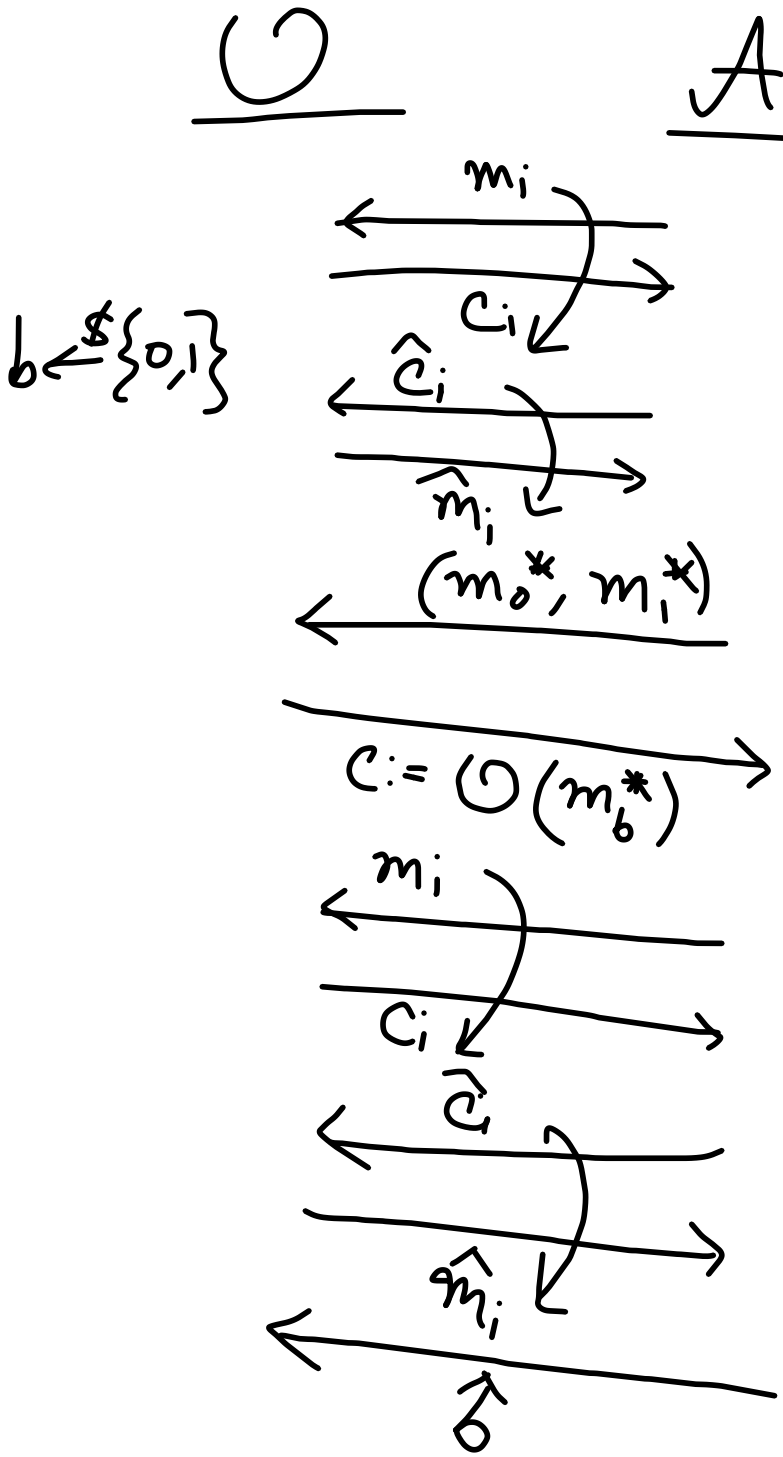
$x_q = \phi_q(d_r, y_1, \ldots, y_{q-1})$

$b = \phi(d_r, y_1, \ldots, y_q)$

$(d_r, \phi_1, \ldots \phi_q, \phi) \Longrightarrow \mathcal{D}$

$\Longrightarrow$ Model of an Adversary

# IND-CCA Notion




$\Pi$ is ind-cca secure if

$\forall A$, $\exists$ negligible func negl s.t.

$$Pr\left[PrivK^{CCA}_{A,\Pi} = 1\right] := Pr\left[b = \hat{b}\right]$$

$$\leq \frac{1}{2} + negl(n)$$

## CPA-Secure:

$\langle r, F_k(r) \oplus m \rangle$

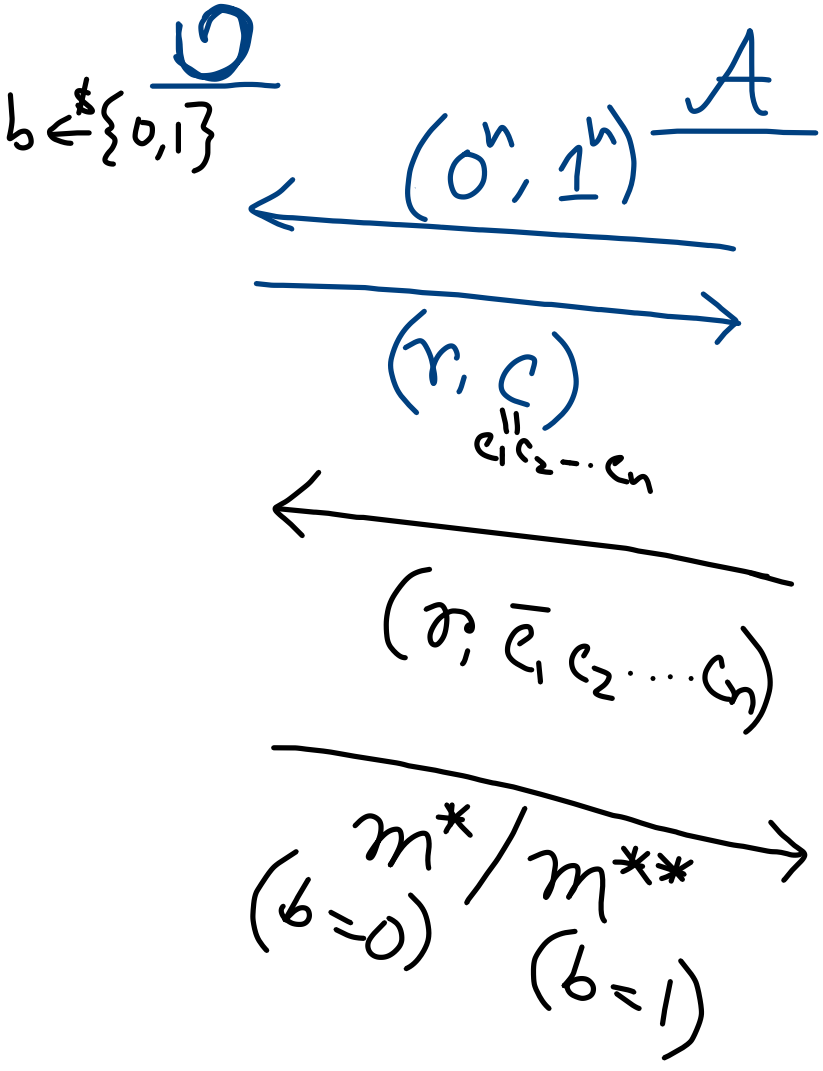
↓

Is it CCA Secure?

**No**

## H/W

CBC,
OFB,
CTR




$c' = \langle r, F_k(r) \rangle$

$c'' = \langle r, F_k(r) \oplus 1^n \rangle$

Left
$c' = \langle r, F_k(r) \rangle$
$c^* = \langle r, F_k(r) \oplus 10^{n-1} \rangle \Rightarrow m = 0^n$

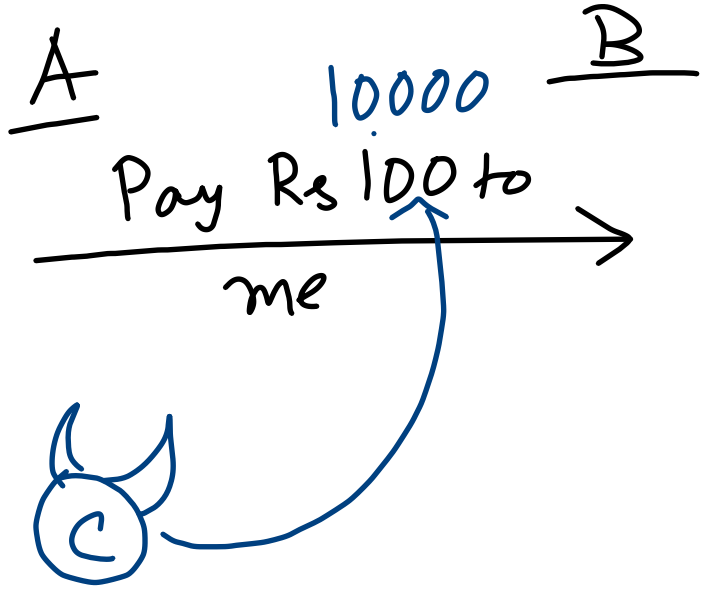
$\Rightarrow \boxed{m^* = 10^{n-1}}$

Right
$c''' = \langle r, F_k(r) \oplus 1^n \rangle$
$c^{**} = \langle r, F_k(r) \oplus 01^{n-1} \rangle \Rightarrow \boxed{m^{**} = 01^{n-1}}$

# CCA Security

- You have to stop adversary making ciphertext queries in a controlled-way.
  - Not possible with simple mode of operations
  - Authenticated Encryption
    - Ciphertext queries maynot be valid
- Non-malleability
  - Invalid C/T queries
  - Valid C/T queries are not related

# Message Authentication Codes




- Message Integrity

$\Pi = (KG, TG, Vrfy)$

Tag Generation

Verify

$$\begin{cases} k \leftarrow KG(1^n) \\ t \leftarrow TG_k(m) \\ 1/0 \leftarrow Vrfy_k(m, t) \\ (\top/\bot) \end{cases}$$

Correctness:

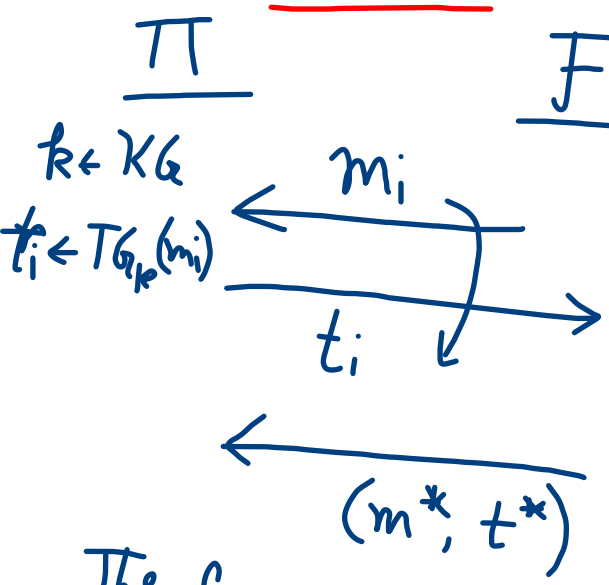
$\forall k, m$

$Vrfy_k(m, TG_k(m)) = 1$ }

# Security

$\Pi = (KG, TG, Vrfy)$ is secure message authentication Code if

$MAC_\Pi$




The forger F wins the game if

i) $t^* = TG_k(m^*)$ — Valid
ii) $m^* \notin \{m_1, \ldots m_q\}$ — Fresh

① → { Existential Forgery

---

$\Pi$ is EUF-CMA (existentially unforgeable under chosen message attack) if

$$Pr[F \text{ wins } MAC_\Pi] \leq negl(n) \quad ①$$

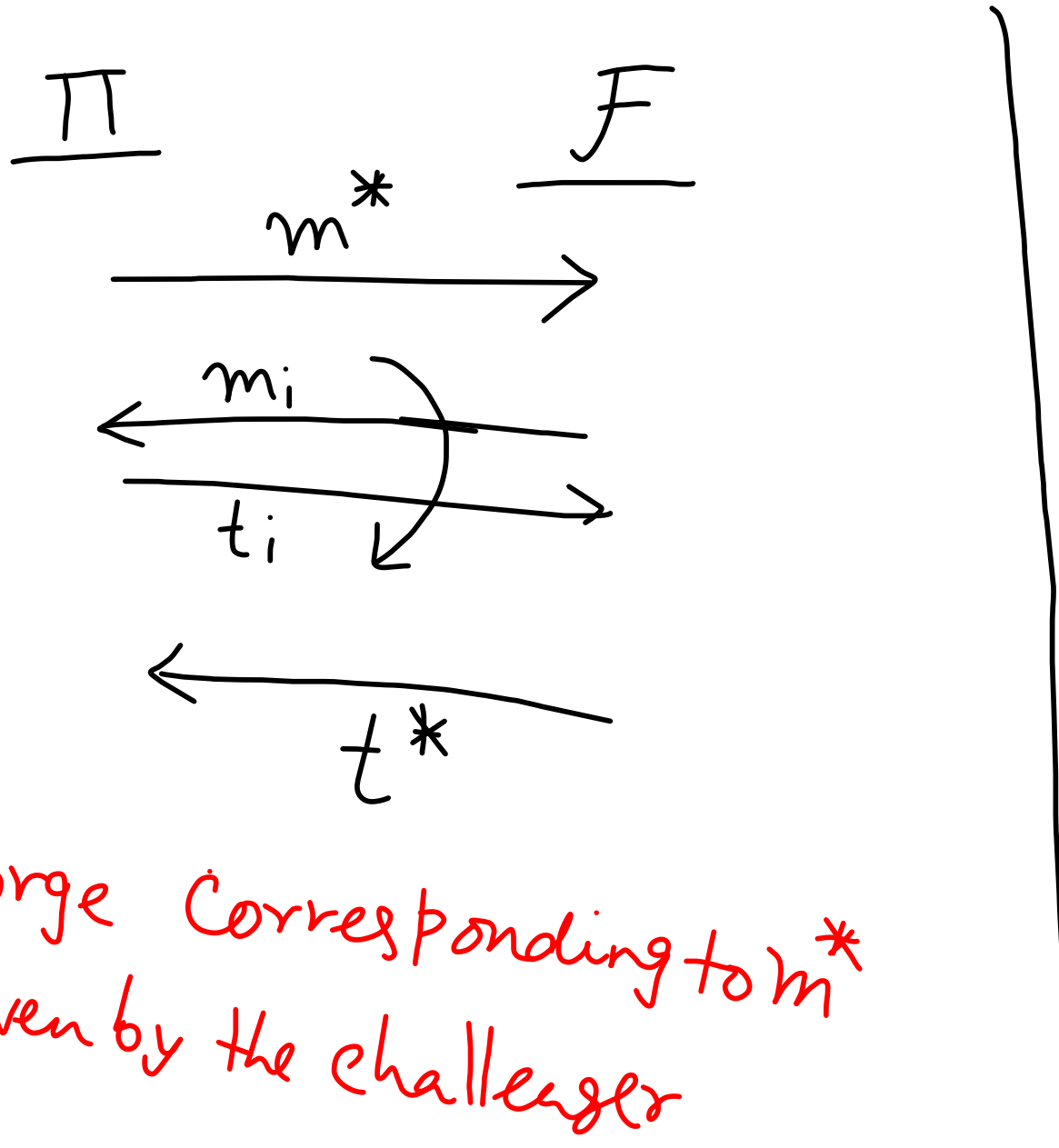
$\Pi$ is SUF-CMA (Strongly unforgeable...) if

$$Pr[F \text{ wins } MAC_\Pi] \leq negl(n) \quad ②$$

i) Same
ii) $(m^*, t^*) \notin \{(m_1, t_1), \ldots, (m_q, t_q)\}$ ②

Strong Forgery

# Universal Forgery




Forge corresponding to $m^*$ given by the challenger

# Fixed length MACs

$F$: PRF $\quad F_k : \{0,1\}^n \to \{0,1\}^n$

$TG_k(m) = F_k(m)$

$\Pi = (KG, TG, Vrfy)$

$Vrfy_k(m, t)$

{

$t^* \leftarrow F_k(m)$

if $t = t^*$ else return 1

return 0

}

If $F$ is a PRF then $\Pi$ is EUF-CMA.

$\mathcal{A}$ $\quad$ $\mathcal{F}$

$m_i$ ⟵ $\quad$ ⟵ $m_i$

$t_i$ ⟶ $\quad$ ⟶ $t_i$

$m^*$ ⟵ $\quad$ ⟵ $(m^*, t^*)$

$t^{**}$ ⟶

if $t^* = t^{**}$ return 1 else ret 0

$\Pr[\mathcal{A}^F = 1] = \Pr[F \text{ wins MAC } \Pi]$

$\Pr[\mathcal{A}^f = 1] = \Pr[F \text{ wins MAC } \tilde{\Pi}] = \frac{1}{2^n}$

$\left| \Pr[F \text{ wins MAC } \Pi] - \frac{1}{2^n} \right| \leq negl(n)$

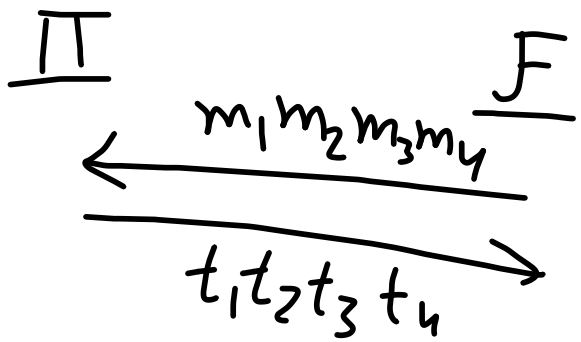
$\tilde{\Pi} \to \tilde{TG} \to \tilde{TG}_k(m) = f(m)$

rand function

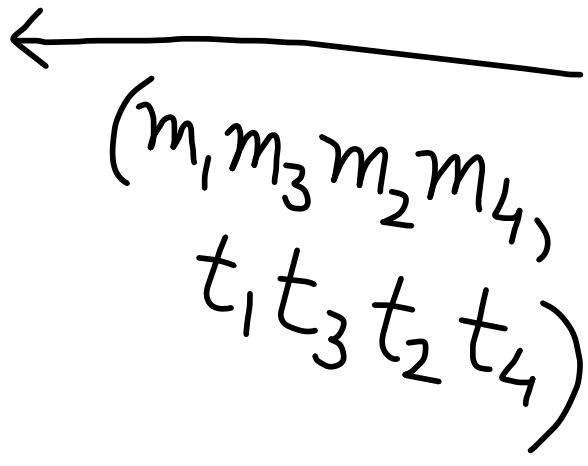
# Variable length MACs

$m_1\ m_2\ m_3\ m_4 \qquad m_i \in \{0,1\}^n$

$t_1\ t_2\ t_3\ t_4 \mid t_i = F_k(m_i)$

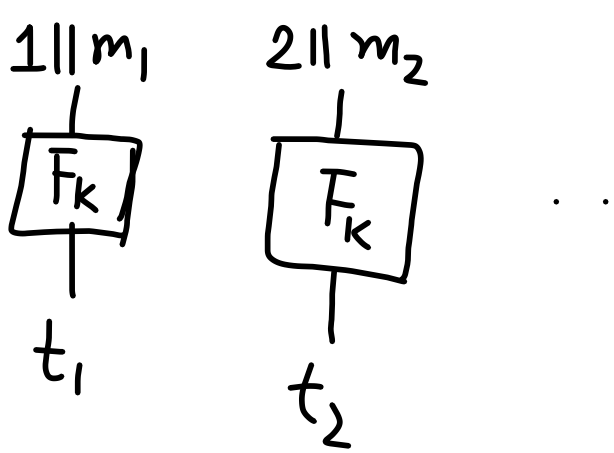


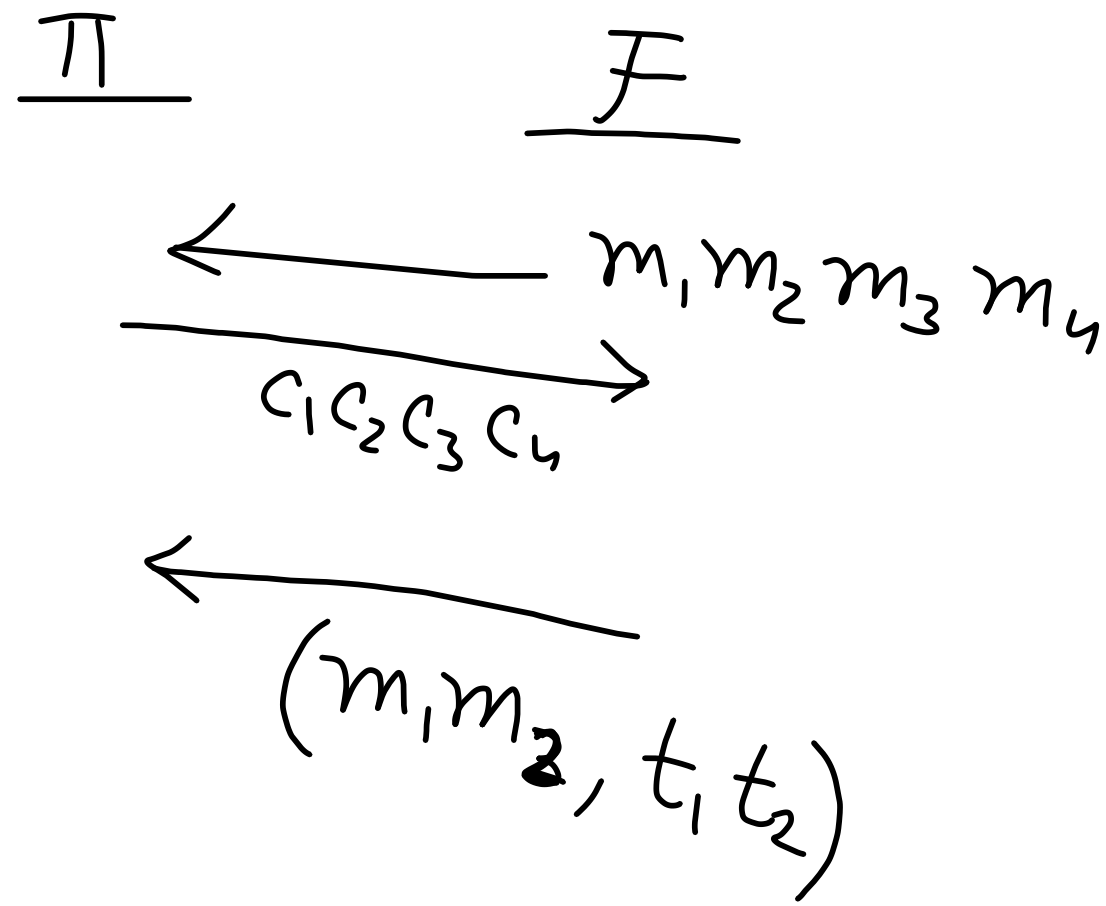

(Block - Rearrangement)



2) $m_1\ m_2\ m_3\ m_4$ $\quad m_i \in \{0,1\}^{n/2}$

$t_1\ t_2\ t_3\ t_4$ $\quad t_i = F_k(i \| m_i)$



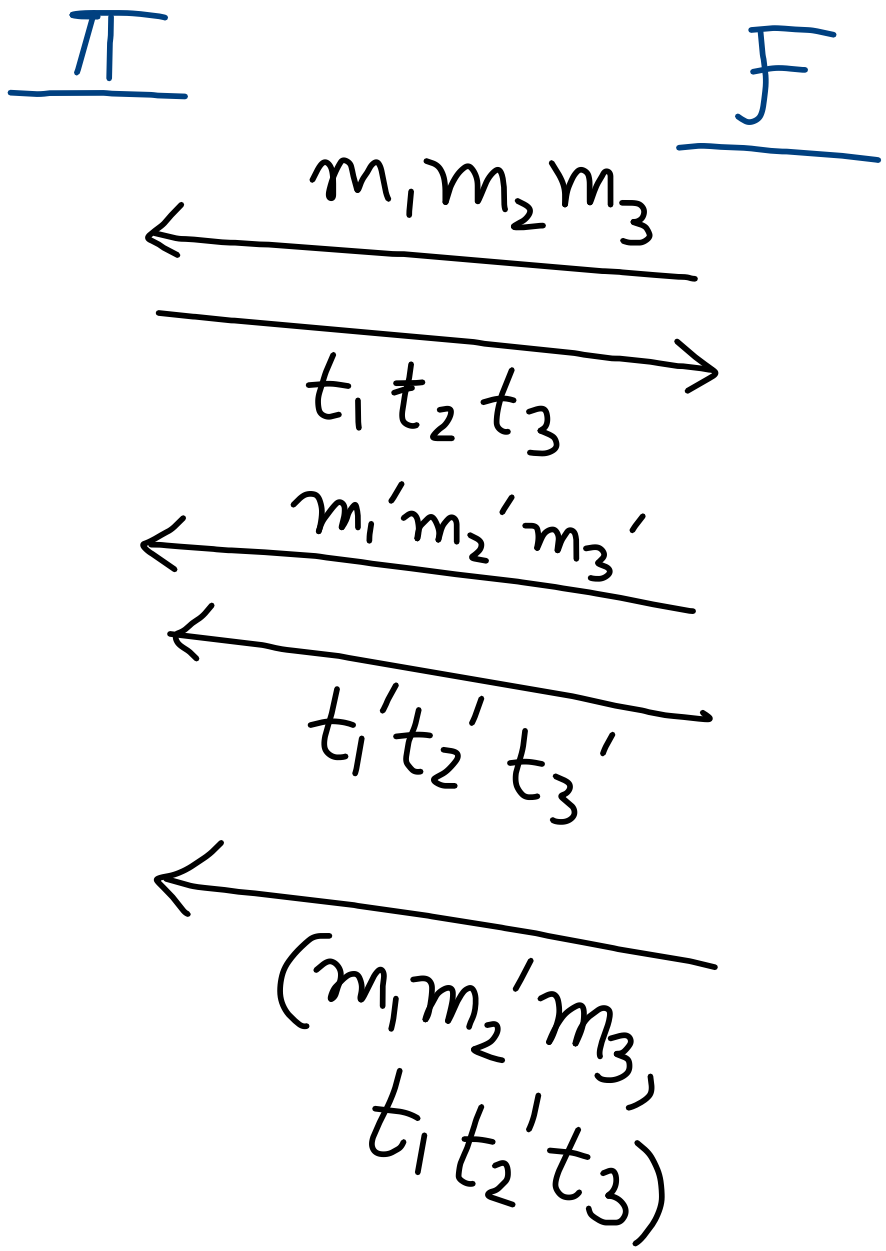



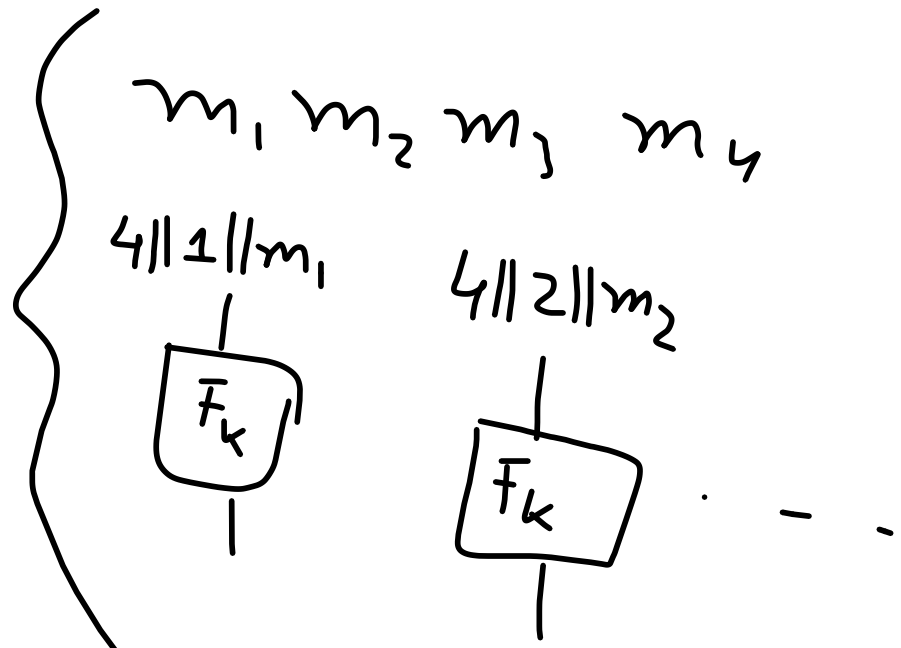

Truncation Attack

3) $t_i = F_k(\overline{\ell}^{n/3} \| \overline{i}^{n/3} \| \overline{m_i}^{n/3})$

$\{ m = m_1 m_2 \cdots m_\ell$

$m_i \in \{0,1\}^{n/3}$




Mix-and-Match

4) $t_i = F_k(r \| l \| i \| m_i)$ $\quad m_i \in \{0,1\}^{n/4}$

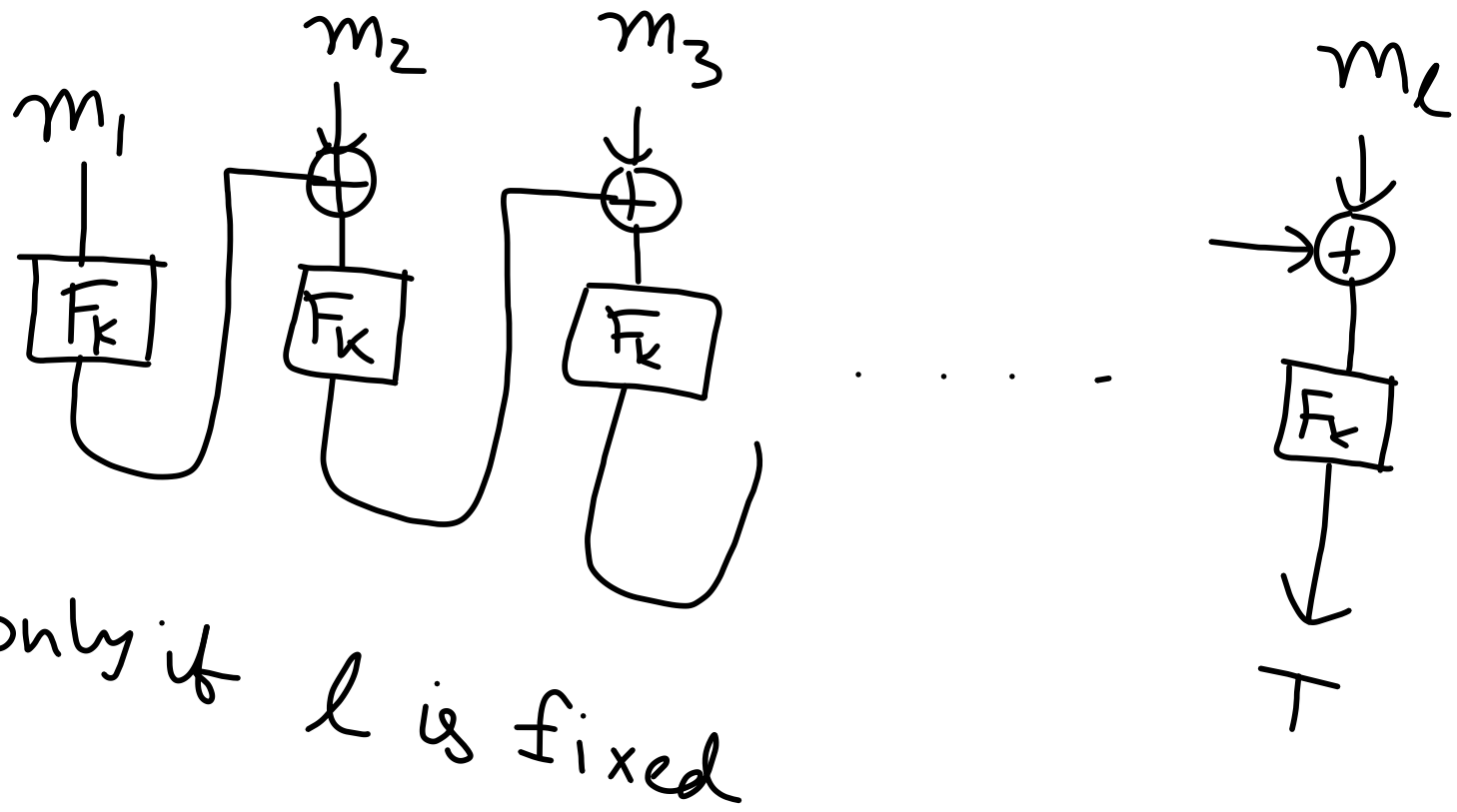
$BAD \Rightarrow r_i = r_j \mid i \neq j$

$$Pr[F_{succeeds}] \leq \frac{q^2}{2^{n/4}} + negl(n)$$

Secure MAC

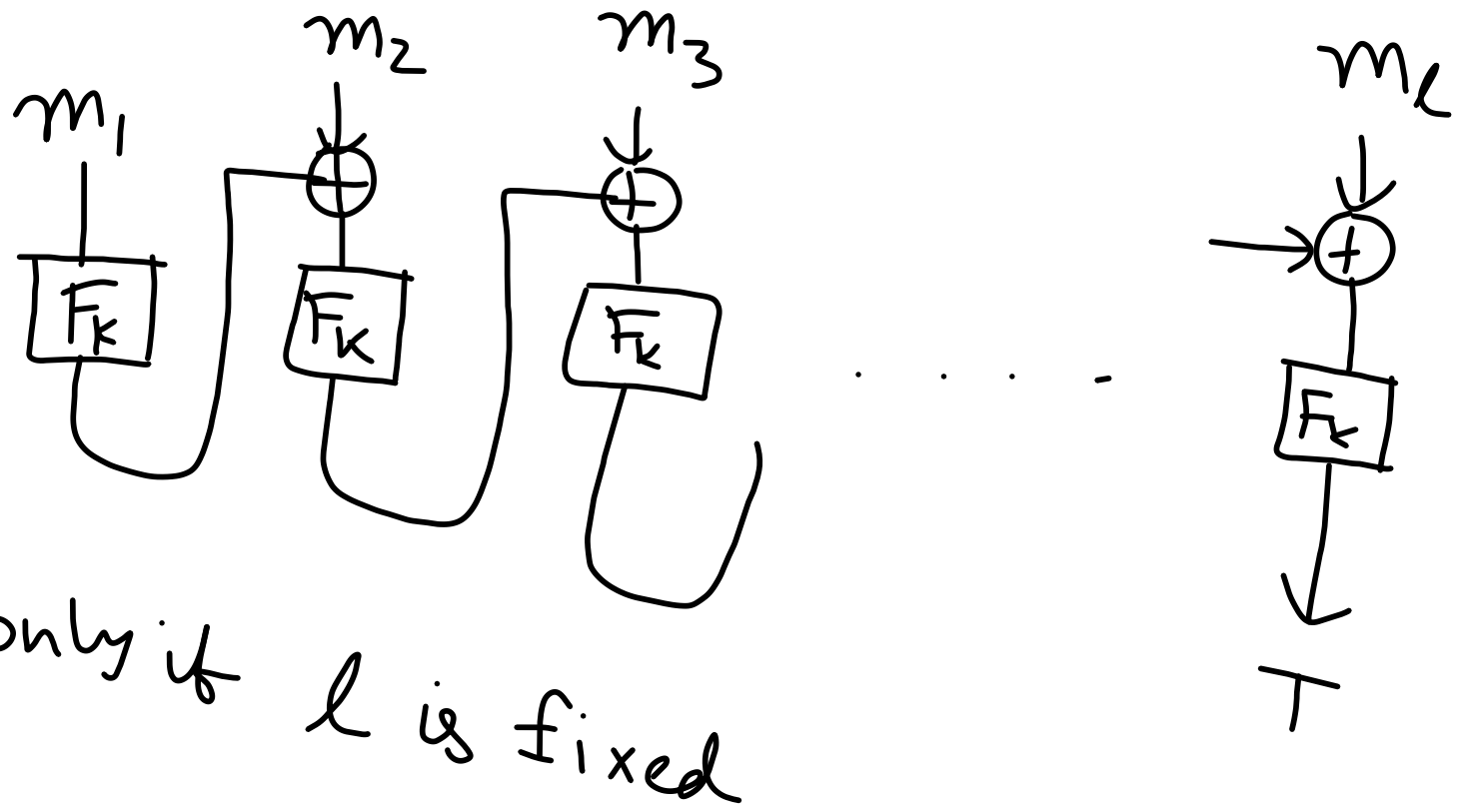
(Reduce the tag size)

# CBC-MAC (fixed length)




Works only if $\ell$ is fixed

# CBC-MAC (fixed length)




Works only if $\ell$ is fixed

# Forgery against CBC-MAC for variable length messages :-

① $m_1$ → $F_k$ → $t_1$

② $m_1$, $m_1 \oplus t_1$ → $F_k$, $F_k$ → $t_1$

Forge $(m_1 \| m_1 \oplus t_1 ; t_1)$

$m_1 \| m_1 \oplus t_1$ : $m^*$ ; $t_1$ : $t^*$

## ① Resisting the Attack

$M = m_1 \| \ldots \| m_\ell$

$M' = \langle \ell \rangle \| m_1 \| \ldots \| m_\ell$

↳ CBC-MAC on $(M')$

## ② Use two different keys

↳ Last key (finalization) is independent & random