# Lecture 2: Symmetric Key Cryptography

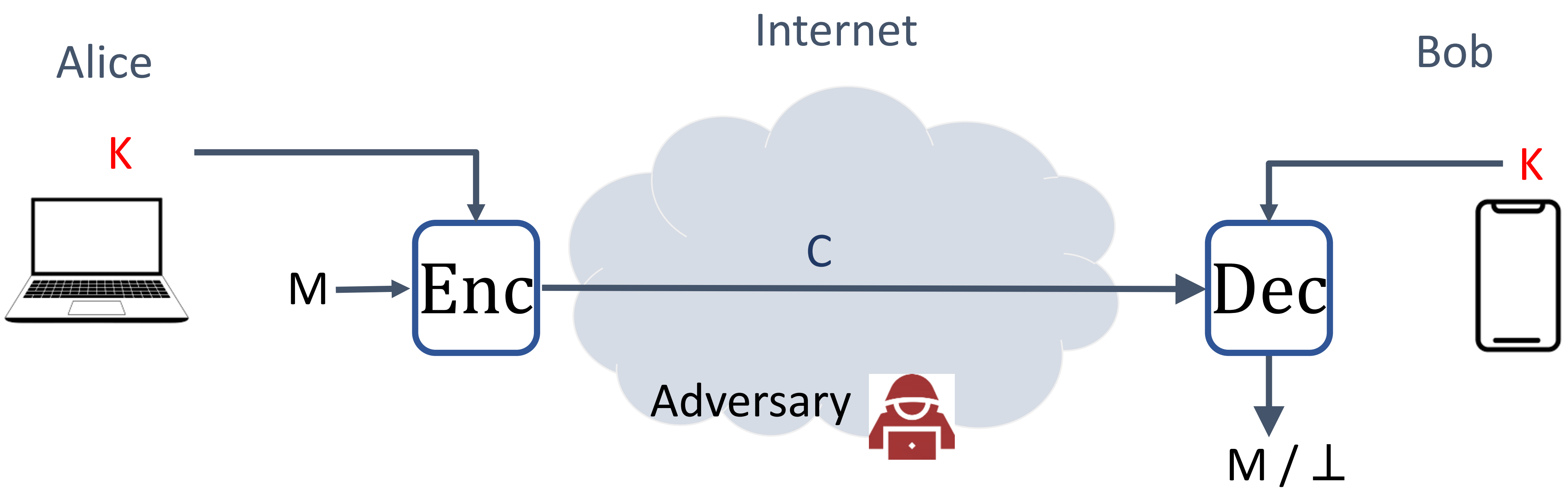
-COMP 6712 Advanced Security and Privacy

Haiyang Xue

haiyang.xue@polyu.edu.hk

2022/1/17

# Symmetric-key cryptography




Enc : encryption algorithm (public)

**K** : shared key between Alice and Bob

Dec : decryption algorithm (public)

# Outline of this lecture

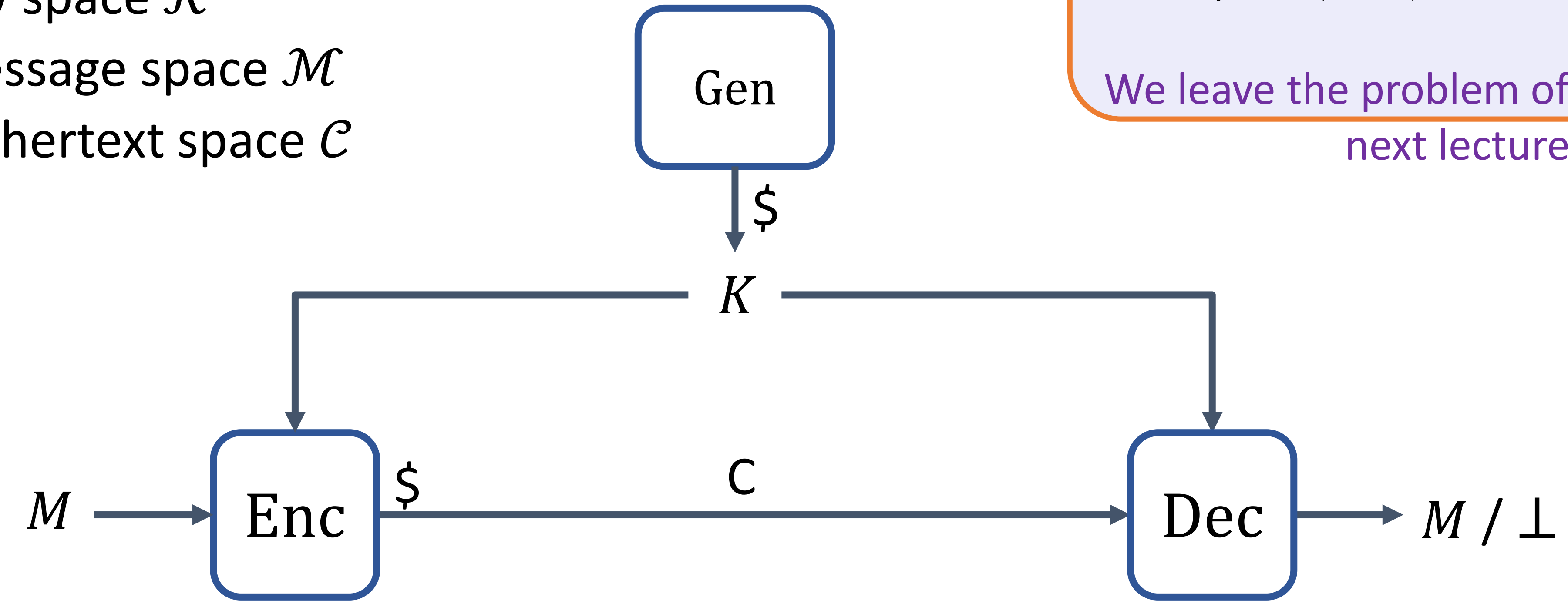
- Syntax and security of symmetric-key cryptography
- Perfect security and one-time pad
- Stream cipher, block cipher and MAC
- Hash function
- Constructions

# Syntax of symmetric encryption scheme

- A **symmetric encryption** $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ consists of three public algorithms:
- with
  - Key space $\mathcal{K}$
  - Message space $\mathcal{M}$
  - Ciphertext space $\mathcal{C}$

**Key Generation:** on input security parameter and randomness,
Outputs $(K, K)$ as the secret keys

We leave the problem of sending K to next lecture

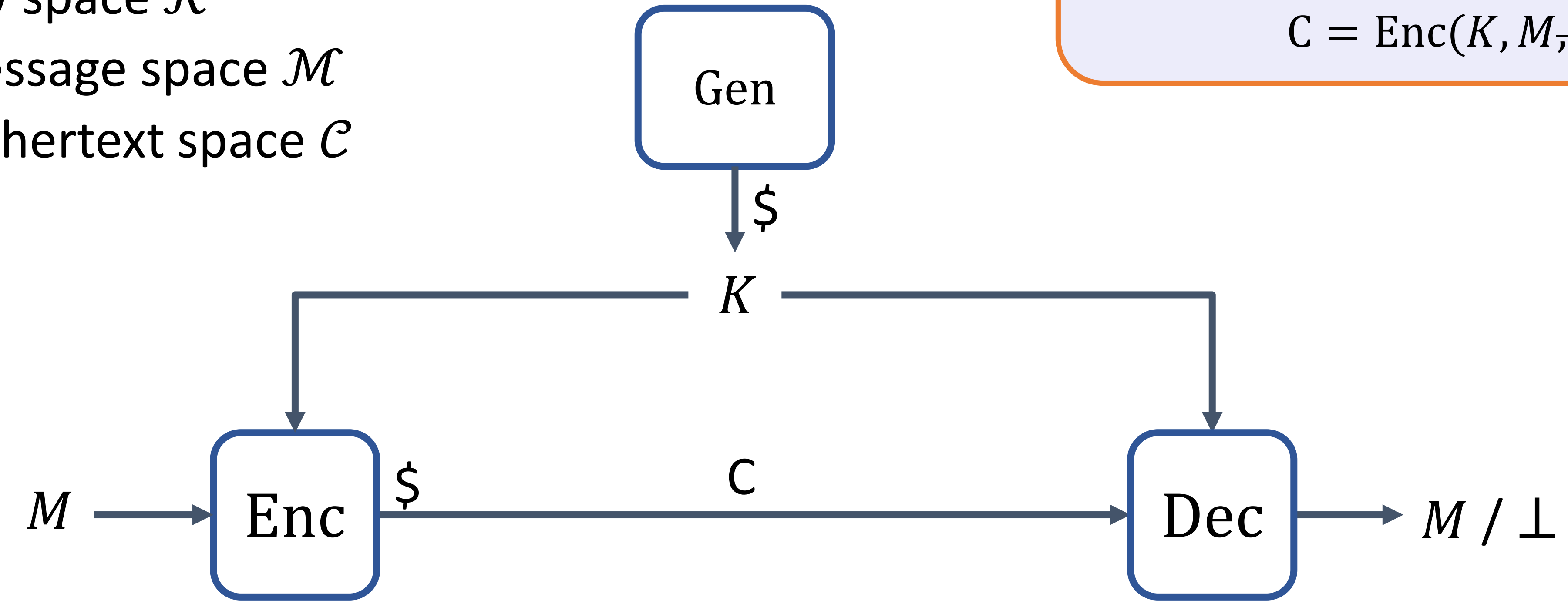


# Syntax of symmetric encryption scheme

- A **symmetric encryption** $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ consists of three public algorithms:
- with
  - Key space $\mathcal{K}$
  - Message space $\mathcal{M}$
  - Ciphertext space $\mathcal{C}$

**Encryption:** on input $M$ from $\mathcal{M}$ and $K$, (and randomness $r$)

$$\mathrm{C} = \mathrm{Enc}(K, M, \cancel{r})$$

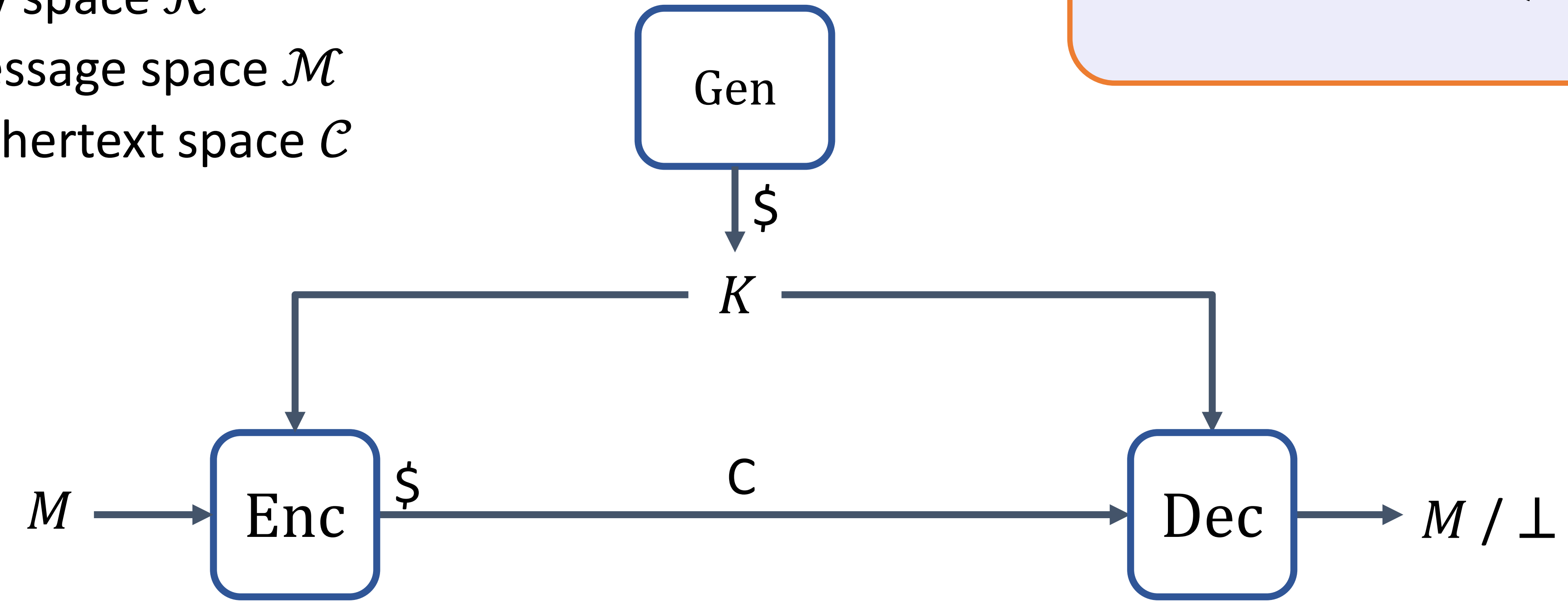


# Syntax of symmetric encryption scheme

- A **symmetric encryption** $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ consists of three public algorithms:
- with
  - Key space $\mathcal{K}$
  - Message space $\mathcal{M}$
  - Ciphertext space $\mathcal{C}$

**Decryption:** on input $C$ from $\mathcal{C}$ and $K$,

$$\mathrm{M}/\perp = \mathrm{Dec}(K, C)$$

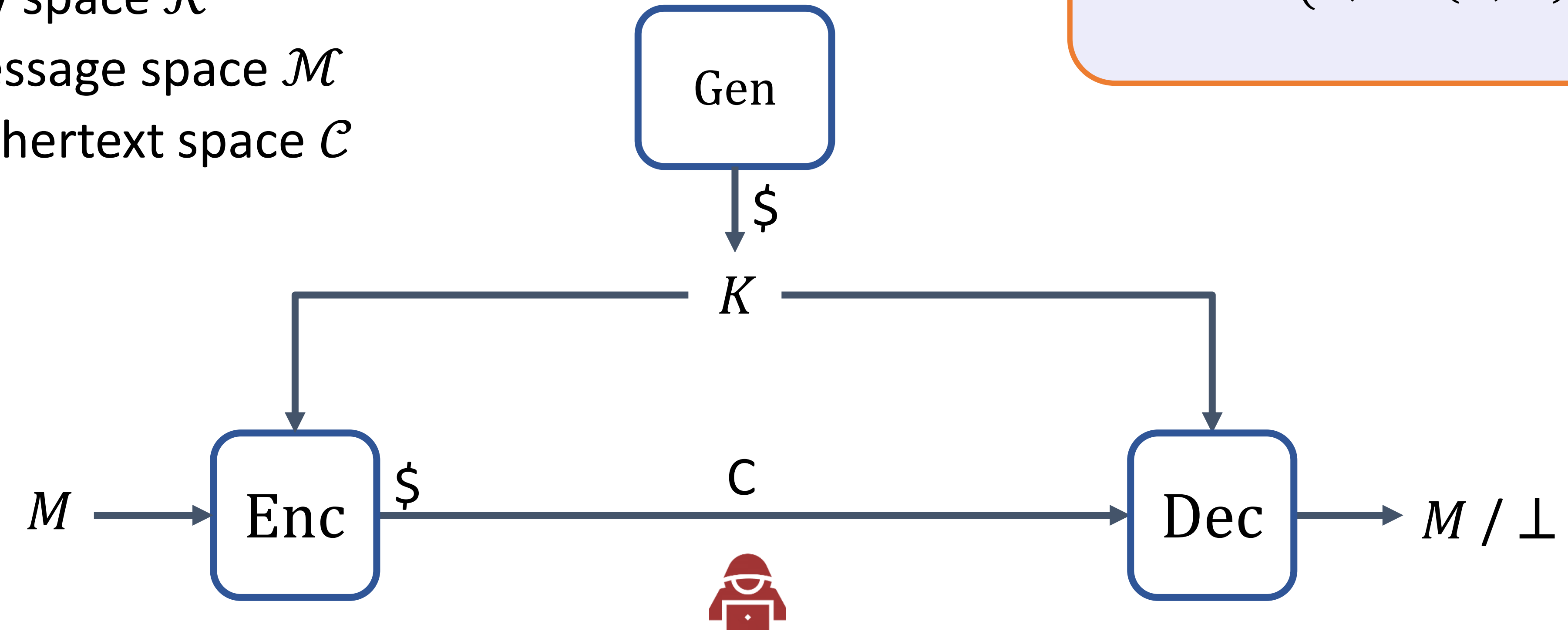


# Syntax of symmetric encryption scheme

- A **symmetric encryption** $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ consists of three public algorithms:
- with
  - Key space $\mathcal{K}$
  - Message space $\mathcal{M}$
  - Ciphertext space $\mathcal{C}$

**Correctness:** For all $K \leftarrow \mathrm{Gen}$ :

$$\mathrm{Dec}\big(K, \mathrm{Enc}(K, M)\big) = M$$




Is it possible to be secure against an adversary with unbounded computational power???

# Perfect security and one-time pad

- If an enc is secure against an adversary with unbounded computational power, it satisfies Perfect security

**Definition:** $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ is said to be **perfectly secret** if for every distribution over $\mathcal{M}$, any $m \in \mathcal{M}$, any $c \in \mathcal{C}$

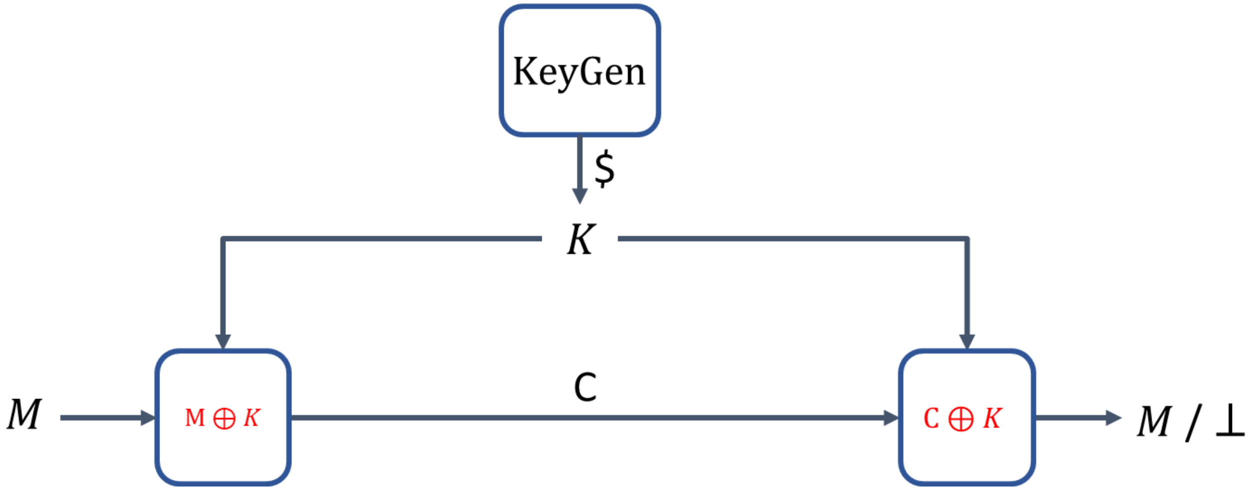
$$\Pr[M = m \mid C = c] = \Pr[M = \mathrm{m}]$$

with probability taken over the random choice $K \leftarrow \mathcal{K}$ and the random coins used by $\mathrm{Enc}$ (if any))

- The ciphertext gives nothing about the message (even for unbounded adversary)

# Is perfect security possible? One-time Pad

- $\mathcal{K} = \{0,1\}^n$
- $\mathcal{M} = \{0,1\}^n$
- $\mathcal{C} = \{0,1\}^n$




$Gen:$

$K \leftarrow \{0,1\}^n$

$Enc: \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C}$

$Enc(K, M) = M \oplus K$

$Dec: \mathcal{K} \times \mathcal{C} \rightarrow \mathcal{M}$

$Dec(K, C) = C \oplus K$

# Is perfect security possible? One-time Pad

- $\mathcal{K} = \{0,1\}^n$
- $\mathcal{M} = \{0,1\}^n$
- $\mathcal{C} = \{0,1\}^n$

$Gen:$

$K \leftarrow \{0,1\}^n$

1110001101

$Enc: \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C}$

$Enc(K, M) = M \oplus K$

$$
\begin{array}{ll}
\phantom{\oplus}\ 0101100100 & M \\
\oplus\ 1110001101 & K \\
\hline
=\ 1011101001 & C
\end{array}
$$

$Dec: \mathcal{K} \times \mathcal{C} \rightarrow \mathcal{M}$

$Dec(K, C) = C \oplus K$

$$
\begin{array}{ll}
\phantom{\oplus}\ 1011101001 & C \\
\oplus\ 1110001101 & K \\
\hline
=\ 0101100100 & M
\end{array}
$$

# One-time Pad

**Theorem:** The One-time Pad encryption scheme has perfect security

- **Have to show:** $\Pr[M = m \mid C = c] = \Pr[M = m]$

$$\Pr[C = c \mid M = m] = \Pr[m \oplus K = c] = \Pr[K = m \oplus c] = \frac{1}{2^n}$$

$$\Pr[C = c] = \sum_{m \in \mathcal{M}} \Pr[C = c \mid M = m] \Pr[M = m] = \frac{1}{2^n} \sum_{m \in \mathcal{M}} \Pr[M = m] = \frac{1}{2^n}$$

$$\Pr[M = m \mid C = c] = \frac{\Pr[C = c | M = m] \Pr[M = m]}{\Pr[C = c]} = \frac{\frac{1}{2^n} \Pr[M = m]}{\frac{1}{2^n}}$$

# Limitation

- But $|\mathcal{K} = \{0,1\}^n| = |\mathcal{M} = \{0,1\}^n|$?
- If we find a way to deliver $K$, why not deliver $M$ directly?

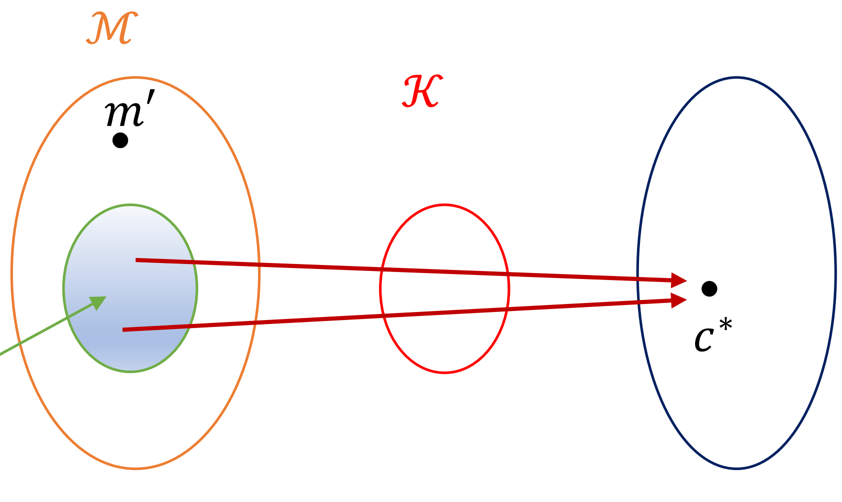
**Theorem:** If $\Pi$ is a perfectly secret enc with key space $\mathcal{K}$ and message space $\mathcal{M}$

$$|\mathcal{K}| \geq |\mathcal{M}|$$

**We show:** if $|\mathcal{K}| < |\mathcal{M}|$, $\Pi$ can not be perfectly secret

We have $|M(c^*)| \leq |\mathcal{K}| < |\mathcal{M}|$,

$\Pr[M = m'] \neq 0$ , while $\Pr[M = m' \mid C = c^*] = 0$




$M(c^*) = \{m | m = Dec(K, c^*), K \in \mathcal{K}\}$

# A short summary

- perfect security against the unbounded adversary
- could be achieved via the one-time pad
- Inherent limitation, key space $\geq$ message space
- How to break the limitation?

# Break the limitation

- Aim low
- ~~Unbounded adversary~~

- Guarantee against efficient adversaries that run for some feasible amount of time. (ex. probabilistic polynomial time (PPT))

- Adversaries can potentially succeed with a small probability

# small probability- negligible function

**Definition:** A positive function $f$ is said to be **negligible** if for every positive polynomial $p$, and sufficiently large $n$

$$f(n) \leq \frac{1}{p(n)}.$$

- Ex

$$2^{-n}$$

$$2^{-\sqrt{n}}$$

$$\frac{1}{n^{1000}}??$$

**Theorem:** for every positive polynomial $q$, if $f$ is **negligible**, so does $q(n) \cdot f(n)$.

# Necessary of PPT and negligible

- probability polynomial time
  - If $|\mathcal{K}| < |\mathcal{M}|$, ciphertext must leak some information to UNBUOUNDed adversary

- Negligible success probability
  - Adversary runs in constant time can win with probability $\frac{1}{|\mathcal{K}|}$

# Computational security

**Definition:** A scheme is $(t, \varepsilon)$-secure if any adversary running for a time at most $t$ succeeds in breaking the scheme with probability at most $\varepsilon$.

**Definition:** A scheme $\Pi$ is said to be **computationally secure** if any PPT adversary succeeds in breaking the scheme with negligible probability.

# IND-eavesdropper

**$\mathbf{Exp}_{\Pi}^{\text{ind-eav}}(A)$**

$b \xleftarrow{\$} \{0,1\}$

$K \xleftarrow{\$} \Pi.\mathrm{Gen}$

$M_0, M_1 \leftarrow A()$ // find stage

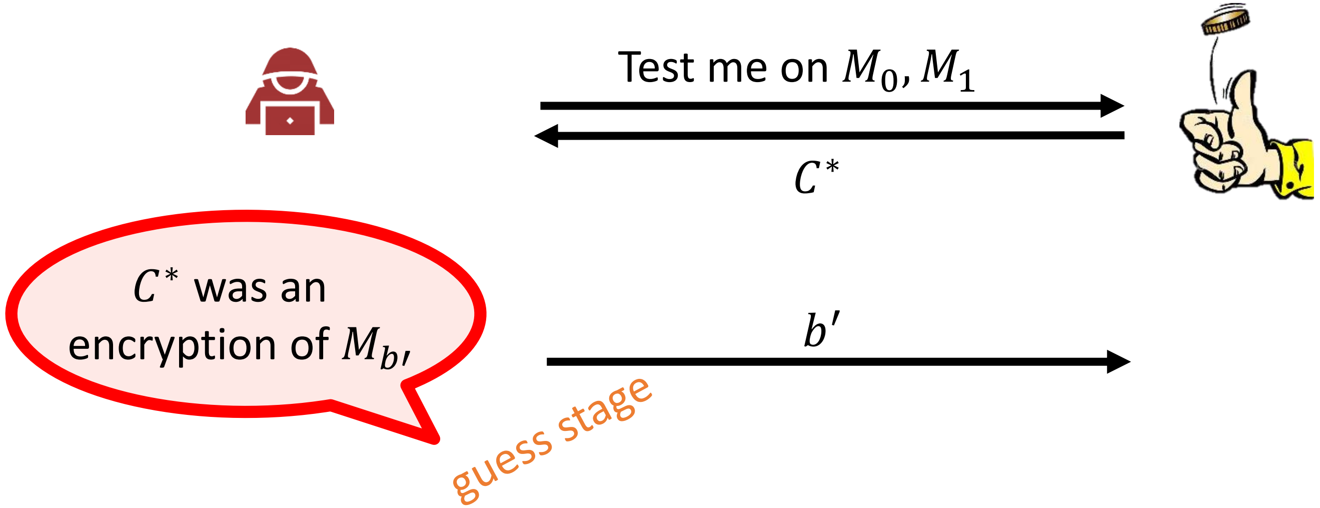
  **if** $|M_0| \neq |M_1|$ **then**

  **return** $\perp$

$C^* \leftarrow \Pi.\mathrm{Enc}(K, M_b)$

$b' \leftarrow A\,(C^*)$ // guess stage

**return** $b' \stackrel{?}{=} b$




**Definition:** The **IND-eav-advantage** of an adversary $A$ is

$$\mathbf{Adv}_{\Pi}^{\text{ind-eav}}(A) = \left|\Pr\left[\mathbf{Exp}_{\Pi}^{\text{ind-eav}}(A) \Rightarrow 1\right] - 1/2\right|$$

# Construction of IND-eavesdropper secure enc

- We could construct a secure enc from PRG
- PRG is generally a function to extends $k$ random bits to $k + l$ pseudo-random bits

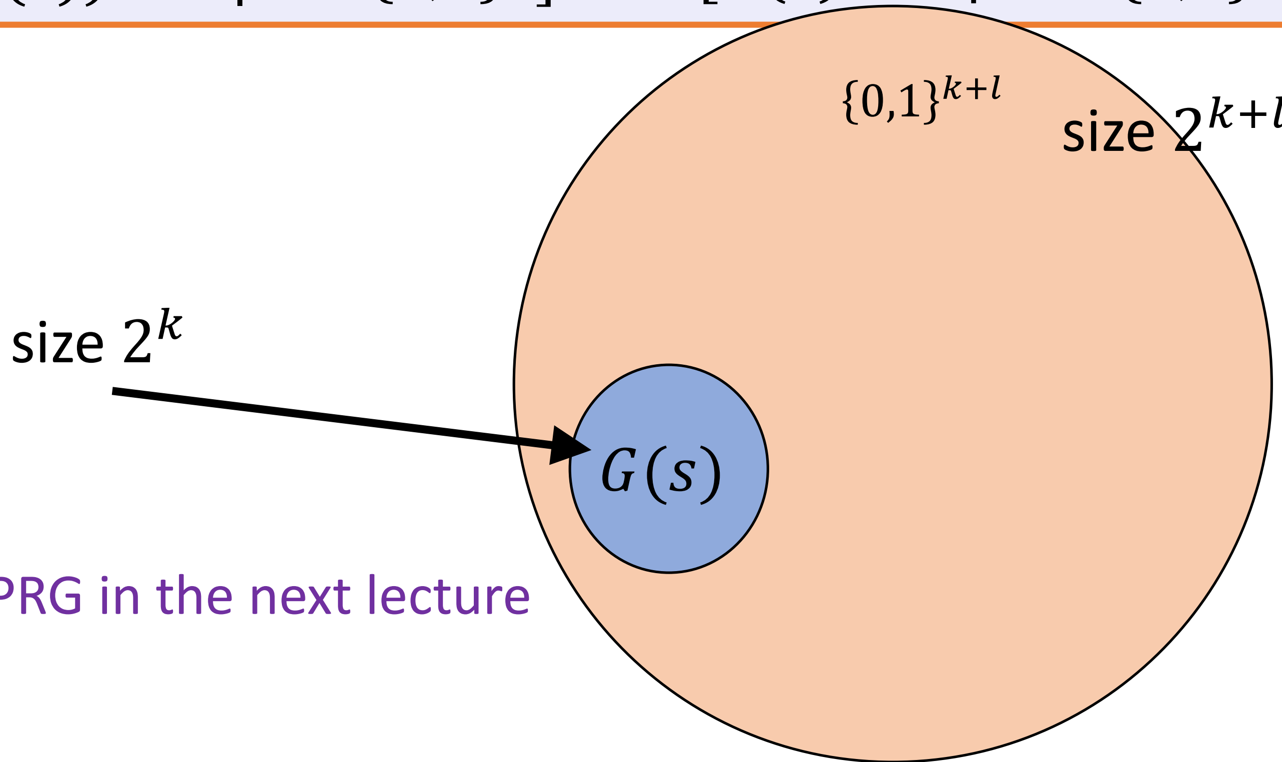
# pseudo-random generator (PRG)

**Definition:** A **pseudorandom random generator (PRG)** is a function

$$G : \{0,1\}^k \to \{0,1\}^{k+l}$$

Such that

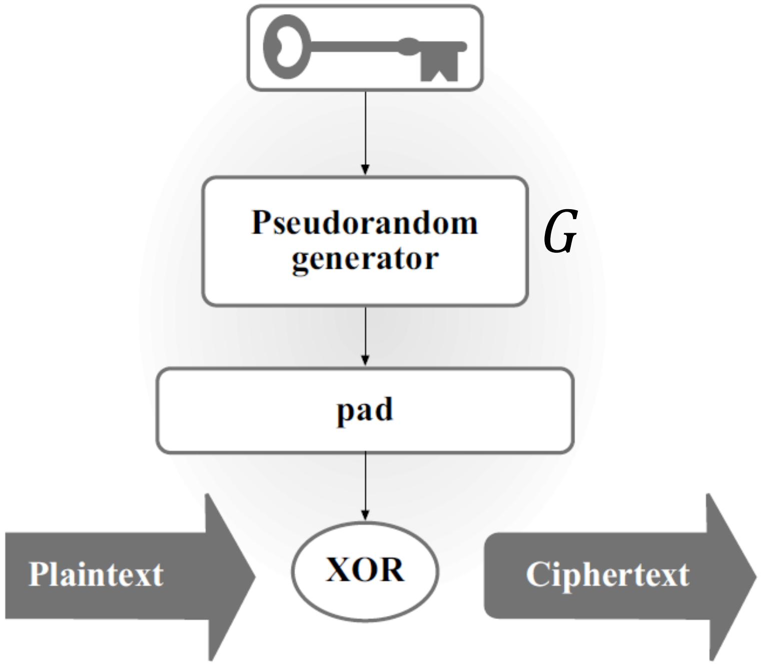
- $0 < l < poly\ (k)$
- For any PPT A, $\Pr[A(G(s)) = 1 | s \leftarrow \{0,1\}^k] - \Pr[A(r) = 1 | \, r \leftarrow \{0,1\}^{k+l}] < negl$




- We leave the construction of PRG in the next lecture

# IND-eavesdropper Enc (with fix length) from PRG

- Let $G : \{0,1\}^k \rightarrow \{0,1\}^{k+l}$ be a PRG
- $\Pi 1.\text{Gen}$: $K \leftarrow \{0,1\}^k$
- $\Pi 1.\text{Enc}(\text{K}, \text{M})$: $C = G(K) \oplus M$
- $\Pi 1.\text{Dec}(\text{K}, \text{C})$: $M = G(K) \oplus C$



# PROOF idea: IND-eavesdropper

$\mathbf{Exp}_{\Pi 1}^{\text{ind-eav}}(A)$

$b \xleftarrow{\$} \{0,1\}$

$K \xleftarrow{\$} \Pi 1.\mathrm{Gen}$

$M_0, M_1 \leftarrow A()$ // find stage

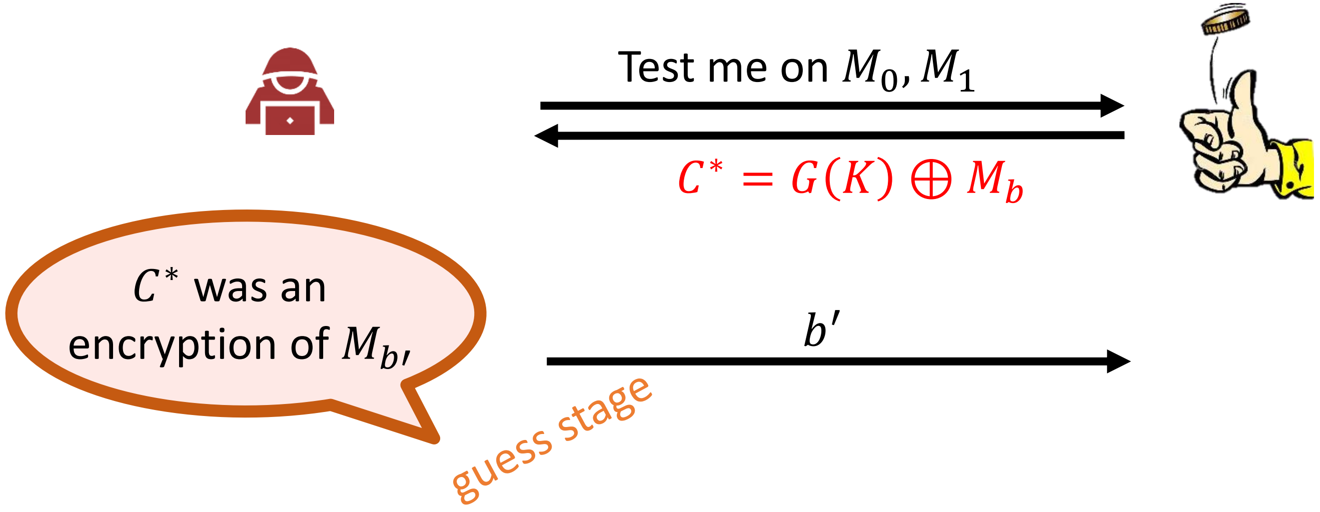
  **if** $|M_0| \neq |M_1|$ **then**

  **return** $\perp$

$C^* \leftarrow G(K) \oplus M_b$

$b' \leftarrow A\,(C^*)$ // guess stage

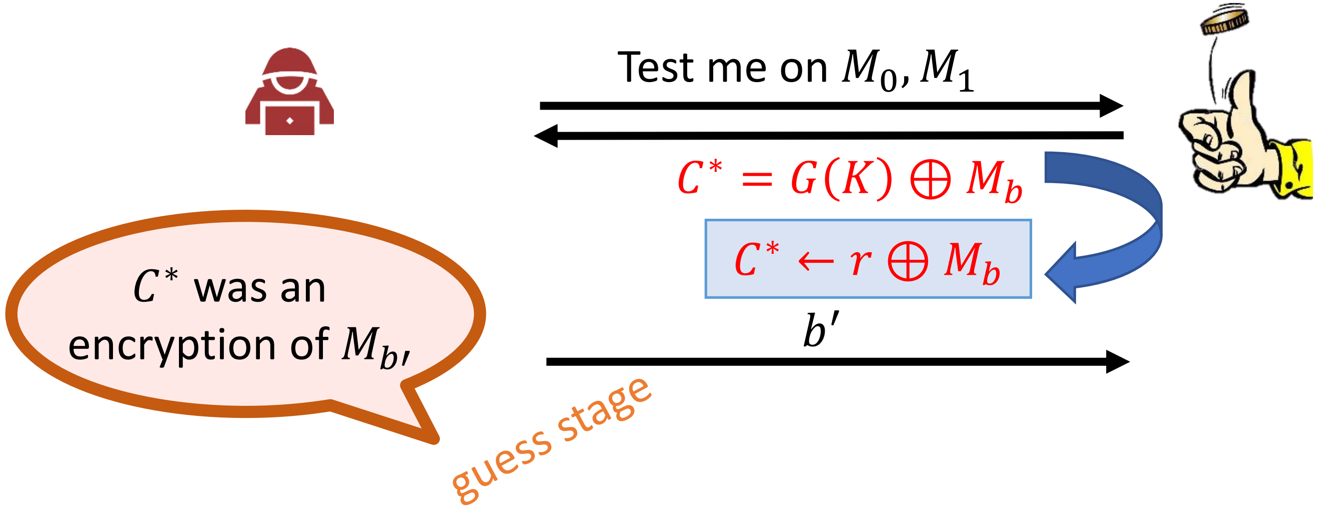
**return** $b' \stackrel{?}{=} b$



# PROOF idea: IND-eavesdropper

$\mathbf{Exp}_{\Pi 1}^{\text{ind-eav}}(A)$

$b \xleftarrow{\$} \{0,1\}$
$K \xleftarrow{\$} \Pi 1.\mathrm{Gen}$
$M_0, M_1 \leftarrow A()$ // find stage
  **if** $|M_0| \neq |M_1|$ **then**
  **return** $\perp$
$C^* \leftarrow G(K) \oplus M_b$ → $C^* \leftarrow r \oplus M_b$
$b' \leftarrow A\,(C^*)$ // guess stage
**return** $b' \stackrel{?}{=} b$




Now, this is an **one-time pad** and the **IND-eav-advantage** of an adversary $A$ is

$$\mathbf{Adv}_{\Pi 1}^{\text{ind-eav}}(A) = 0$$

# PROOF idea: IND-eavesdropper

**$\mathbf{Exp}_{\Pi 1}^{\text{ind-eav}}(A)$**

$b \overset{\$}{\leftarrow} \{0,1\}$

$K \overset{\$}{\leftarrow} \Pi 1.\mathrm{Gen}$

$M_0, M_1 \leftarrow A()$ // find stage

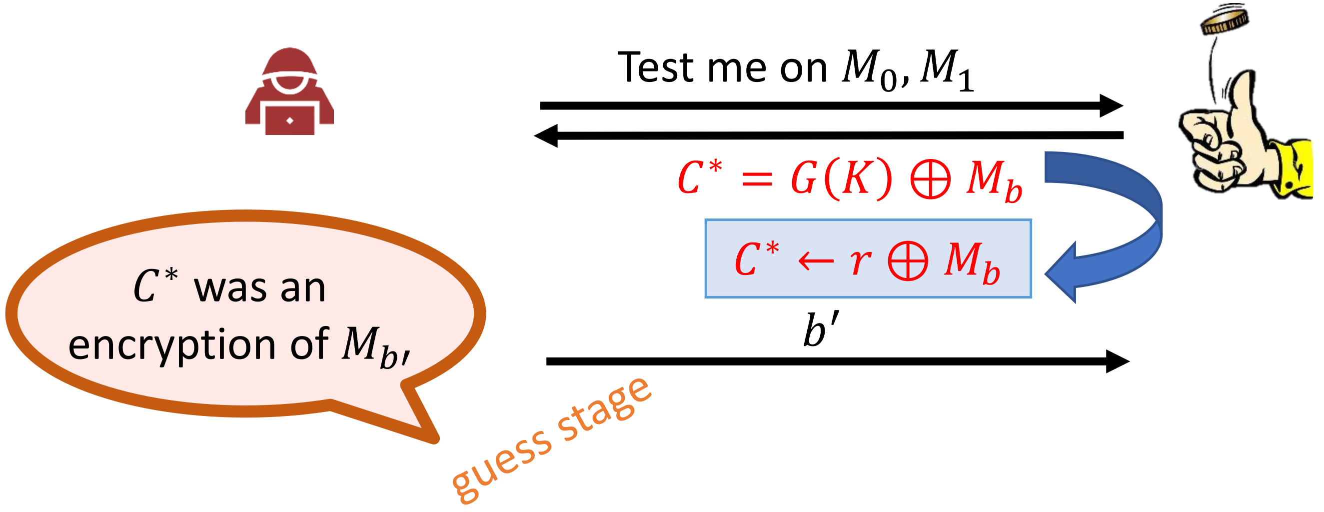
**if** $|M_0| \neq |M_1|$ **then**

**return** $\perp$

$C^* \leftarrow G(K) \oplus M_b$ → $C^* \leftarrow r \oplus M_b$

$b' \leftarrow A(C^*)$ // guess stage

**return** $b' \overset{?}{=} b$

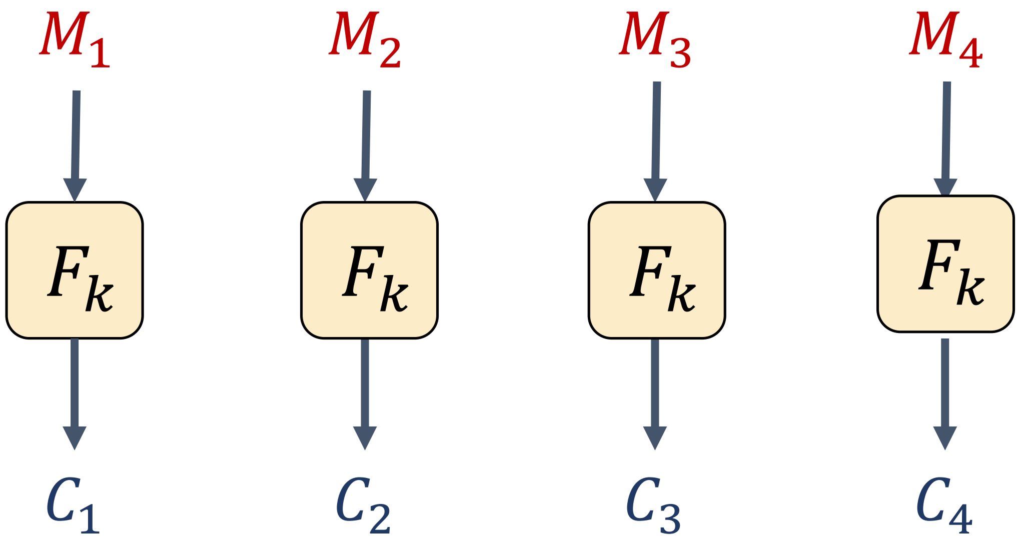



Any PPT adversary can not find the switch, since $G$ is a PRG

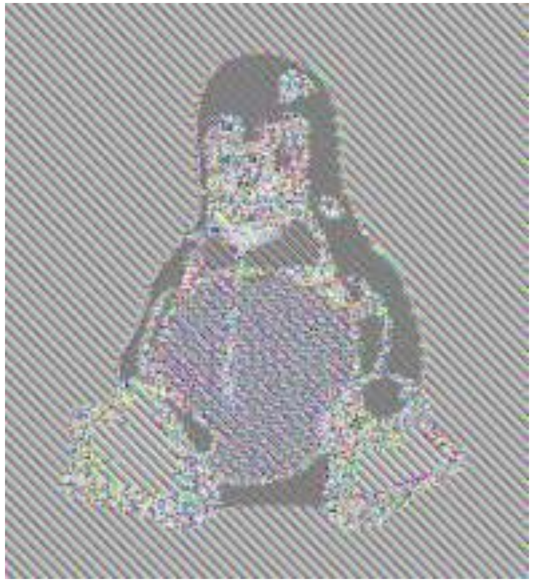
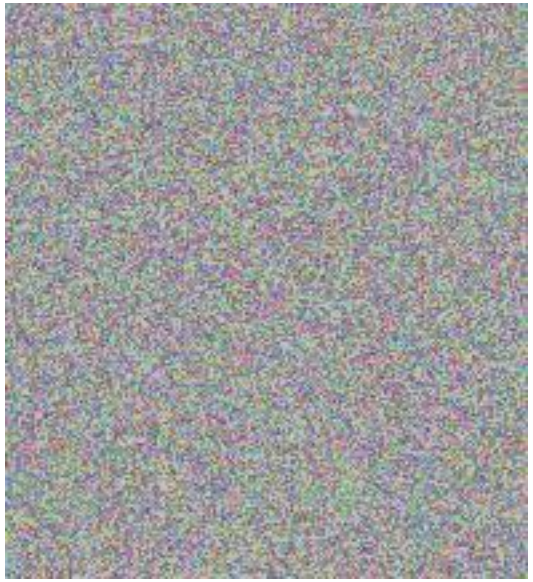
# Electronic Code Book (ECB) mode (for longer message)

- Given a block cipher $\Pi 1. F_k \colon \{0,1\}^n \to \{0,1\}^n$

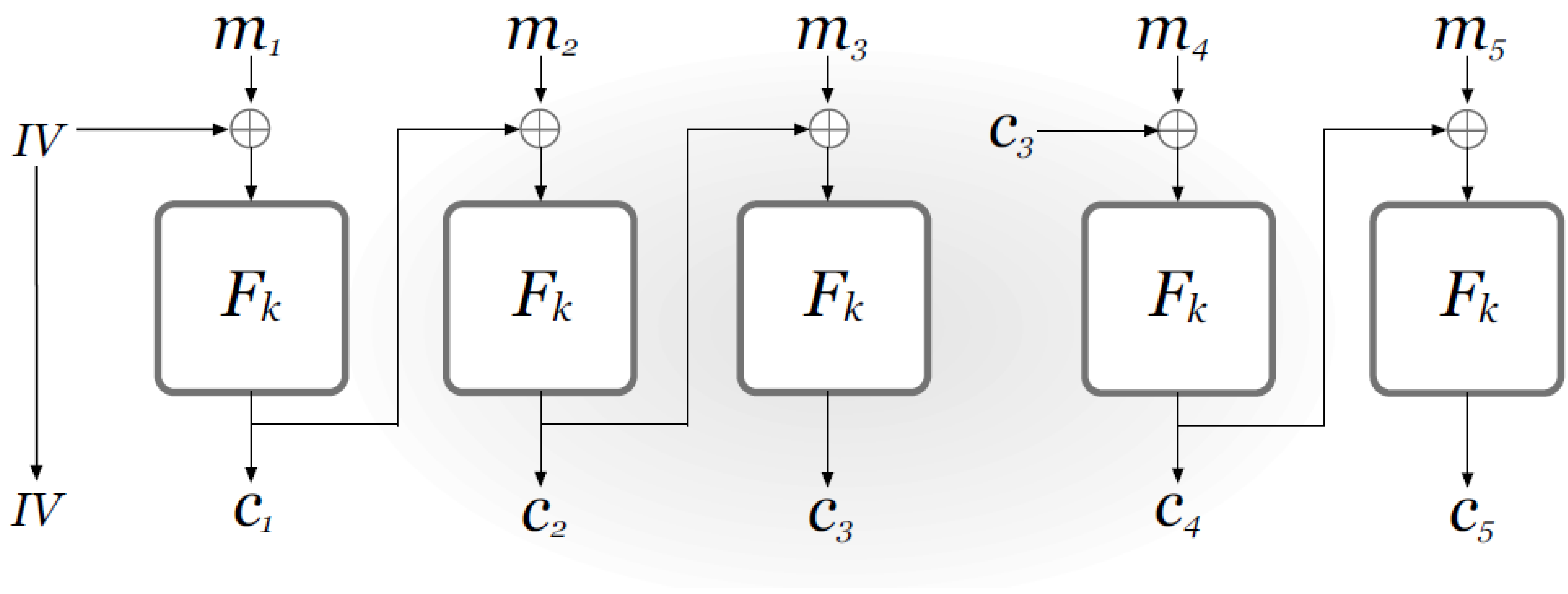
- $\mathrm{ECB}[F_k] = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$



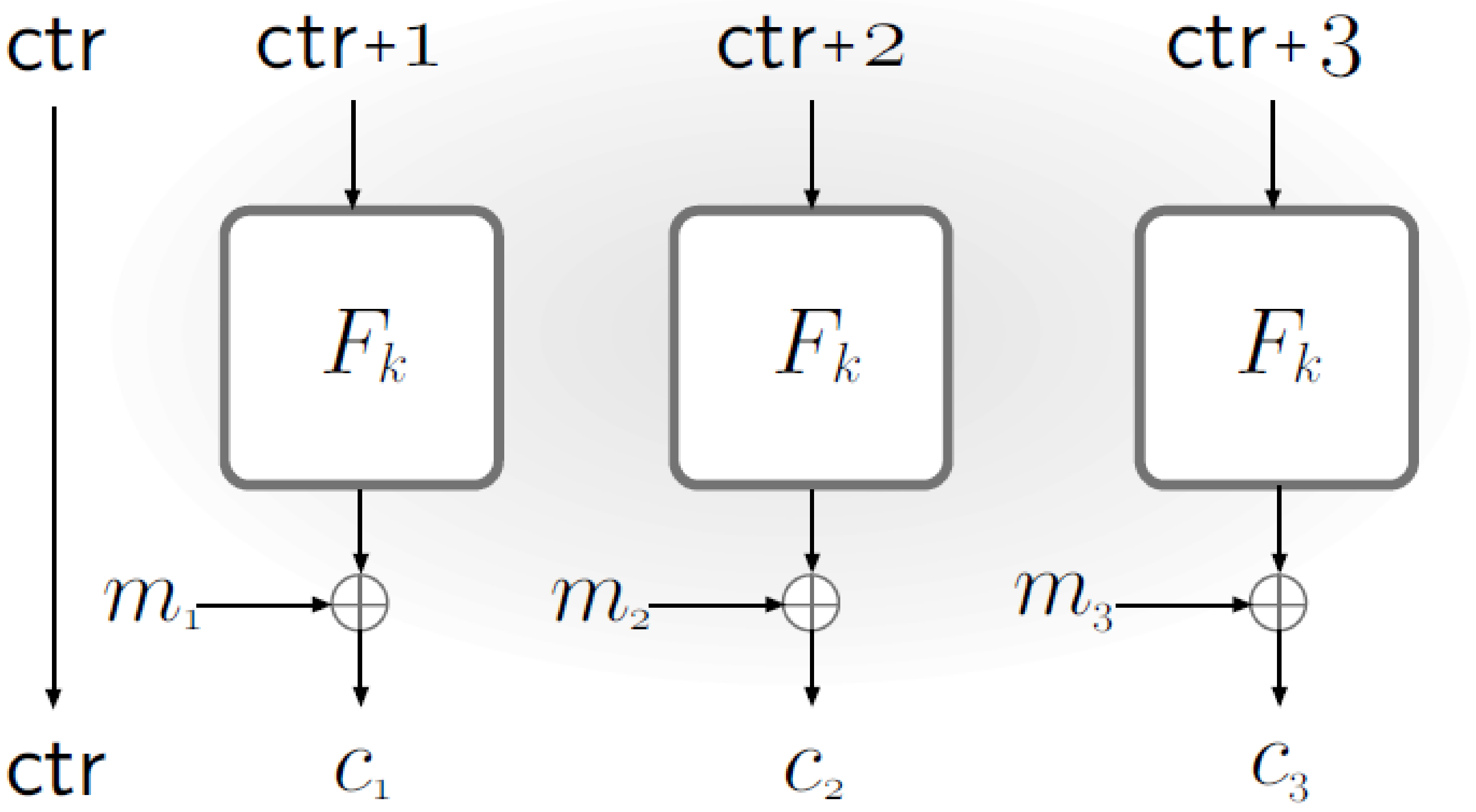
# Weakness of ECB

| Plaintext | ECB encrypted | Properly encrypted |
| --- | --- | --- |
|  |  |  |

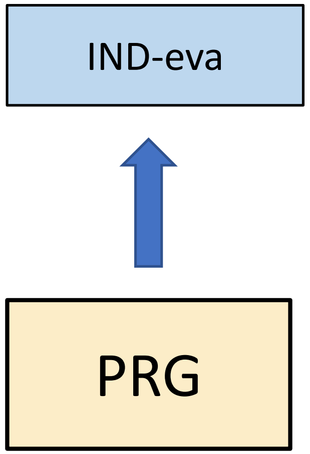
# Cipher Block Chaining (CBC) mode



# Counter (CTR) mode



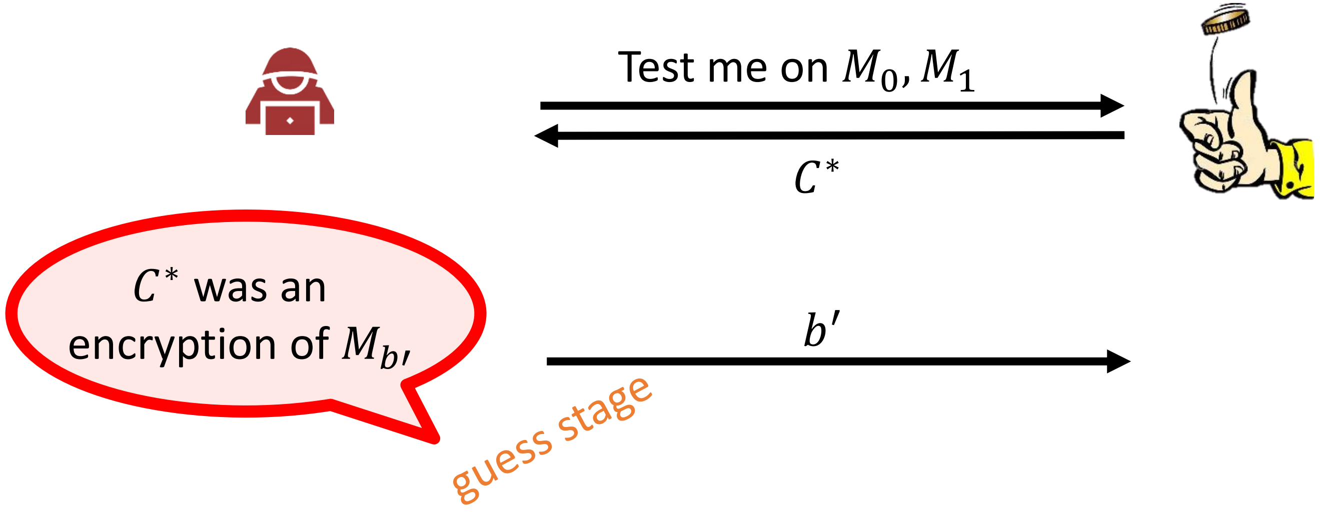
# A short summary



# A short summary

- With aim of computational security, we can encrypt a long message with a short key
- With PRG, we could build IND-eavesdropper Enc
- We can further encrypt a longer message by splitting the message in blocks. It may operate in several models, EBC, CBC, CTR etc.
- IND-eavesdropper is a very weak security aim.

# IND-eavesdropper is weak




**Definition:** The **IND-eav-advantage** of an adversary $A$ is

$$\mathbf{Adv}_{\Pi}^{\text{ind-eav}}(A) = \left|\Pr\left[\mathbf{Exp}_{\Pi}^{\text{ind-eav}}(A) \Rightarrow 1\right] - 1/2\right|$$

# Strong Security: IND-CPA

- In World War II
- British placed naval mines at certain locations, knowing that the Germans—when finding those mines—would encrypt the locations and send them back to Germany

- C = Enc (location of mines)

https://en.wikipedia.org/wiki/Naval_mine

An adversary may have the capability to choose a message and get the ciphertext

# IND-CPA (choose plaintext attack)

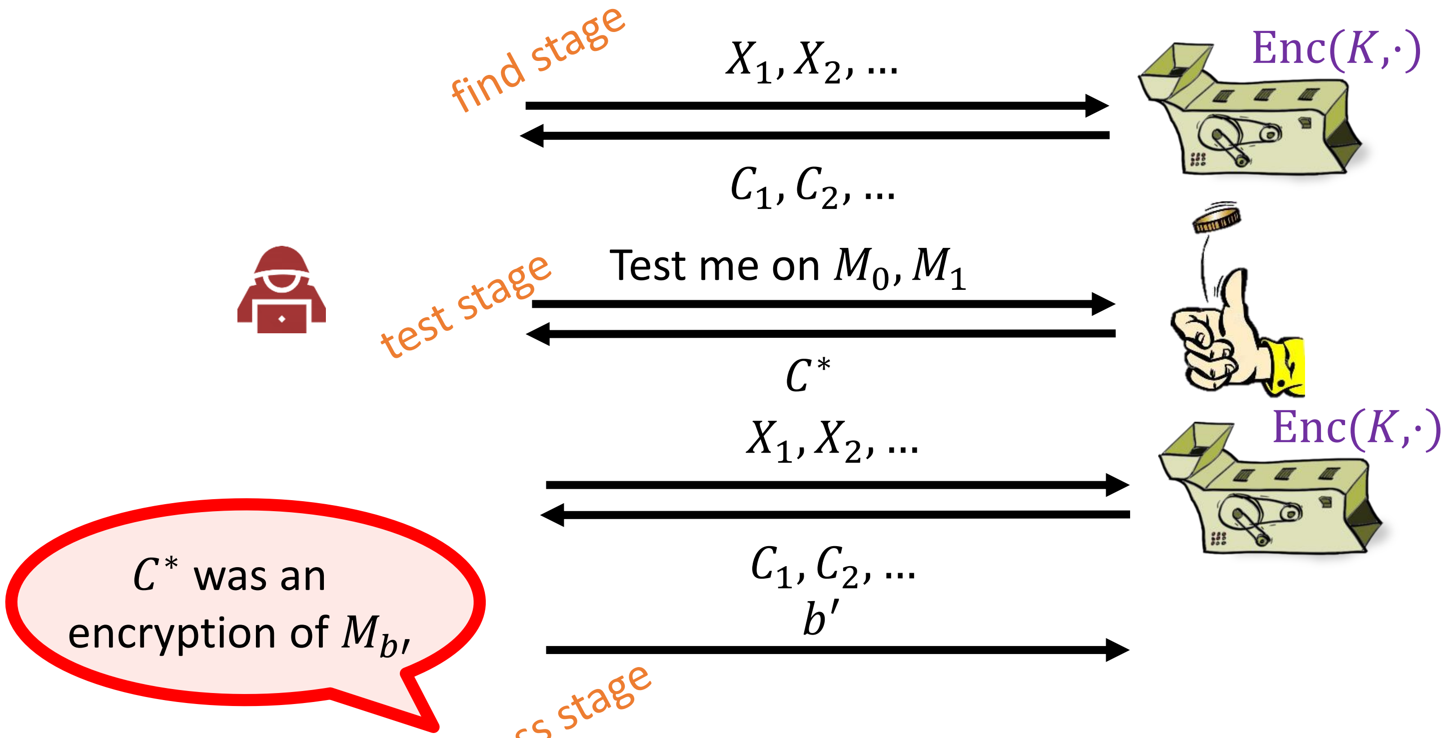
$\mathbf{Exp}_{\Pi}^{\text{ind-cpa}}(A)$

$$
\begin{array}{l}
b \overset{\$}{\leftarrow} \{0,1\} \\
K \overset{\$}{\leftarrow} \Pi.\mathrm{Gen} \\
M_0, M_1 \leftarrow A^{Enc(K,\cdot)} \quad \text{// find stage} \\
\quad \mathbf{if}\ |M_0| \neq |M_1|\ \mathbf{then} \\
\quad \mathbf{return}\ \perp \\
C^* \leftarrow \Pi.\mathrm{Enc}(K, M_b) \quad \text{// test stage} \\
b' \leftarrow A^{Enc(K,\cdot)}(C^*) \quad \text{// guess stage} \\
\mathbf{return}\ b' \overset{?}{=} b
\end{array}
$$

$Enc(K, M)$

---

$\mathbf{return}\ \Pi.\mathrm{Enc}(K, M)$




**Definition:** The **IND-CPA-advantage** of an adversary $A$ is

$$\mathbf{Adv}_{\Pi}^{\text{ind-cpa}}(A) = \left|\Pr\left[\mathbf{Exp}_{\Pi}^{\text{ind-cpa}}(A) \Rightarrow 1\right] - 1/2\right|$$

# IND-CPA Insecurity of $\Pi 1$

**Adversary $A$**

1. Query $C \leftarrow \Pi 1.\mathrm{Enc}(K, 0^{128})$ in the find stage
2. Submit $M_0 = 0^{128}$ and $M_1 = 1^{128}$
3. Receive challenge $C^*$
4. if $C^* = C$ output 0
5. else, output 1

Actually, this attack works for any DETERMINISTIC Enc

# Construction of IND-CPA secure enc

- We could construct an IND-CPA secure enc from PRF
- PRF generalizes the notion of PRG
- instead of considering “random-looking” strings we consider “random-looking” functions

# pseudorandom function (PRF)

**Definition:** A **pseudorandom function (PRF)** is a function

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

satisfying security in next page

- $k, in, out$ are called **key-length**, **input-length**, and **output-length** of $F$

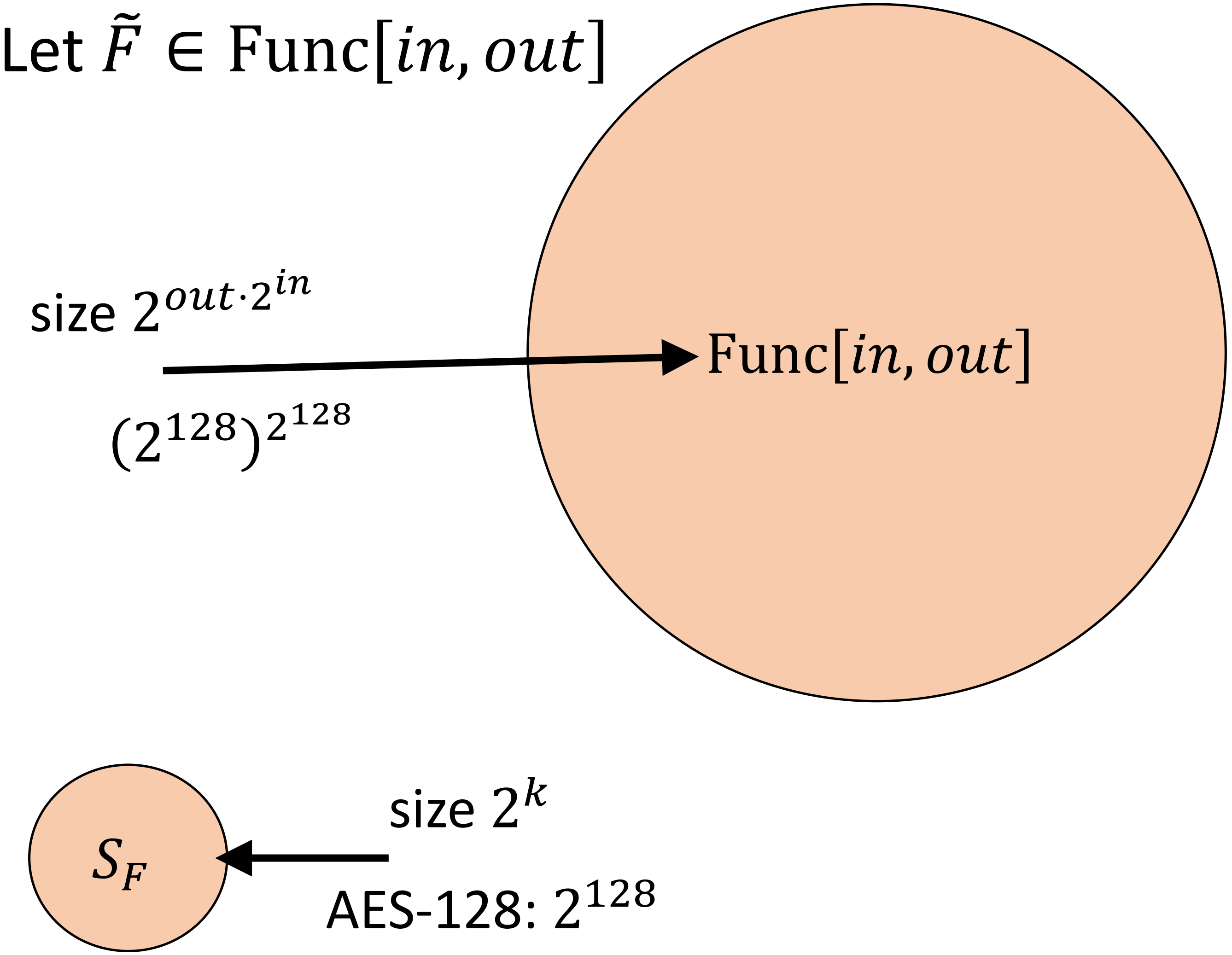
- Think of a PRF as a *family* of functions:
  - For each $K \in \{0,1\}^k$ we get a function $F_K : \{0,1\}^{in} \to \{0,1\}^{out}$ defined by $F_K(X) = F(K, X)$

# Secure PRFs

- Let $F : \{0,1\}^k \times \{0,1\}^{in} \to \{0,1\}^{out}$
- $S_F = \{ \; F_K \; | \, K \in \{0,1\}^k \} \subseteq \text{Func}[in, out]$
- $\text{Func}[in, out]$: the set of *all* functions from $\{0,1\}^{in}$ to $\{0,1\}^{out}$
- $F$ is **secure** if

$$\Pr\left[A^{F_K(\cdot)}(\quad) = 1 \,\middle|\, F_K \leftarrow S_F\right] - \Pr\left[A^{\tilde{F}(\cdot)}(\quad) = 1 \,\middle|\, \tilde{F} \leftarrow \text{Func}[in, out]\right] < negl$$

- Let $\tilde{F} \in \mathrm{Func}[in, out]$




$2^{in}$ (rows)

| $X$ | $\tilde{F}(X)$ |
|---|---|
| 000 … 000 | 101 … 111 |
| 000 … 001 | 001 … 001 |
| 000 … 010 | 111 … 100 |
| 000 … 011 | 101 … 000 |
| ⋮ | ⋮ |
| 111 … 111 | 100 … 010 |

$out$ (width of $\tilde{F}(X)$)

- We leave the construction of PRF in the next lecture

# IND-CPA secure Π2

Let $F_k$ be a PRF

**Alg** $\Pi 2.\mathrm{Enc}(K, M)$

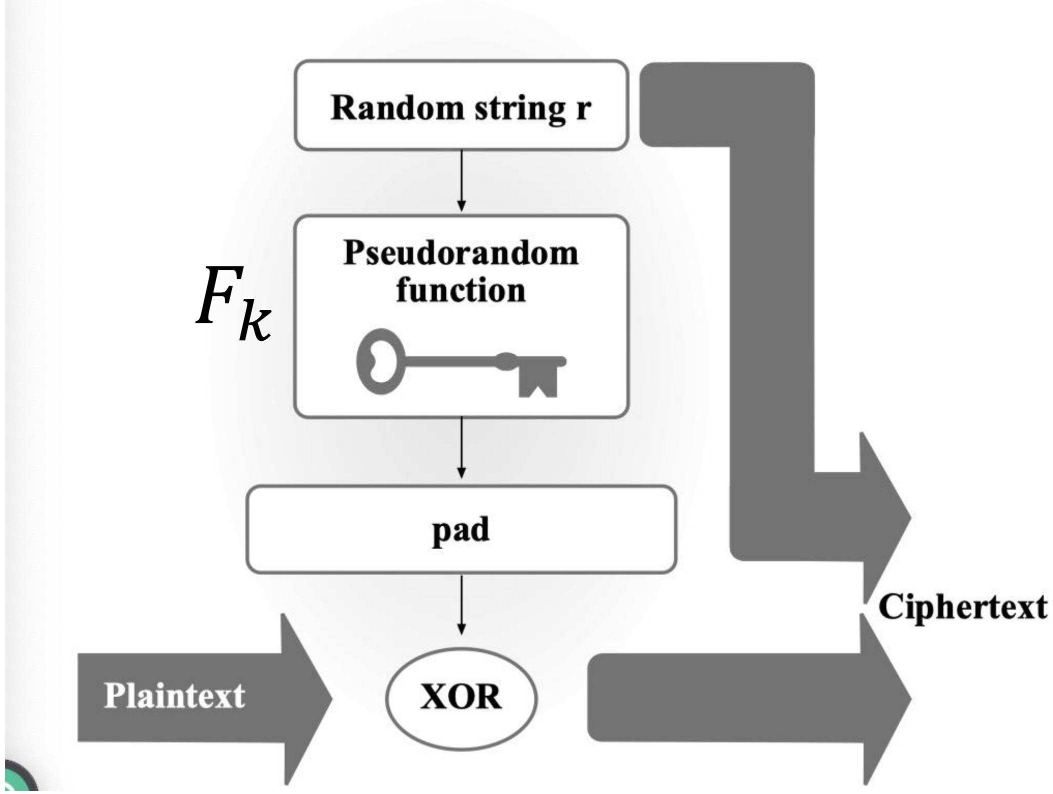
----------------------------------

1. $r \leftarrow \{0,1\}^\wedge n$
2. $c_2 = F_k(r) \oplus M$
3. **return** $< r, c_2 >$

**Alg** $\Pi 2.\mathrm{Dec}(K, C)$

----------------------------------

1. **return** $c_2 \oplus F_k(r)$



# Proof idea: IND-CPA (choose plaintext attack)

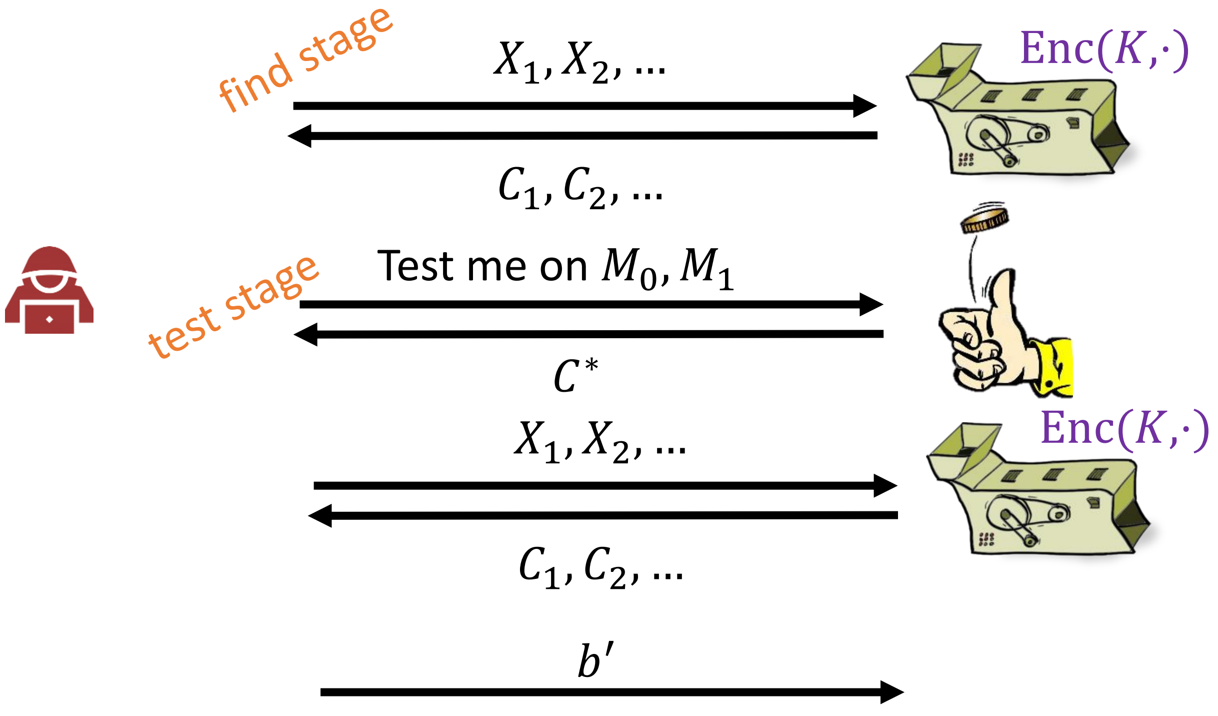
$\mathbf{Exp}_{\Pi 2}^{\text{ind-cpa}}(A)$

1. $b \xleftarrow{\$} \{0,1\}$
2. $K \xleftarrow{\$} \Pi 2.\mathrm{Gen}$
3. $M_0, M_1 \leftarrow A^{Enc(K,\cdot)}$ // find stage
4.
5.
6. $C^* \leftarrow < r^*, F_K(r^*) \oplus M_b >$ // test stage
7. $b' \leftarrow A^{Enc(K,\cdot)}(C^*)$ // guess stage
8. **return** $b' \stackrel{?}{=} b$

$Enc(K, M)$

----------------------------------

1. **return** $< r, F_K(r) \oplus M >$



# Proof idea: IND-CPA (choose plaintext attack)

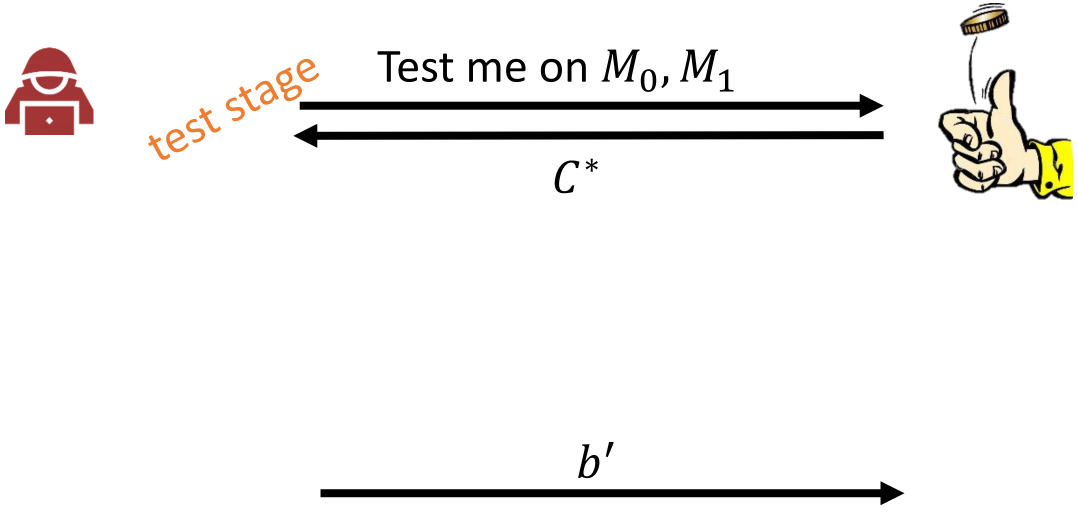
$\mathbf{Exp}^{\text{ind-cpa}}_{\Pi 2}(A)$

1. $b \overset{\$}{\leftarrow} \{0,1\}$
2. $K \overset{\$}{\leftarrow} \Pi 2.\mathrm{Gen}$
3. $M_0, M_1 \leftarrow A^{Enc(K,\cdot)}$ // find stage
4. 
5. 
6. $C^* \leftarrow < r^*, F_K(r^*) \oplus M_b >$ // test stage
7. $b' \leftarrow A^{Enc(K,\cdot)}(C^*)$ // guess stage
8. **return** $b' \overset{?}{=} b$

$Enc(K, M)$

---

1. **return** $< r, F_K(r) \oplus M >$ $\quad < r, \tilde{F}(r) \oplus M >$




Step 1: Due to PRF

# Proof idea: IND-CPA (choose plaintext attack)

**$\mathbf{Exp}_{\Pi 2}^{\text{ind-cpa}}(A)$**

1. $b \xleftarrow{\$} \{0,1\}$
2. $K \xleftarrow{\$} \Pi 2.\text{Gen}$
3. $M_0, M_1 \leftarrow A^{Enc(K,\cdot)}$ // find stage
4. 
5. 
6. $C^* \leftarrow < r^*, F_K(r^*) \oplus M_b >$ / $< r^*, \tilde{F}(r^*) \oplus M_b >$
7. $b' \leftarrow A^{Enc(K,\cdot)}(C^*)$ // guess stage
8. **return** $b' \stackrel{?}{=} b$

$Enc(K, M)$

---------------------------------

1. **return** $< r, F_K(r) \oplus M >$ $< r, \tilde{F}(r) \oplus M >$

test stage

Test me on $M_0, M_1$

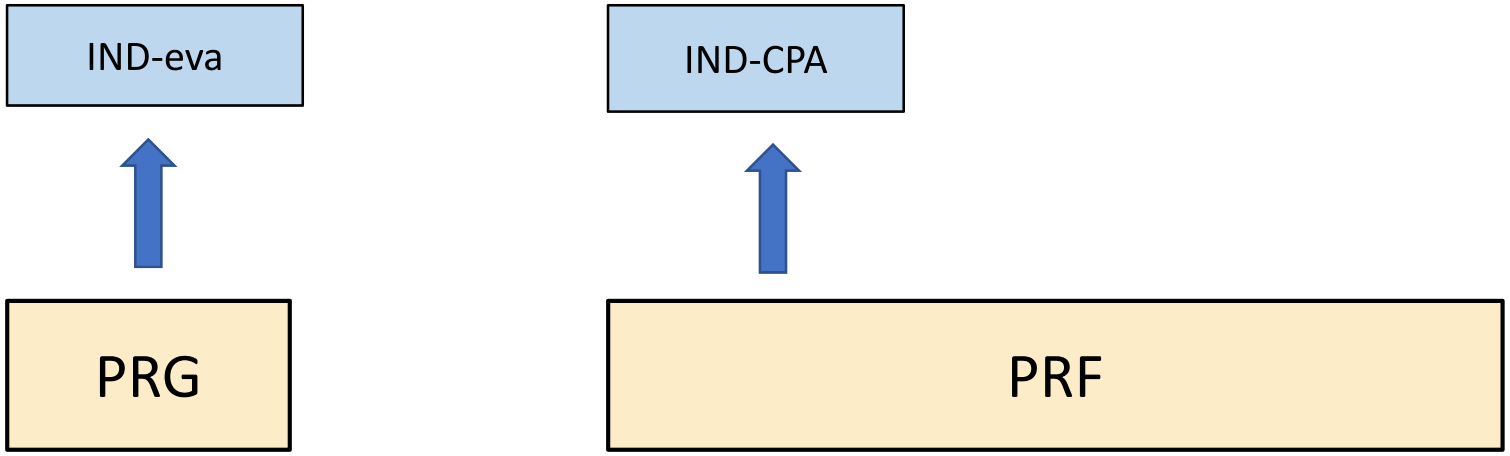
$C^*$

$b'$

Step 1: Due to PRF

Step 2: Due to PRF

# A short summary

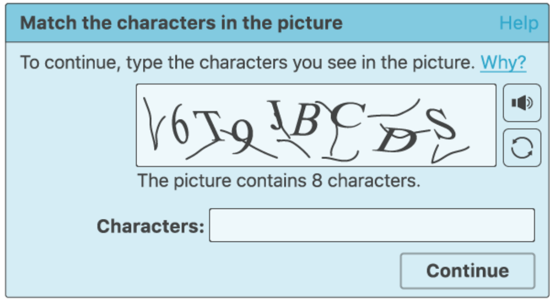


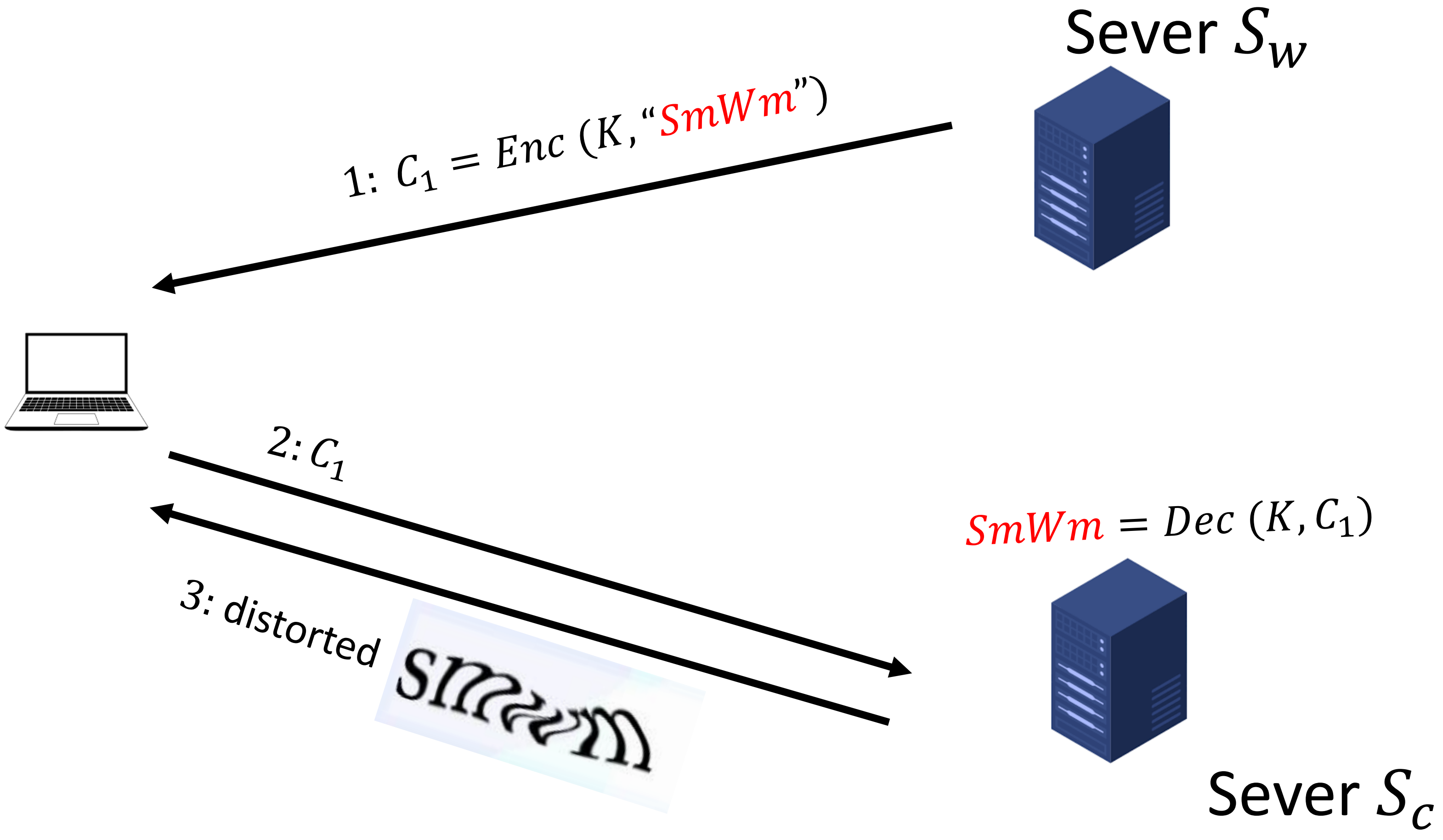
# A short summary

- Define IND-CPA is necessary
- $\Pi 1$ is not IND-CPA secure
- With PRF in hand, we can construct generic IND-CPA secure Enc
- In addition, CTR mode $\Pi 1$ is also IND-CPA secure
- Stronger security????

# Stronger Security: IND-CCA

- Example CAPTCHA




An adversary may have the capability to choose a ciphetext and get the message

# IND-CCA (choose ciphertext attack)

$\mathbf{Exp}_{\Pi}^{\text{ind-cpa}}(A)$

$b \overset{\$}{\leftarrow} \{0,1\}$

$K \overset{\$}{\leftarrow} \Pi.\mathrm{Gen}$

$M_0, M_1 \leftarrow A^{Enc(K,\cdot)}$ // find

  **if** $|M_0| \neq |M_1|$ **then**

  **return** $\perp$

$C^* \leftarrow \Pi.\mathrm{Enc}(K, M_b)$ // test

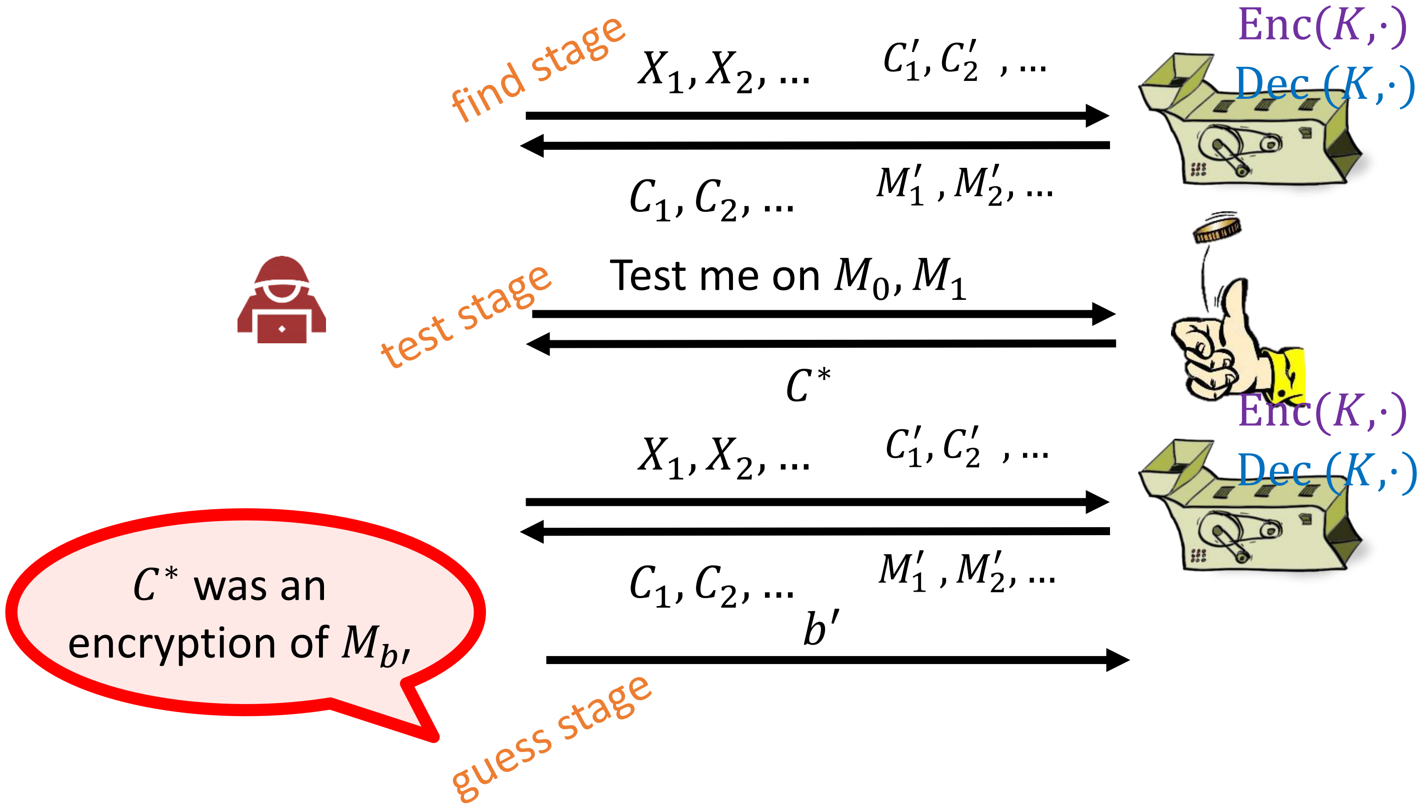
$b' \leftarrow A^{Enc(K,\cdot)}(C^*)$ // guess

**return** $b' \overset{?}{=} b$

$Enc(K, M)$

----------------------------------

**return** $\Pi.\mathrm{Enc}(K, M)$



# IND-CCA (choose ciphertext attack)

$\mathbf{Exp}_{\Pi}^{\text{ind-cca}}(A)$

$b \xleftarrow{\$} \{0,1\}$

$K \xleftarrow{\$} \Pi.\text{Gen}$

$M_0, M_1 \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}$ // find

  **if** $|M_0| \neq |M_1|$ **then**

  **return** $\perp$

$C^* \leftarrow \Pi.\text{Enc}(K, M_b)$ // test

$b' \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}(C^*)$ // guess

**return** $b' \stackrel{?}{=} b$

$Enc(K, M)$

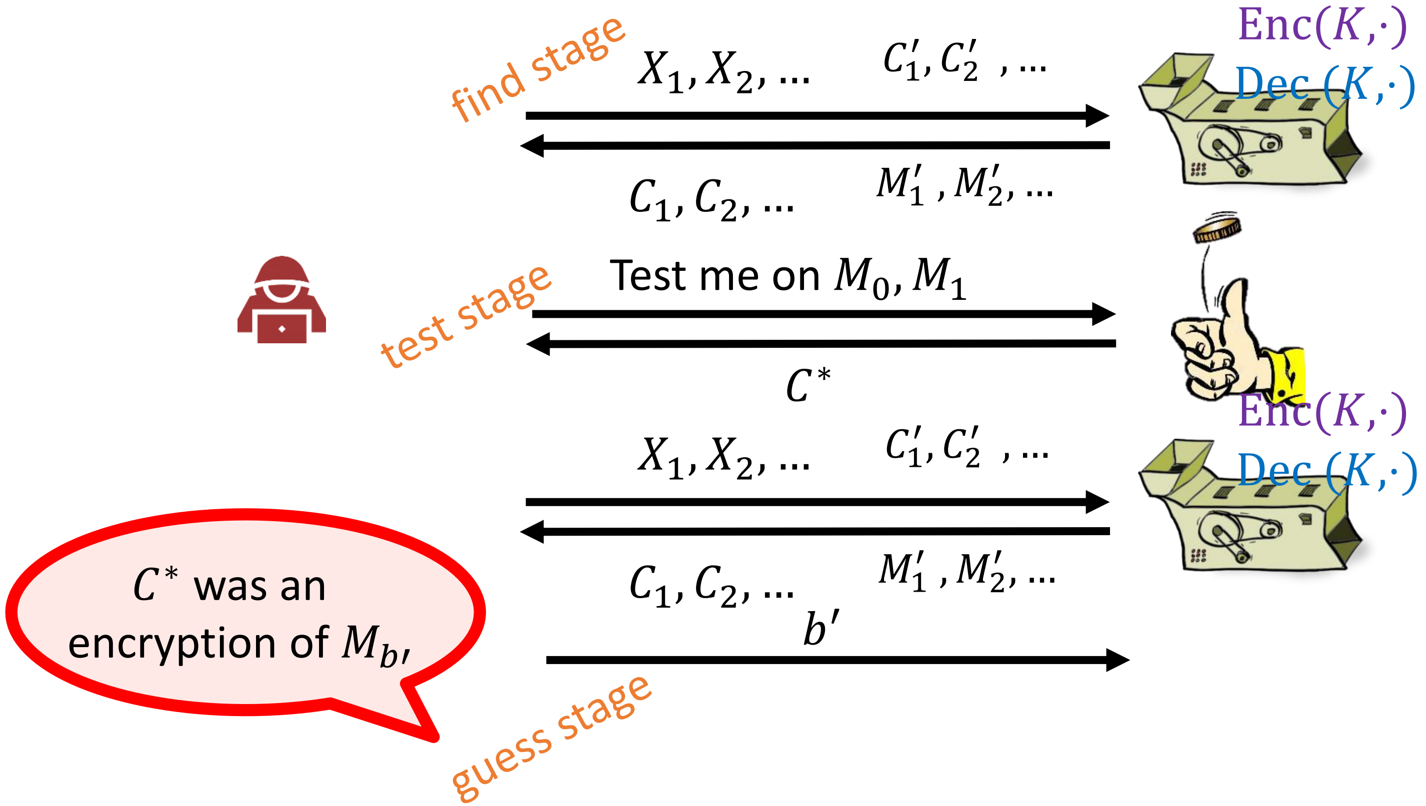
--------------------------------

**return** $\Pi.\text{Enc}(K, M)$

$Dec(K, C), C \neq C^*$

--------------------------------

**return** $\Pi.\text{Dec}(K, C)$




**Definition:** The **IND-CCA-advantage** of an adversary $A$ is

$$\mathbf{Adv}_{\Pi}^{\text{ind-cca}}(A) = \left|\Pr\left[\mathbf{Exp}_{\Pi}^{\text{ind-cca}}(A) \Rightarrow 1\right] - 1/2\right|$$

# IND-CCA Insecurity of $\Pi 2$

**Adversary $A$**

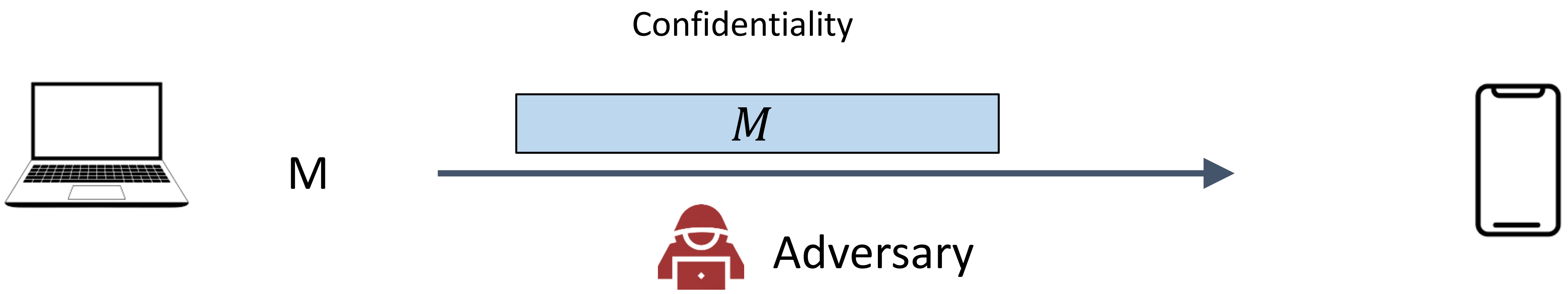
1. On receiving $C^* =< r^*, F_K(r^*) \oplus M_b >$
2. Query $C =< r^*, F_K(r^*) \oplus M_b \oplus M_0 >$ to Dec
3. On receiving $M_0 \oplus M_0$, set b=0
4. otherwise, b=1

# Constructions

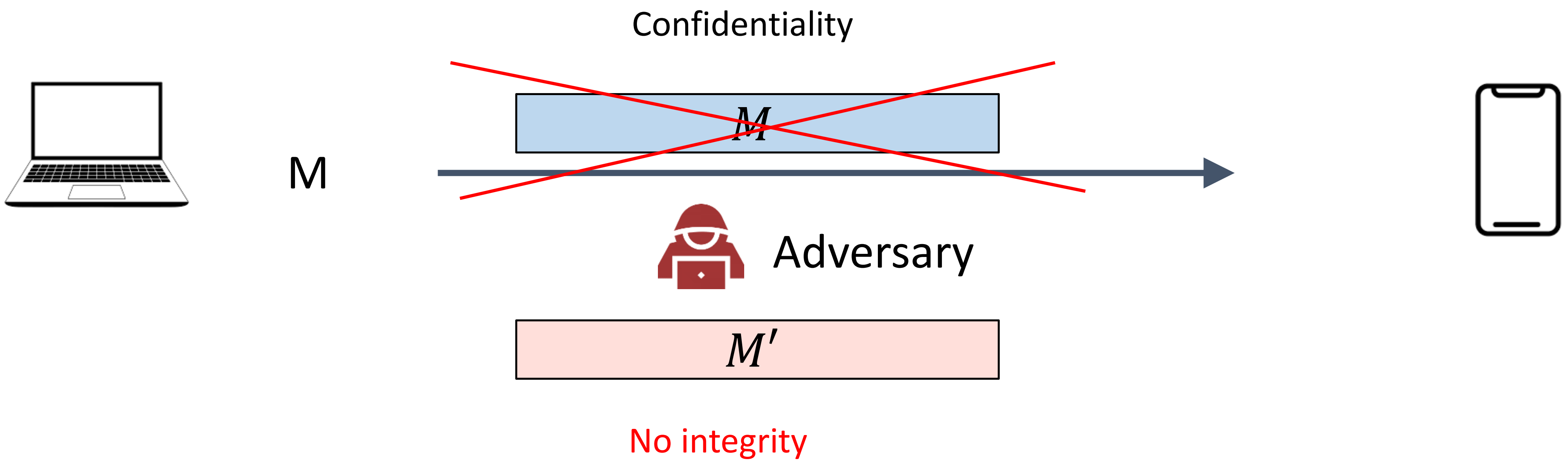
- We leave the construction of CCA secure Enc in the following part

- We actually construct a much stronger enc after introducing MAC

# Massage Authenticated Code



# Massage Authenticated Code

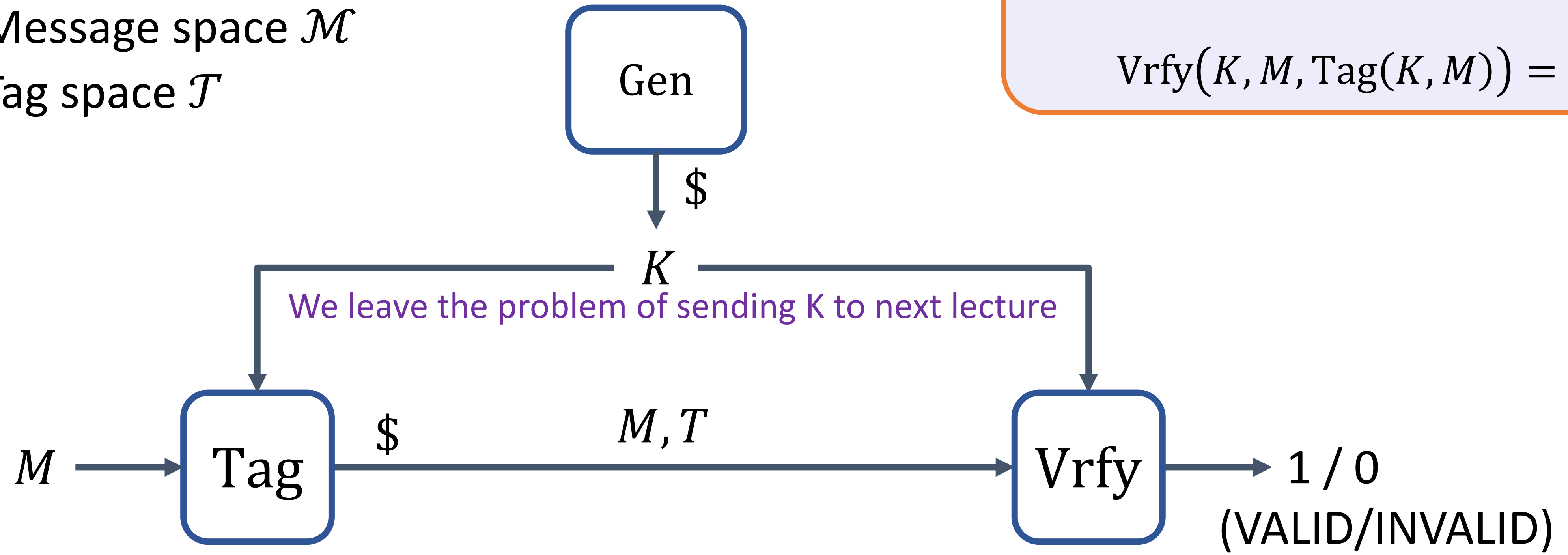


# Message authentication code (MAC)– syntax

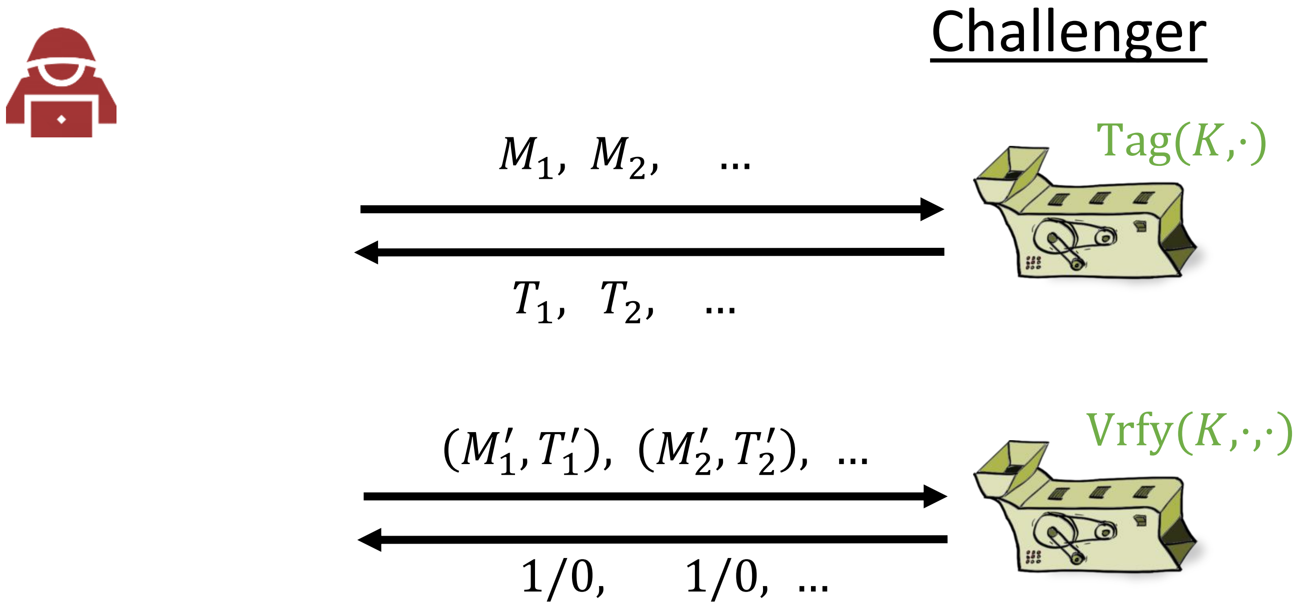
- A **message authentication scheme** $\Pi = (\mathrm{Gen}, \mathrm{Tag}, \mathrm{Vrfy})$ consists of three public algorithms:
- Associated to $\Pi$:
  - Key space $\mathcal{K}$
  - Message space $\mathcal{M}$
  - Tag space $\mathcal{T}$

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

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



# UF-CMA secure MAC

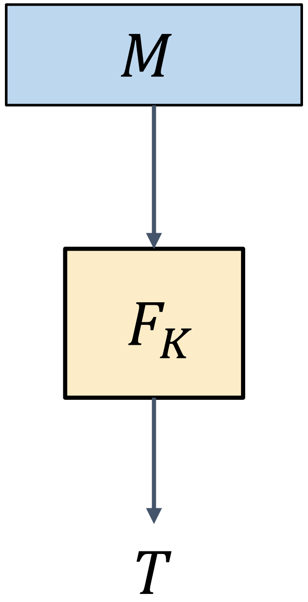



Adversary *wins* if a pair $(M_i', T_i')$ is valid,
and was not among the pairs $(M_1, T_1), (M_2, T_2)$, ...

# PRFs are good MACs

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

PRF




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

---

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

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

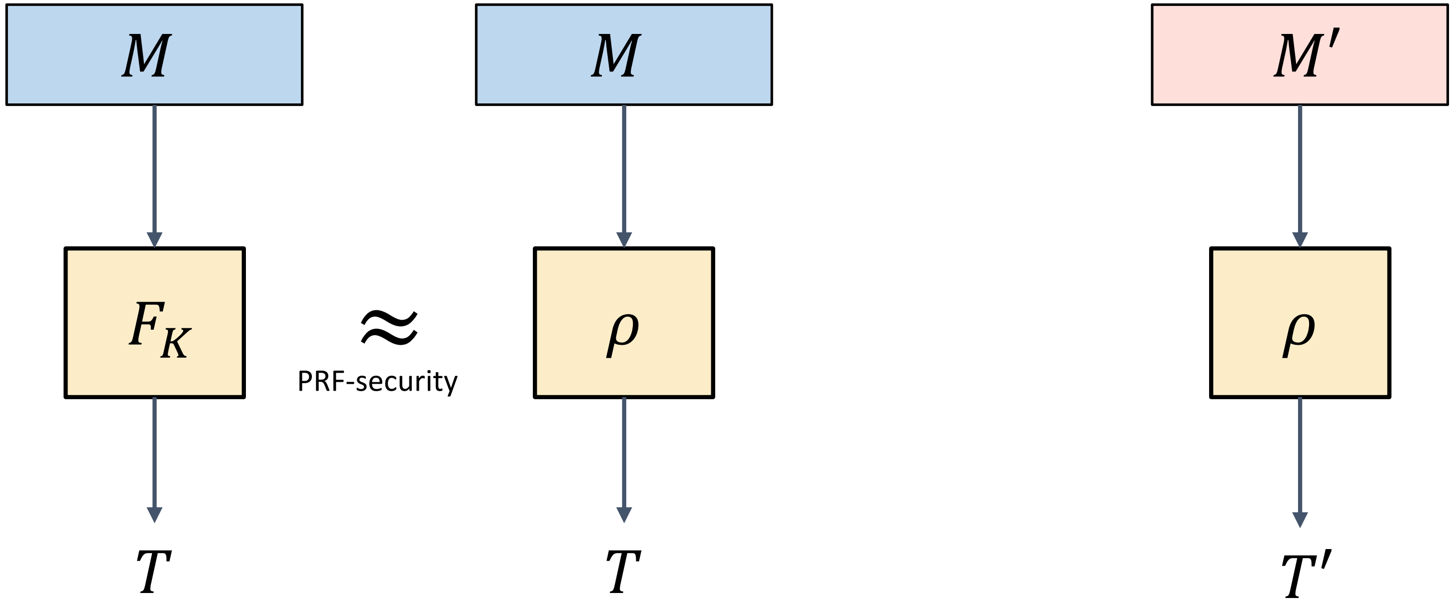
---

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

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

# PRFs are good MACs – proof sketch

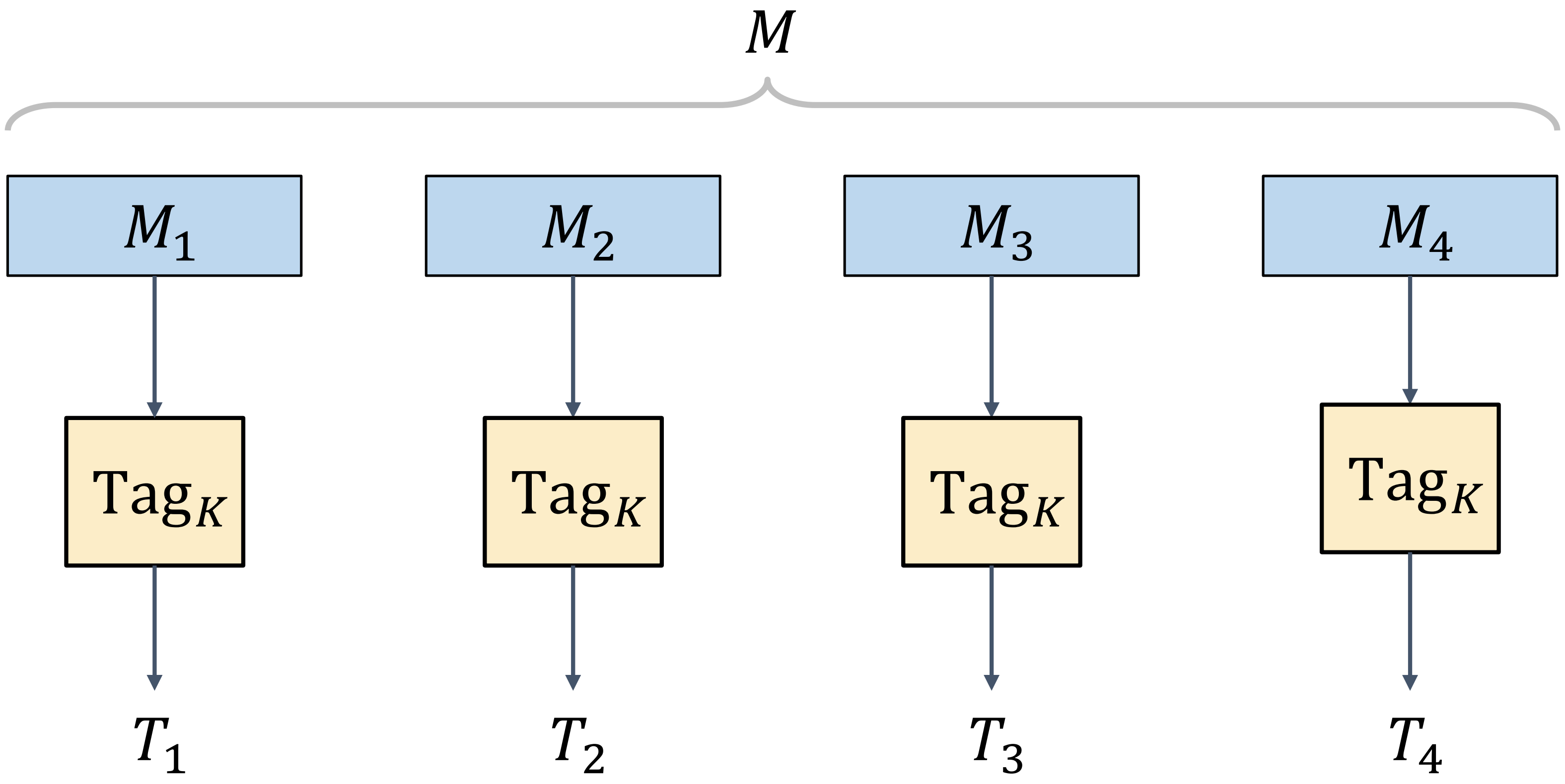
**Theorem:** If $F$ is a secure PRF then $\Sigma_{\mathrm{PRF}}$ is UF-CMA secure for *fixed-length* messages $M \in \{0,1\}^{in}$




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

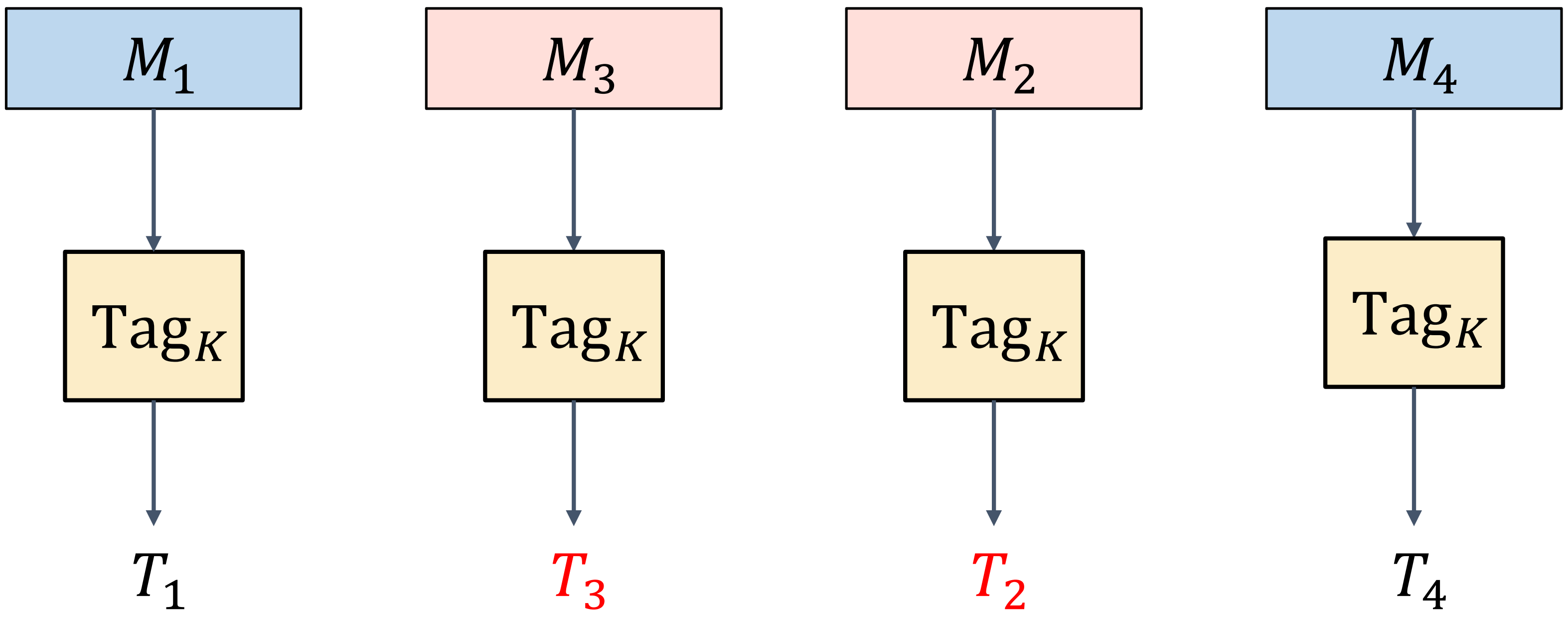
$$\rho \xleftarrow{\$} \mathrm{Func}[in, out]$$

# MAC for longer message Attempt 1:EBC



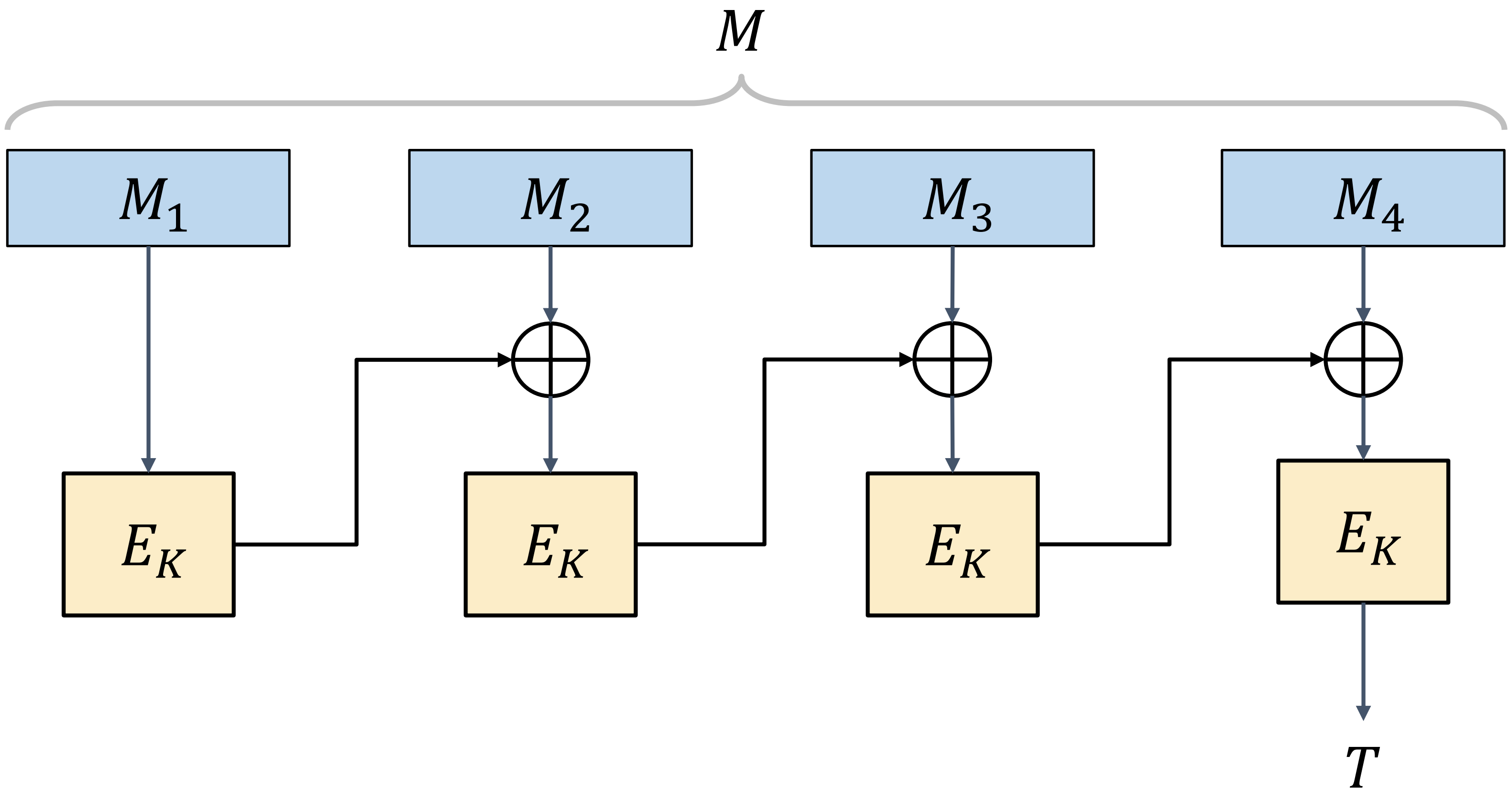

$$T = T_1 || T_2 || T_3 || T_4$$

# Attempt 1 – an attack




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

# CBC-MAC

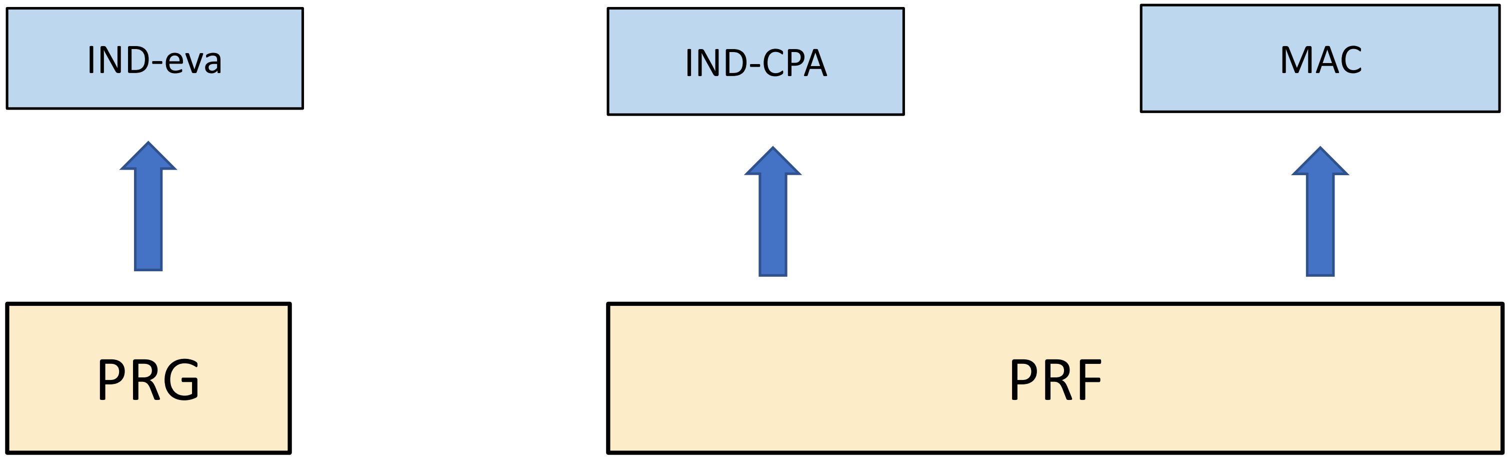



✓ Secure

# A short summary

- IND-CCA security is necessary
- Existing studied schemes are not IND-CCA secure
- MAC could be used to provide integrity.
- With IND-CPA enc and MAC, we are ready to construct IND-CCA

# A short summary



# Recall IND-CCA

$\mathbf{Exp}_{\Pi}^{\text{ind-cpa}}(A)$

$b \xleftarrow{\$} \{0,1\}$

$K \xleftarrow{\$} \Pi.\text{Gen}$

$M_0, M_1 \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}$ // find

$C^* \leftarrow \Pi.\text{Enc}(K, M_b)$ // test

$b' \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}(C^*)$ // guess

**return** $b' \stackrel{?}{=} b$

$Enc(K, M)$

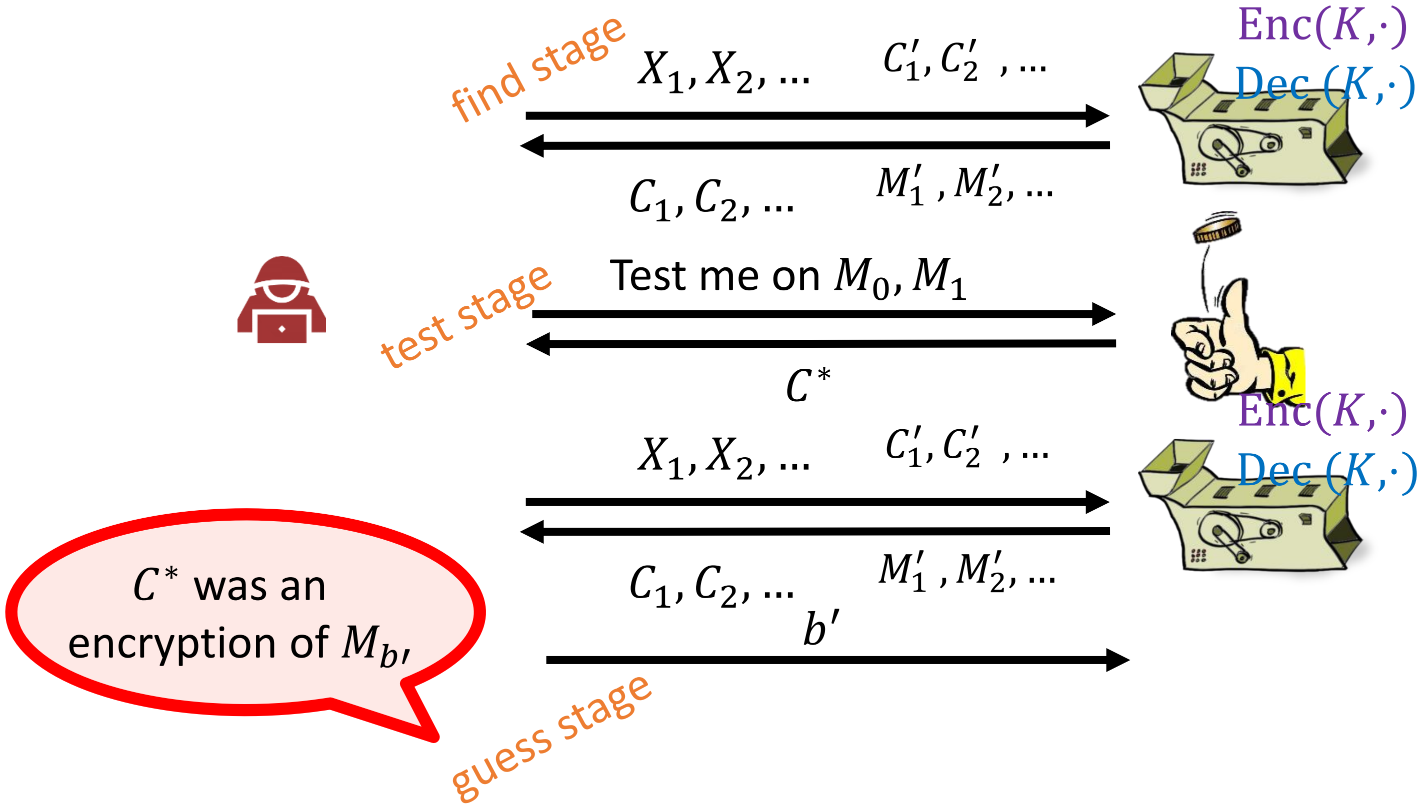
---------------------------------

**return** $\Pi.\text{Enc}(K, M)$

$Dec(K, C), C \neq C^*$

---------------------------------

**return** $\Pi.\text{Dec}(K, C)$

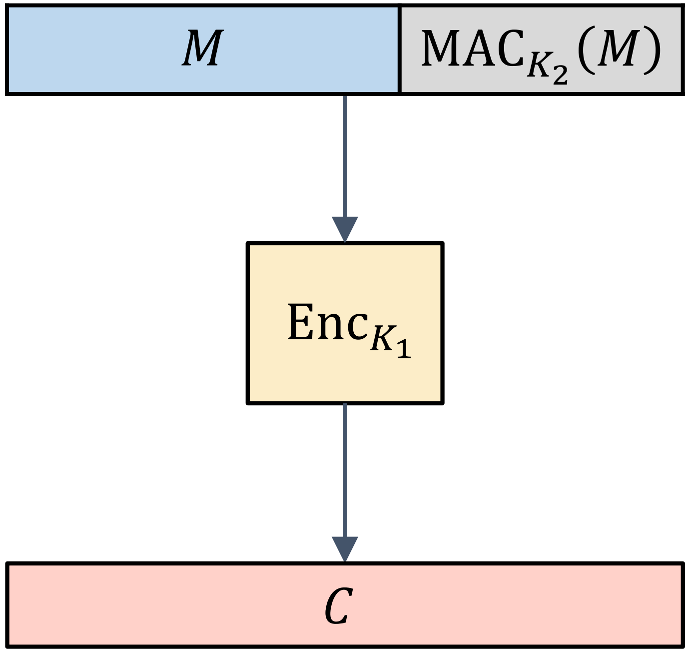



**Definition:** The **IND-CCA-advantage** of an adversary $A$ is

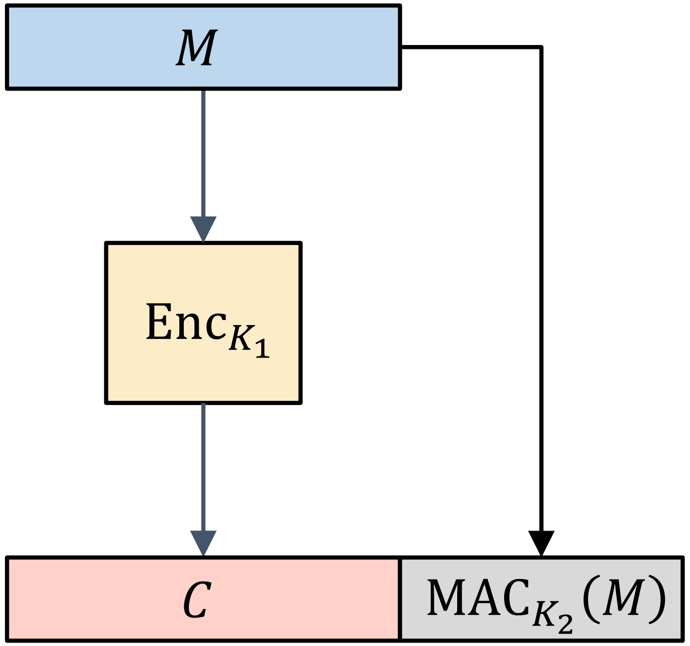
$$\mathbf{Adv}_{\Pi}^{\text{ind-cca}}(A) = \left|\Pr\left[\mathbf{Exp}_{\Pi}^{\text{ind-cca}}(A) \Rightarrow 1\right] - 1/2\right|$$

# Generic composition: IND-CPA + MAC?→ IND-CCA

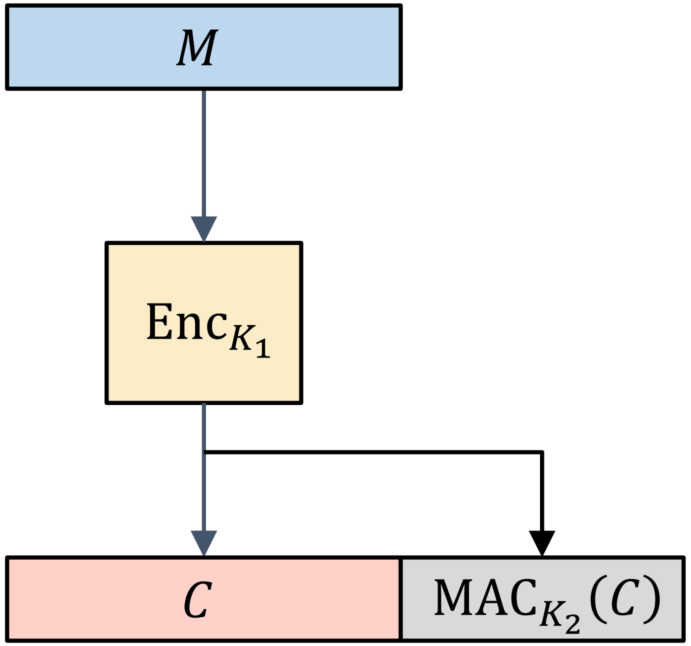
**MAC-then-Encrypt (MtE)**




**Encrypt-and-MAC (E&M)**




**Encrypt-then-MAC (EtM)**



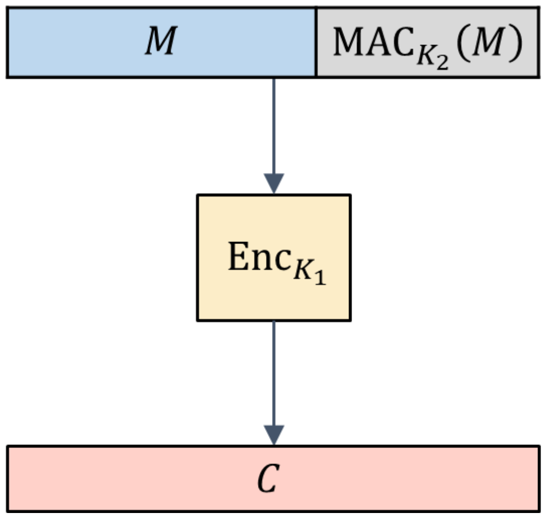
# First Attempt: MAC-then-Encrypt (MtE)

- If $Enc(K,M)$ is IND-CPA secure,
- $\mathrm{r}\,||Enc(K,M)$is also IND-CPA secure, where $\mathrm{r}$ is a random bit
- If $\mathrm{Enc_K}(\cdot) = r||Enc(K,\cdot)$

**CCA Adversary $A$**

1. Query $\bar{r}||Enc(K,M,MAC_{k_2}(M))$ to Dec

MAC-then-Encrypt (MtE)

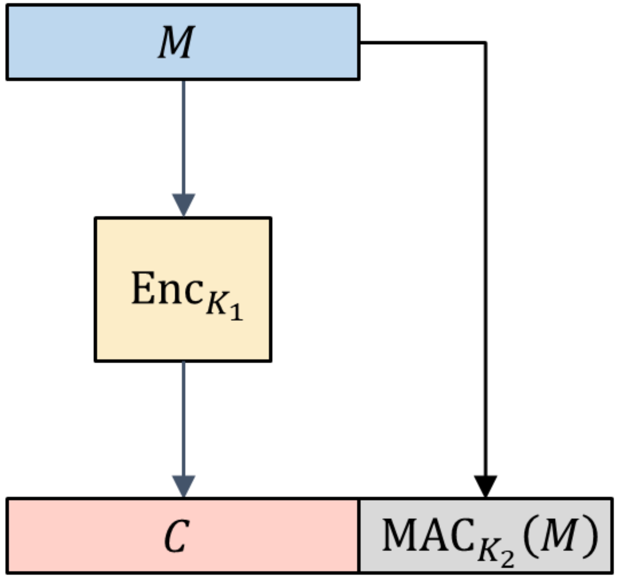


# Second Attempt: Encrypt-and-MAC (E&M)

- If $MAC_k(M)$ is a UF secure MAC,
- M || $MAC_k(M)$ is also a UF secure MAC

MAC does not provide confidentiality to the input

**Encrypt-and-MAC (E&M)**



# Encrypt-then-MAC (EtM)

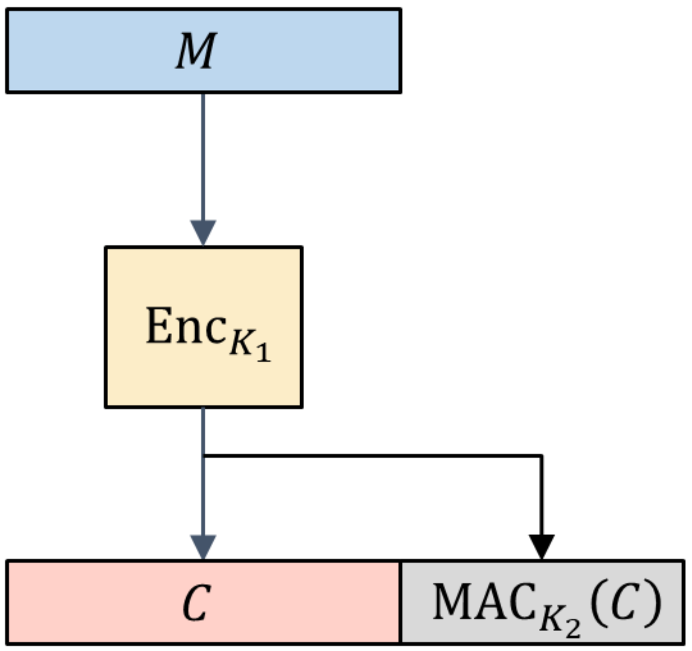
Let $\Pi2 = (\mathrm{Enc}, \mathrm{Dec})$ be an IND-CPA enc
Let $\Pi_\mathrm{m} = (\mathrm{Tag}, \mathrm{Vrfy})$ be a secure MAC

**Alg** $\Pi3.\mathrm{Gen}$
----------------------------------
1. **return** random $\mathrm{K} = (K_1, K_2)$

**Alg** $\Pi3.\mathrm{Enc}(K, M)$
----------------------------------
1. $C = \Pi2.\mathrm{Enc}(K_1, M)$
2. **return** $< C, \mathrm{Tag}(\mathrm{K}_2, \mathrm{C}) >$

**Alg** $\Pi3.\mathrm{Dec}(K, c_1||c_2)$
----------------------------------
1. **return** $\Pi2.\mathrm{Dec}(\mathrm{K}_2, \mathrm{c}_1)$ if $\mathrm{Vrfy}(\mathrm{K}_2, \mathrm{c}_1, \mathrm{c}_2) = 1$

**Encrypt-then-MAC (EtM)**



# Proof idea: IND-CCA

$\mathbf{Exp}_{\Pi3}^{\text{ind-cpa}}(A)$

1. $b \xleftarrow{\$} \{0,1\}$
2. $K \xleftarrow{\$} \Pi3.\text{Gen}$
3. $M_0, M_1 \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}$ // find
4. 
5. 
6. $C^* \leftarrow \Pi2.\text{Enc}(K_1, M_b)||MAC(K_2, \quad)$
7. $b' \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}(C^*)$ // guess
8. **return** $b' \overset{?}{=} b$

$Enc(K, M)$

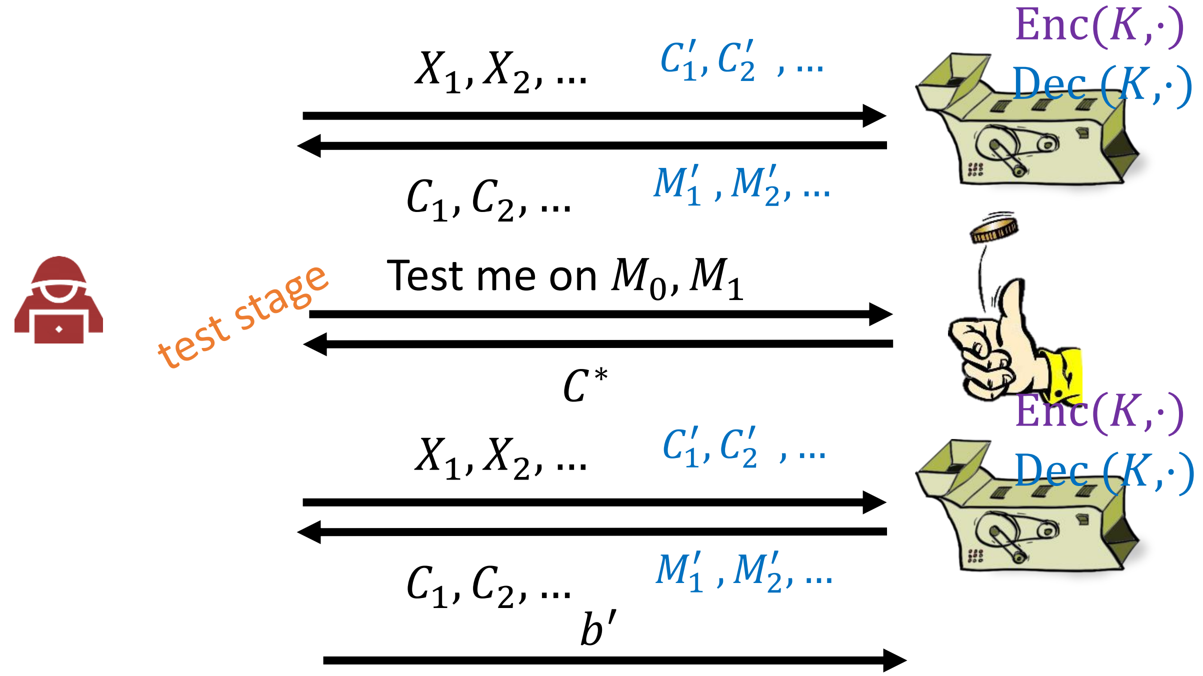
----------------------------------

1. **return** $\Pi2.\text{Enc}(K_1, M_b)||MAC(K_2, \quad)$

$Dec(K, c_1||c_2), c_1||c_2 \neq C^*$

----------------------------------

1. **return** $\Pi2.\text{Dec}(\text{K}_2, \text{c}_1)$ if $\text{Vrfy}(\text{K}_2, \text{c}_1, \text{c}_2) = 1$



# Proof idea: IND-CCA

$\mathbf{Exp}_{\Pi 3}^{\text{ind-cpa}}(A)$

$b \xleftarrow{\$} \{0,1\}$

$K \xleftarrow{\$} \Pi 3.\text{Gen}$

$M_0, M_1 \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}$ // find

$C^* \leftarrow \Pi 2.\text{Enc}(K_1, M_b)||MAC(K_2, \ )$

$b' \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}(C^*)$ // guess

**return** $b' \stackrel{?}{=} b$

$Enc(K, M)$

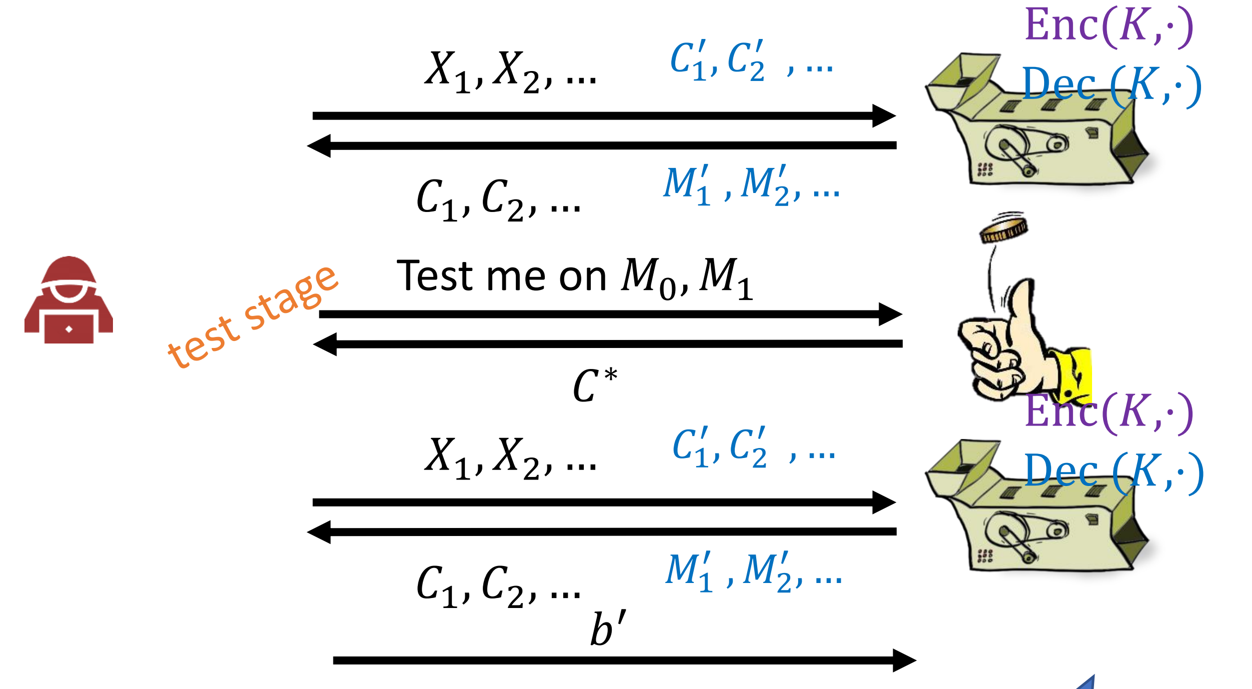
---------------------------------

1. **return** $\Pi 2.\text{Enc}(K_1, M_b)||MAC(K_2, \ )$

$Dec(K, c_1||c_2), c_1||c_2 \neq C^*$

---------------------------------

1. **return** $\Pi 2.\text{Dec}(\text{K}_2, \text{c}_1)$ if $\text{Vrfy}(\text{K}_2, \text{c}_1, \text{c}_2) = 1$




Dec(K, ) does not help

UF-CMA MAC

If $\text{Vrfy}(\text{K}_2, \text{c}_1, \text{c}_2) = 1$,
$c_1||c_2$ must be output of $Enc(K,\cdot)$,
return the message of that query

# Proof idea: IND-CCA

$\mathbf{Exp}_{\Pi 3}^{\text{ind-cpa}}(A)$

$b \xleftarrow{\$} \{0,1\}$
$K \xleftarrow{\$} \Pi 3.\mathrm{Gen}$
$M_0, M_1 \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}$ // find

$C^* \leftarrow \Pi 2.\mathrm{Enc}(K_1, M_b)||MAC(K_2, \ )$
$b' \leftarrow A^{Enc(K,\cdot)Dec(K,\cdot)}(C^*)$ // guess
**return** $b' \stackrel{?}{=} b$

$Enc(K, M)$

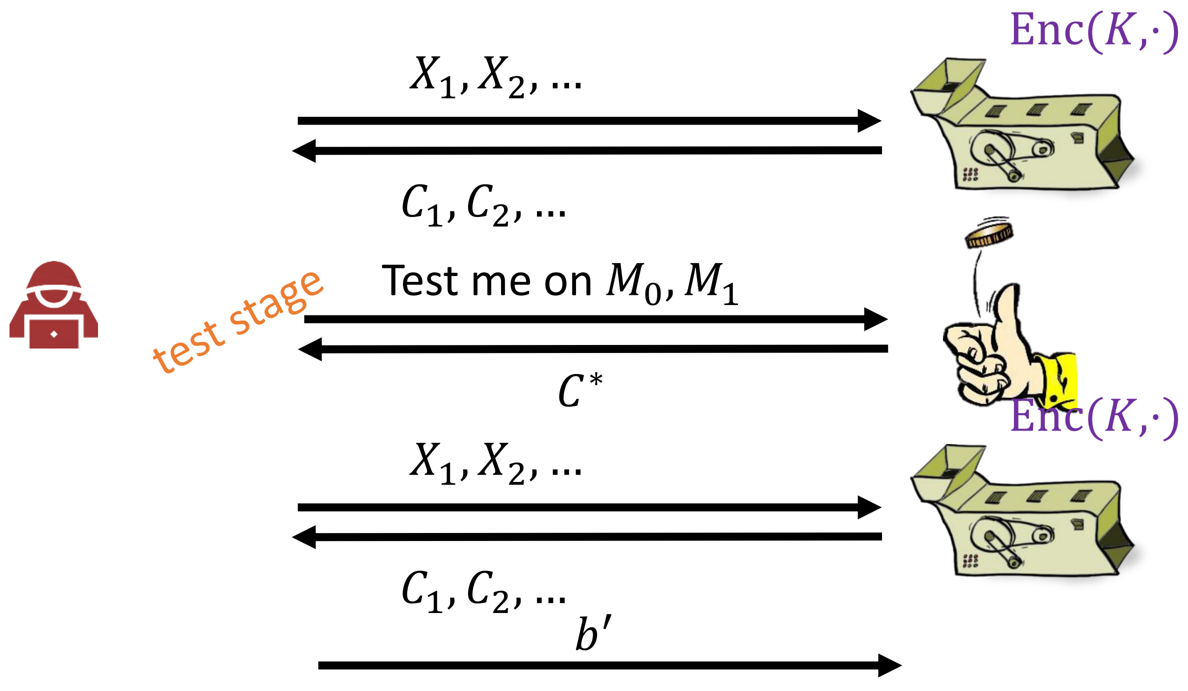
---------------------------------

**return** $\Pi 2.\mathrm{Enc}(K_1, M_b)||MAC(K_2, \ )$

$Dec(K, c_1||c_2), c_1||c_2 \neq C^*$

---------------------------------

**return** $\Pi 2.\mathrm{Dec}(\mathrm{K}_2, \mathrm{c}_1)$ if $\mathrm{Vrfy}(\mathrm{K}_2, \mathrm{c}_1, \mathrm{c}_2) = 1$

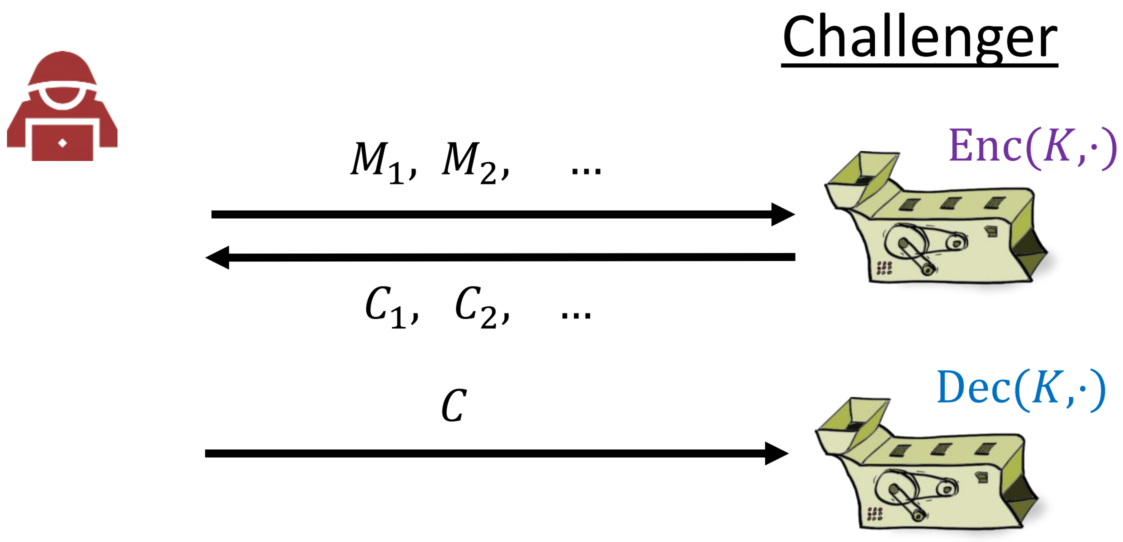



Dec(K, ) does not help

IND-CPA is enough to handle the other cases

# Π3 satisfies more

unforgeable encryption



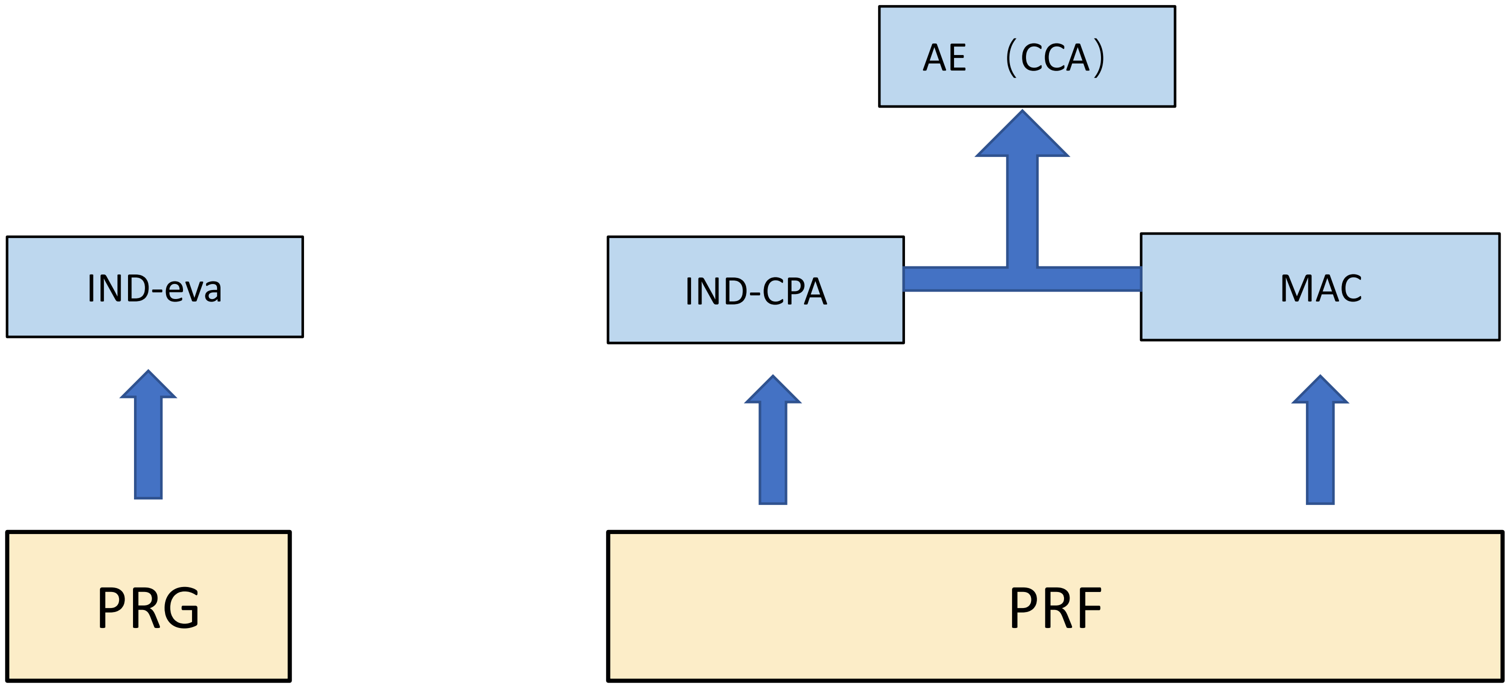

Adversary *wins* if $\mathrm{m} = \mathrm{Dec}\,(K, C) \neq \bot$,
and $\mathrm{m}$ was not among the set $\{M_1, M_2, \cdots\}$

**Definition:** An **authenticated encryption** is a unforgeable encryption that is IND-CCA secure.

# A short summary

- IND-CCA security is necessary
- We could construct an IND-CCA secure scheme from IND-CAP + MAC using Encrypt-then-MAC (EtM)
- The resulting scheme is actually an Authenticated Encryption (AC)

# A short summary



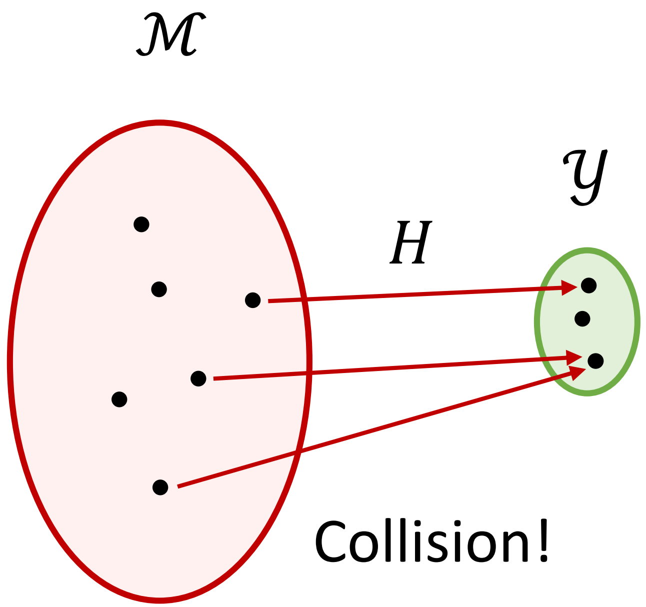
# Hash function


https://www.thesslstore.com/blog/what-is-a-hash-function-in-cryptography-a-beginners-guide/

# Hash functions

$H : \mathcal{M} \to \mathcal{Y}$ — Keyless function

$|\mathcal{M}| \gg |\mathcal{Y}|$ — Compressing




- SHA1 *: $\{0,1\}^{<2^{64}} \to \{0,1\}^{160}$
- SHA2−256 : $\{0,1\}^{<2^{64}} \to \{0,1\}^{256}$
- SHA3−512 : $\{0,1\}^{<2^{128}} \to \{0,1\}^{512}$

**Collision Resistant**

**One way**

* Insecure now. https://en.wikipedia.org/wiki/SHA-1#Attacks

# Collision resistance

**$\mathbf{Exp}_H^{\mathrm{cr}}(A)$**

1. $(X_1, X_2) \leftarrow A_H$
2. **if** $X_1 \neq X_2$ **and** $H(X_1) = H(X_2)$ **then**
3. **return** 1
4. **else**
5. **return** 0

***A***

1. Output $(X_1, X_2)$ where $X_1, X_2$ is a collision for $H$

$X_1, X_2$ must *exist* since $|\mathcal{M}| \gg |\mathcal{Y}|$

hence $\mathbf{Adv}_H^{\mathrm{cr}}(A) = 1$ for unbounded A

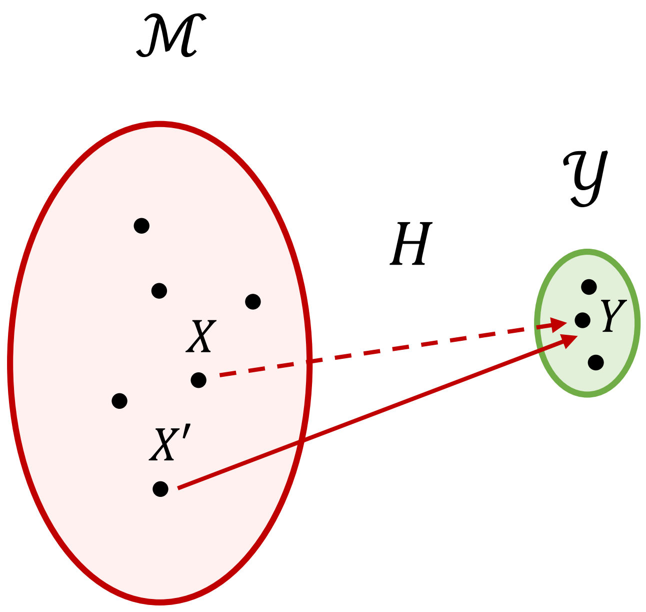
…but how do we actually find $X_1, X_2$?!

**Definition:** The **CR-advantage** of an adversary $A$ against $H$ is

$$\mathbf{Adv}_H^{\mathrm{cr}}(A) = \Pr[\mathbf{Exp}_H^{\mathrm{cr}}(A) \Rightarrow 1]$$

# One-way security

| $\mathbf{Exp}_H^{\mathrm{ow}}(A)$ |
|---|
| 1. $X \xleftarrow{\$} \mathcal{M}$ |
| 2. $Y \leftarrow H(X)$ |
| 3. $X' \leftarrow A_H(Y)$ |
| 4. **return** $H(X') \stackrel{?}{=} Y$ |




**Definition:** The **OW-advantage** of an adversary $A$ against $H$ is

$$\mathbf{Adv}_H^{\mathrm{ow}}(A) = \Pr[\mathbf{Exp}_H^{\mathrm{cr}}(A) \Rightarrow 1]$$

# Relation between notions

| $\mathbf{Exp}_H^{\mathrm{cr}}(A)$ |
|---|
| 1. $(X_1, X_2) \leftarrow A_H$ |
| 2. **if** $X_1 \neq X_2$ **and** $H(X_1) = H(X_2)$ **then** |
| 3.     **return** 1 |
| 4. **else** |
| 5.     **return** 0 |

| $\mathbf{Exp}_H^{\mathrm{ow}}(A)$ |
|---|
| 1. $X \xleftarrow{\$} \mathcal{M}$ |
| 2. $Y \leftarrow H(X)$ |
| 3. $X' \leftarrow A_H(Y)$ |
| 4. **return** $H(X') \stackrel{?}{=} Y$ |

Collision-resistance $\Longrightarrow$ One-wayness

**Proof idea:** suppose $A_{\mathrm{ow}}$ is an algorithm that breaks one-wayness

1. Pick $X \xleftarrow{\$} \mathcal{M}$ and give $Y \leftarrow H(X)$ to $A_{\mathrm{ow}}$
2. $A_{\mathrm{ow}}$ outputs $X'$
3. output $(X, X')$ as a collision ( $H(X') = Y = H(X)$ )

Problem: what if $X' = X$? Very unlikely assuming $|\mathcal{M}| \gg |\mathcal{Y}|$

# Relation between notions

| $\mathbf{Exp}_H^{\mathrm{cr}}(A)$ |
|---|
| 1. $(X_1, X_2) \leftarrow A_H$ |
| 2. **if** $X_1 \neq X_2$ **and** $H(X_1) = H(X_2)$ **then** |
| 3.     **return** 1 |
| 4. **else** |
| 5.     **return** 0 |

| $\mathbf{Exp}_H^{\mathrm{ow}}(A)$ |
|---|
| 1. $X \xleftarrow{\$} \mathcal{M}$ |
| 2. $Y \leftarrow H(X)$ |
| 3. $X' \leftarrow A_H(Y)$ |
| 4. **return** $H(X') \stackrel{?}{=} Y$ |

Collision-resistance $\Longrightarrow$ One-wayness

Collision-resistance $\not\Longleftarrow$ One-wayness

Suppose $H : \mathcal{M} \rightarrow \{0,1\}^{256}$ is one-way. Define

$$H'(X) = \begin{cases} 0^{256} & \text{if } X = 0 \text{ or } X = 1 \\ H(X) & \text{otherwise} \end{cases}$$

$H'$ is one-way

$H'$ is **not** collision-resistant

# Application– MAC domain extension (HMAC)

$\mathrm{MAC} : \mathcal{K} \times \{0,1\}^n \to \mathcal{T}$ $\quad H : \{0,1\}^* \to \{0,1\}^n$

$\mathrm{MAC}' : \mathcal{K} \times \{0,1\}^* \to \mathcal{T}$

$\mathrm{MAC}'(K, M) = \mathrm{MAC}\big(K, H(M)\big)$ ⟵ Hash-then-MAC paradigm

**Theorem:** If $H$ is collision-resistant and $\mathrm{MAC}$ is UF-CMA secure, then $\mathrm{MAC}'$ is UF-CMA secure

# A short summary

- Hash functions are compressing functions
- Collision resistance and one-wayness are two properties of hash function
- Hash could be used to build HMAC

# Summary

- Syntax and security of symmetric-key cryptography
- Perfect security and one-time pad
- Stream cipher, block cipher and MAC
- Hash function
- Constructions

# Recap

| **Primitives** | **Security** | **Examples** |
|---|---|---|
| Pseudorandom function (PRF) | Indistinguishability from random function | AES<br>HMAC |
| Encryption | IND-eva<br>IND-CPA<br>IND-CCA | PRG<br>$+PRF<br>Enc-t-Mac |
| MAC | Integrity | PRF<br>CBC-MAC<br>HMAC |
| Authenticated Encryption | IND-CCA + unforgeable encryption | IND-CPA+MAC<br>AES-256-GCM |
| Hash function | Collision-resistance + one-wayness | SHA2-256<br>SHA2-512<br>SHA3 |

# Theoretical constructions

- How to construct PRG and PRF
- from one-way function

- We will talk in the next lecture since PKE also relies on them

Thank you

Happy Chinese New Year