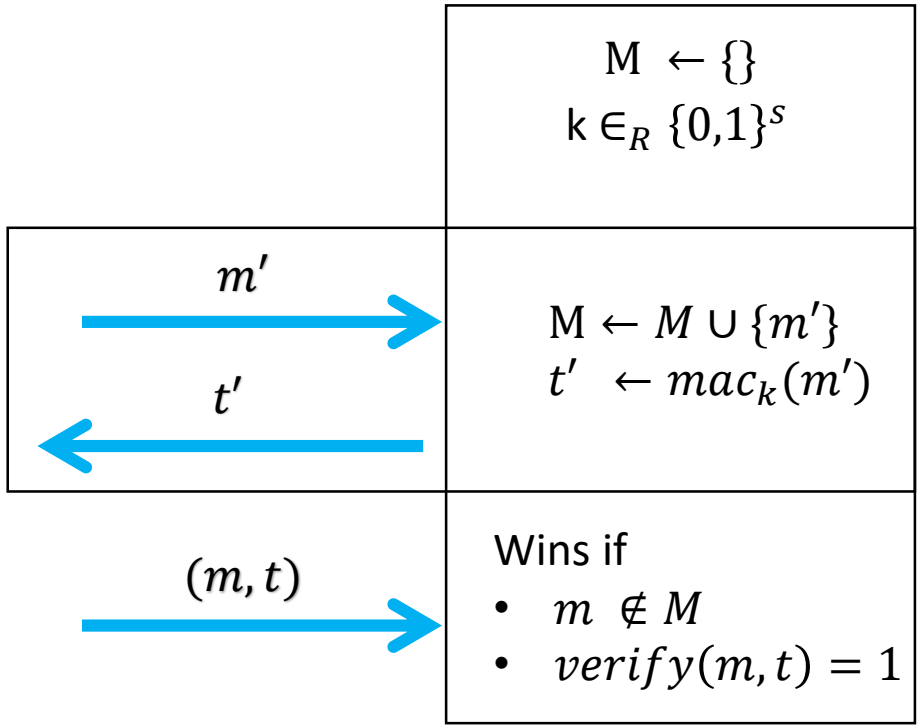
# Authenticated encryption

# Mac forgery game

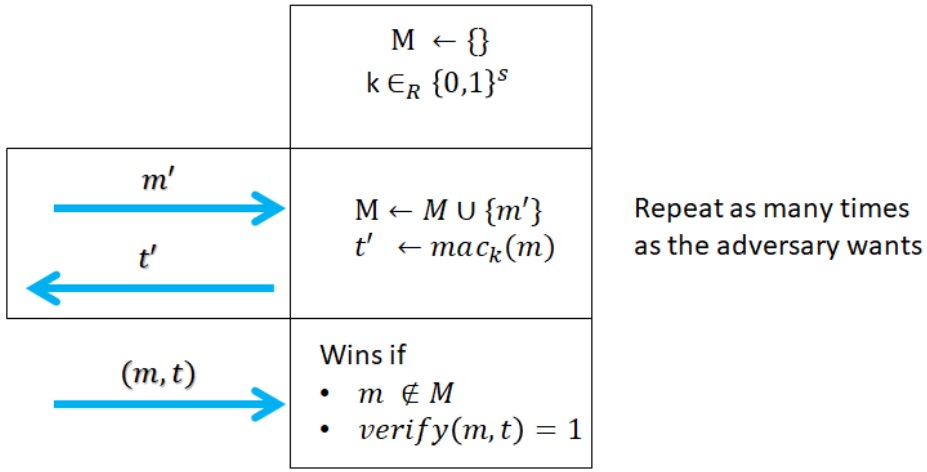



Repeat as many times
as the adversary wants

# Mac forgery game

- Allow the adversary to learn tags for as many message as he wants

- A mac scheme is secure if
  - $\Pr[adv\ wins\ the\ forgery\ game] \leq negl(s)$



# Does authentication imply secrecy

- Consider the question from the quiz

> Let $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Verify})$ be a message authentication code, and define
>
> $$\begin{aligned}
> &\mathsf{Gen}' := \mathsf{Gen} \\
> &\mathsf{Mac}'(k, m) := (\mathsf{Mac}(k, m), m) \\
> &\mathsf{Verify}'(k, m, t) := \\
> &\qquad (t', m') \leftarrow t \\
> &\qquad \text{return } \mathsf{Verify}(k, m, t').
> \end{aligned}$$
>
> Is $(\mathsf{Gen}', \mathsf{Mac}', \mathsf{Verify}')$ also a message authentication code? Briefly explain your answer.

- The answer is yes.

- To prove this is the case, we will take an adversary which forges a mac for this scheme and breaks the original mac scheme

# Does authentication imply secrecy

- Consider the question from the quiz

Let $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Verify})$ be a message authentication code, and define

$$\begin{aligned}
&\mathsf{Gen}' := \mathsf{Gen}\\
&\mathsf{Mac}'(k, m) := (\mathsf{Mac}(k, m), m)\\
&\mathsf{Verify}'(k, m, t) :=\\
&\qquad (t', m') \leftarrow t\\
&\qquad \text{return } \mathsf{Verify}(k, m, t').
\end{aligned}$$

Is $(\mathsf{Gen}', \mathsf{Mac}', \mathsf{Verify}')$ also a message authentication code? Briefly explain your answer.

More formally, if the scheme is insecure => ∃ $A \in PPT$ which produces $t = (t', m')$
such that $Verify(k, m, (t', m')) = acctept$ for a fresh m'

However since $Verify'(k, m, (t', m')) = Verify(k, m, t')$, this means that the adversary created a mac tag for the original scheme.
Hence, the original scheme is not a mac scheme.

By contradiction, we have a mac scheme.

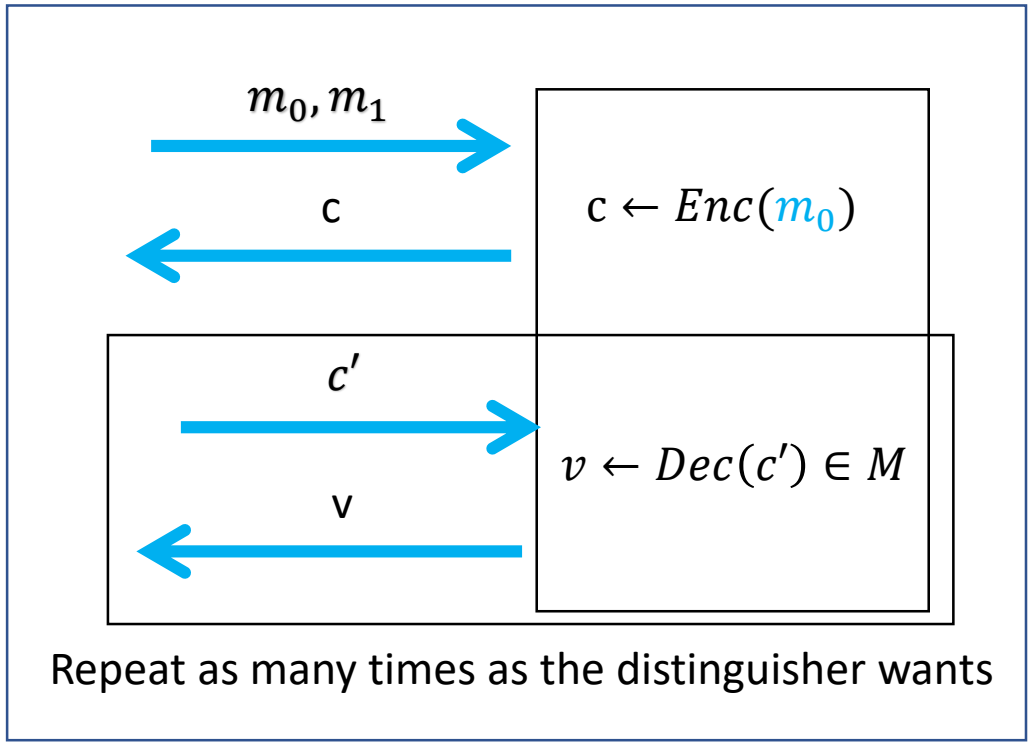
# Lesson

- Lesson
  - Authentication $\not\Rightarrow$ Encryption
  - Encryption $\not\Rightarrow$ Authentication
- In the future, if I ever see anyone mention ciphertext in a question that only talks about macs, there will be a loss of points.

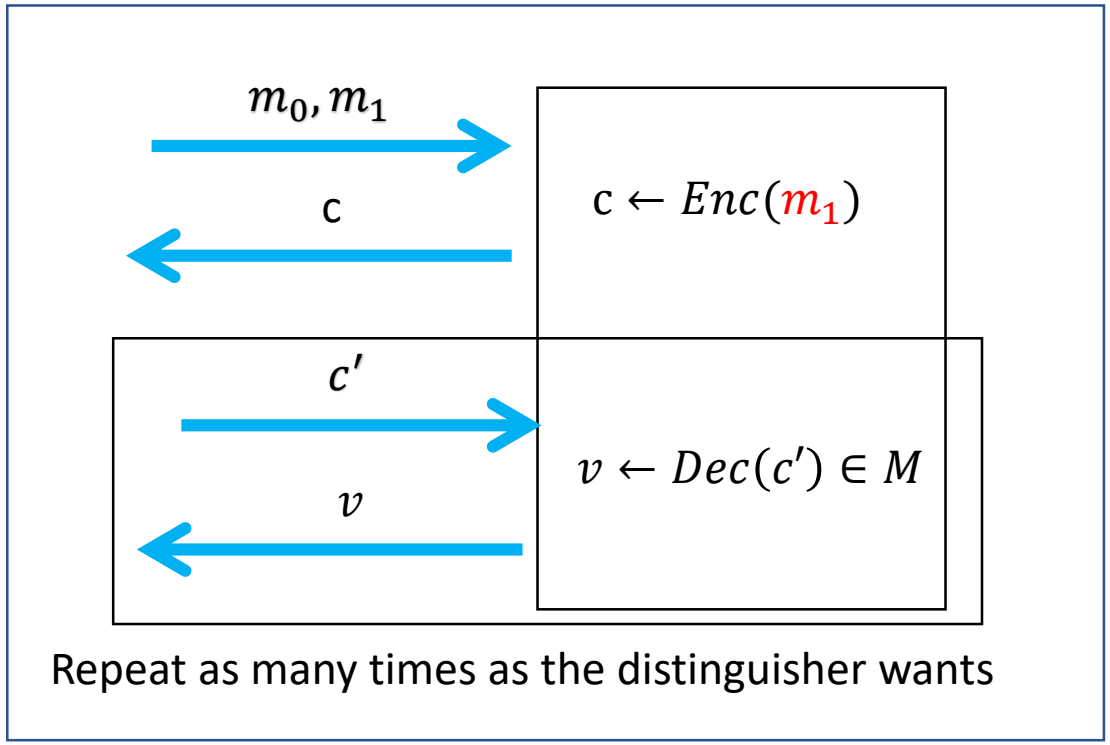
# Validation-oracle indistinguishability game

- Validation-oracle games
  - Adversary chooses $m_0, m_1$
  - In $G_0$, the game returns $Enc(m_0)$
  - In $G_1$, the game returns $Enc(m_1)$
  - In both $G_0$ and $G_1$, the adversary can send extra ciphertexts and the oracle tells the adversary if the decryption of the ciphertext falls into the message space
- The adversary has to guess which game he is playing

# Validation-oracle indistinguishability game



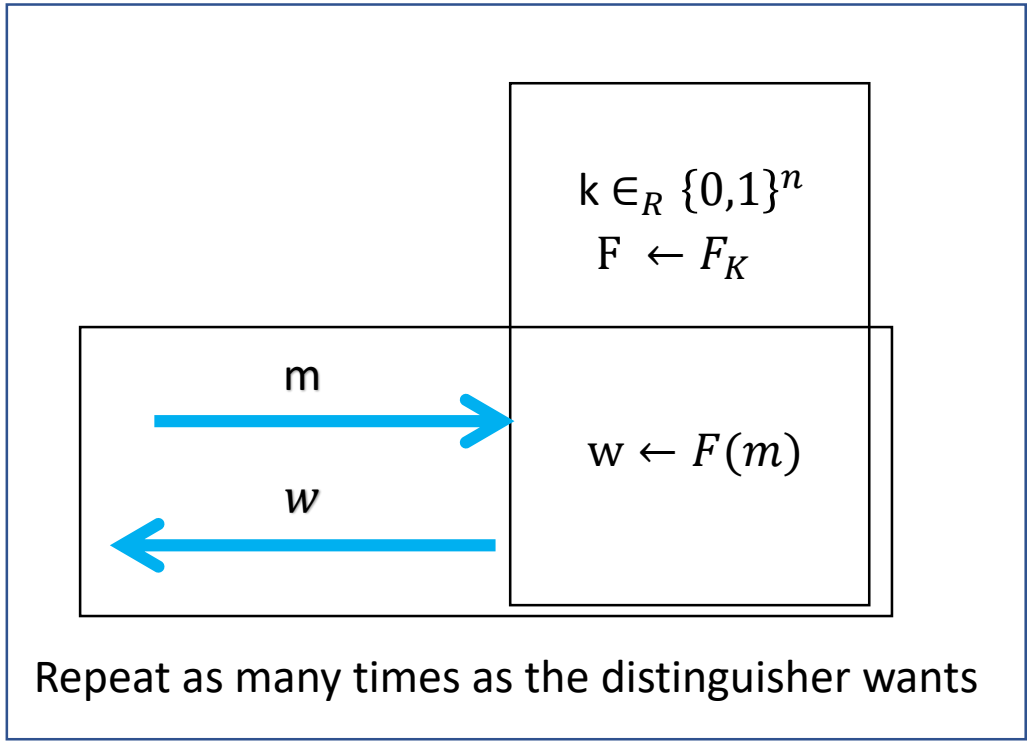

$G_0$



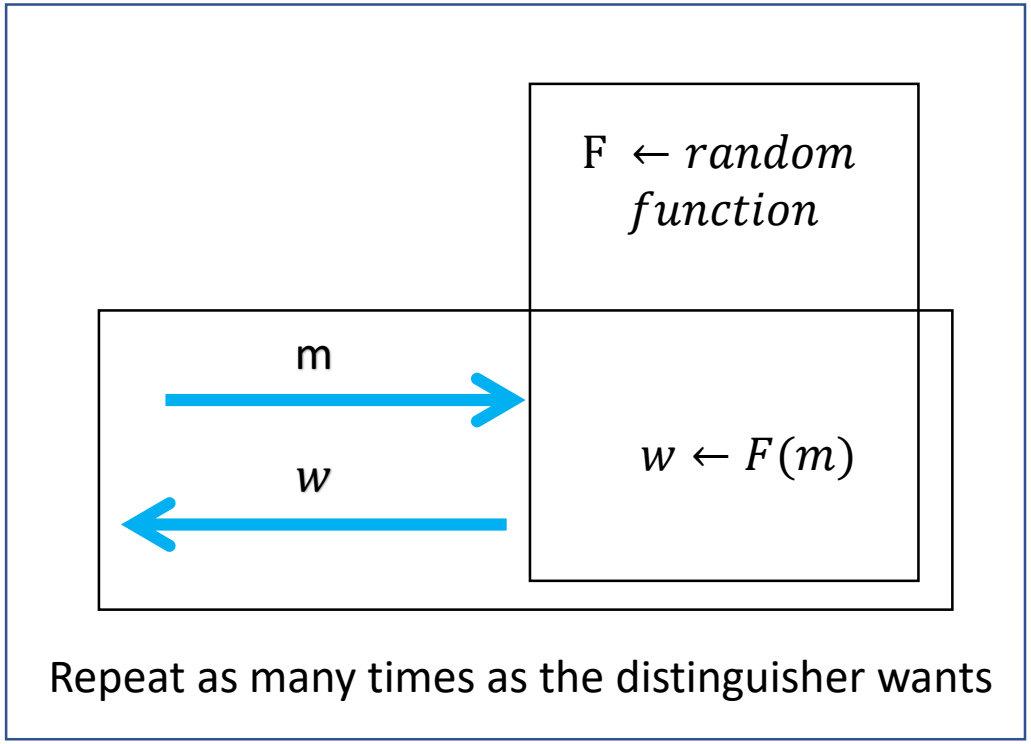

$G_1$

# Pseudo-random function

- A class of functions $(F_1, \dots, F_{2^n})$ is pseudo-random if the following two games are indistinguishable




$G_0$




$G_1$

# CPA-secure encryption scheme from PRF

- $Keygen(\{1\}^s)$
  - $k \in_R \{0,1\}^s$
- $Enc_k(m)$
  - $r \in_R \{0,1\}^n$
  - $c \leftarrow (r, F_k(r) \oplus m)$
- $Dec_k(c)$
  - $(r, d) \leftarrow c$
  - $m \leftarrow F_k(r) \oplus d$
- Important property $Dec(Enc(m) \oplus \epsilon) = m + \epsilon$

# Breaking security of the scheme using validation oracle

- Let the message space be M = {1100,0110,0101}
- Important property:
  - Let $(\mathrm{r}, \mathrm{v}) = Enc(m)$ then $Dec(r, v \oplus \epsilon) = m \oplus \epsilon$
- Given validation oracle
  - Consider what happens if we decrypt $(r, v \oplus r, v \oplus \epsilon) \oplus (0, x)$ with $\epsilon = 0011$
    - $1100 \rightarrow 1111 \notin M$
    - $0110, 0101 \rightarrow 0110,\ 0110 \in M$

# Why do we care about the validation oracle

- When people encrypt messages and send it to servers, it is typical that if the decrypted message does not have the right format it returns an error

- Original PKCS paper (detailing how to use Crypto in the real world) had an attack where the attacker can modify the ciphertext and learn one bit depending on if an error received a message

# General format of a validation attack

- Take the message space M
- Generate a modification of the ciphertext which maps certain encrypted messages back to the ciphertext and others not
- Especially useful if the encryption scheme is homomorphic:

  $(Enc, Dec)$ is homomorphic if there exists $\odot$,$\otimes$ such that

$$Enc(m_1 \odot m_2) = Enc(m_1) \otimes Enc(m_2)$$

# Some homomorphic encryption scheme

- Especially useful if the encryption scheme is homomorphic:
  $(Enc, Dec)$ is homomorphic if there exists $\odot$,$\otimes$ such that
  $$Enc(m_1 \odot m_2) = Enc(m_1) \otimes Enc(m_2)$$

- One-time pad
  - $\odot := \oplus$
  - $\otimes := \oplus$

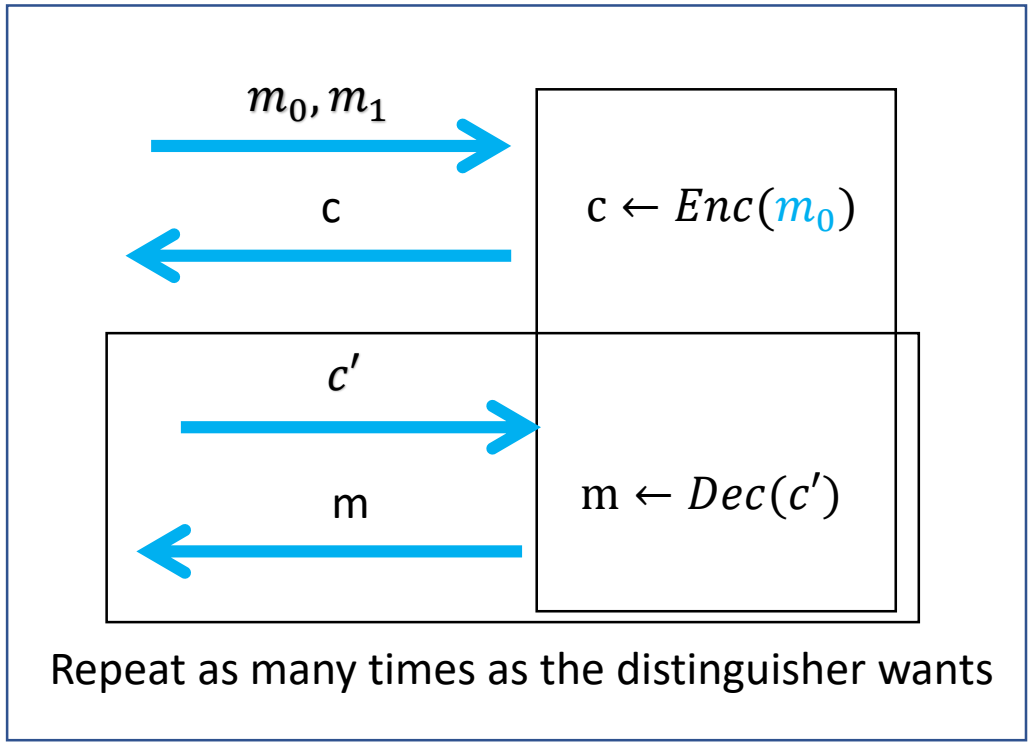
- RSA, El-gammal
  - $\odot := +$
  - $\otimes := \times$

# Authenticated encryption

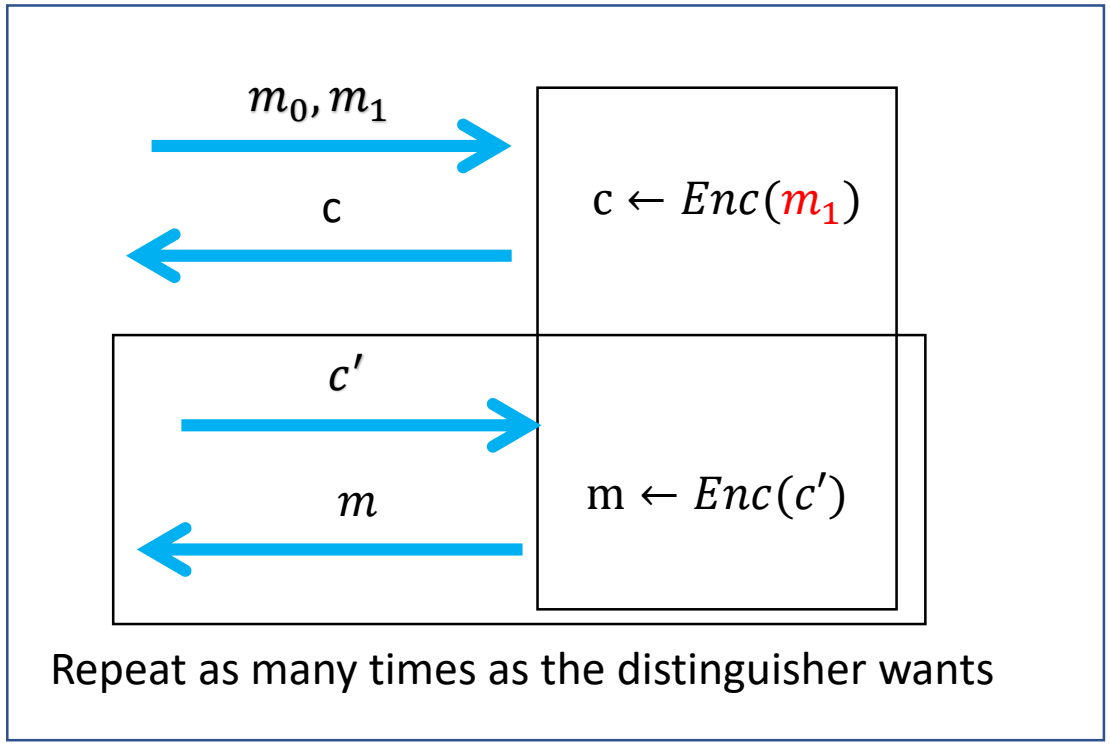
- Authenticated encryption
  - Authenticated (adversary cannot forge a ciphertext)
  - Encrypted  (adversary cannot learn message)

# Chosen-ciphertext game

Distinguisher loses automatically if $c = c'$

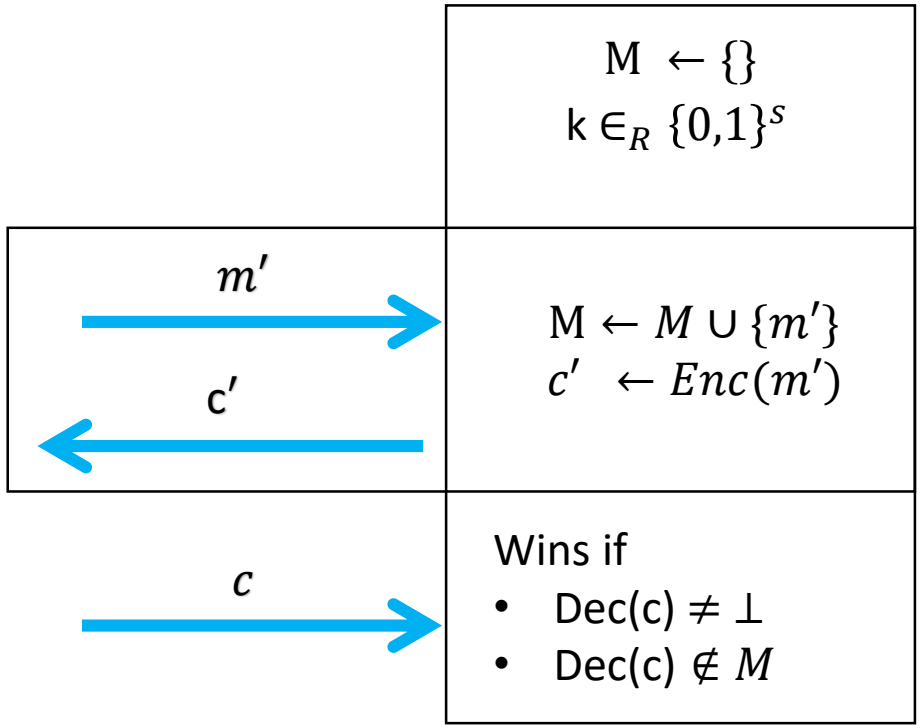



$G_0$




$G_1$

# Unforgeability game



# Authenticated encryption

- An Encryption scheme $(Gen, Enc, Dec)$ is an authenticated encryption scheme if
  - Unforgeable
  - CCA-secure

# Three Candidates for AE from mac + enc

- We assume
  - $Enc$ is a secure encryption scheme
  - $Auth$ is a secure message authentication code

| Encrypt-and-mac | encrypt-then-mac | Mac-then-encrypt |
|---|---|---|
| $\overline{Enc}(m)$<br>$c' \leftarrow Enc(m)$<br>$t \leftarrow Auth(m)$<br>$c \leftarrow (c', t)$ | $\overline{Enc}(m)$<br>$c' \leftarrow Enc(m)$<br>$t \leftarrow Auth(c')$<br>$c \leftarrow (c', t)$ | $\overline{Enc}(m)$<br>$t \leftarrow Auth(m)$<br>$w \leftarrow (m, t)$<br>$c \leftarrow Enc(w)$ |

- Show which two are insecure and which is secure, here are the hint
  - $Enc'(m_1 \| m_2) = Enc(m_1) \| Enc(m_2)$ is a secure encryption scheme
  - Auth'(m) = (auth(m),m) is a secure encryption scheme

# Insecure schemes

- Encrypt and mac

**Encrypt-and-mac**

$\overline{Enc}(m)$
$c' \leftarrow Enc(m)$
$t \leftarrow Auth(\mathrm{m})$
$c \leftarrow (c', t)$

- Answer: if authentication leaks the message, then this encryption scheme also leaks the message

# Insecure schemes

- Mac then encrypt

**Encrypt-and-mac**

$\overline{Enc}(m)$
$c' \leftarrow Enc(m)$
$t \leftarrow Auth(m)$
$c \leftarrow (c', t)$

- Answer: Let $(c_0, t_0) = \overline{Enc}(m_0), (c_2, t_2) = \overline{Enc}(m')$

- We have that $Dec(c_2, t_0) = m_0$ if and only if $c_2 = Dec(m_0)$

# Authenticated encryption with associated (public) data

- $Enc(data, m) \rightarrow (data, c, t)$
- Correctness:
  - $Dec(data, Enc(data, m)) = (data, m)$
- Authentication
  - Impossible to create a fresh pair such that:
    - $(data, c')$ has been seen before
    - Dec $(data', c', t') \neq \perp$
- Indistinguishability

# Galois-counter mode

- Combines Information theoretic mac with counter-mode
  - Uses one-time mac over binary field.
- $Enc(data, m)$
  - $IV \in_R \{0,1\}^{n/2}$
  - $m_1, \dots, m_d \leftarrow m$
  - $A_1, \dots, A_\ell \leftarrow data$
  - $k_1 \leftarrow E_k(IV, 0)$
  - $k_2 \leftarrow E_k(IV, 1)$
  - $c_1, \dots, c_d \leftarrow CTR(m_1, \dots, m_n ; counter = (IV \,||\, 2))$
  - $\tau \leftarrow otm(d, \ell, data, c_1, \dots, c_d)$
  - $\mathrm{c} \leftarrow (d, \ell, data, c_1, \dots, c_d, \tau)$