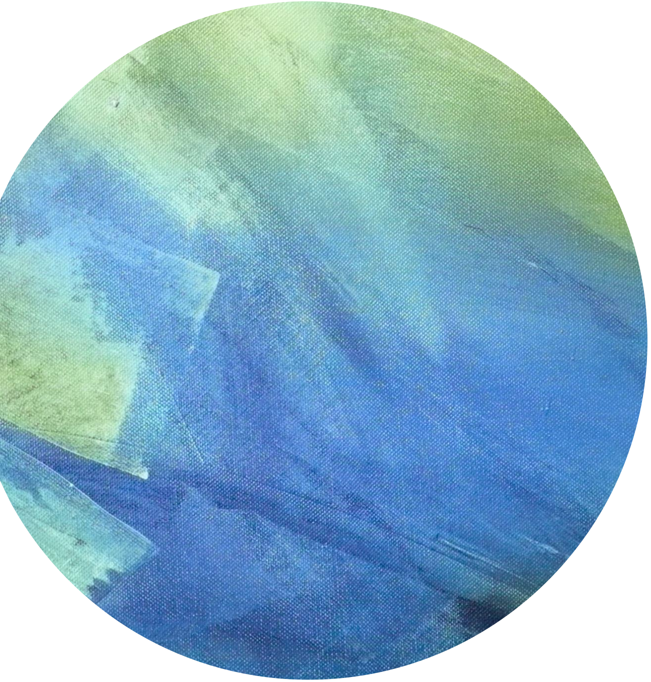
Lecture 6



Outline

Encrypting many messages

MACs

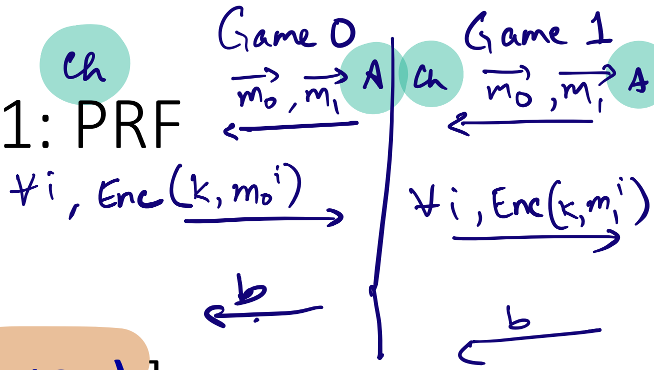
Message Authentication Code

Administrative Details

- Participation credit begins today.
- You get a point each time you ask an interesting question or answer a question that I ask or answer a question well on piazza.
- If you get 5 or more points, you get 100% participation credit.
- Scribe volunteer? Claire

Encrypting large messages

Many-time Encryption, Soln 1: PRF



$\left(b \mid \mathrm{Game}_0\right) \approx_c \left(b \mid \mathrm{Game}_1\right)$

negl in size of Encryption key

Let $F: K \times R \longrightarrow M$ be a secure PRF.

For $m \in M$ define $\mathrm{Enc}(k,m) = [\ r \leftarrow R,$ output $(r, F(k,r) \oplus m)\ ]$

Is Enc semantically secure under CPA?

To encrypt m, output

$ct = (r, \mathrm{PRF}(k,r) \oplus m).$

$m_0^{(1)} \neq m_1^{(1)} \Rightarrow (m_0^{(2)}, m_1^{(2)})_L \neq (m_0^{(2)}, m_1^{(2)})_R$

Adaptive many-msg Games:

Game$_0$

Ch. | Adv

$k \leftarrow K$

$\leftarrow m_0^{(1)}, m_1^{(1)}$

$\text{Enc}(k, m_0^{(1)}) \rightarrow$

$\leftarrow m_0^{(2)}, m_1^{(2)}$

$\text{Enc}(k, m_0^{(2)}) \rightarrow$

⋮

$\leftarrow b$

Game$_1$

Ch. | Adv

$k \leftarrow K$,

$\leftarrow m_0^{(1)}, m_1^{(1)}$

$\text{Enc}(k, m_1^{(1)}) \rightarrow$

$\leftarrow m_0^{(2)}, m_1^{(2)}$

$\text{Enc}(k, m_1^{(2)}) \rightarrow$

$\leftarrow b$

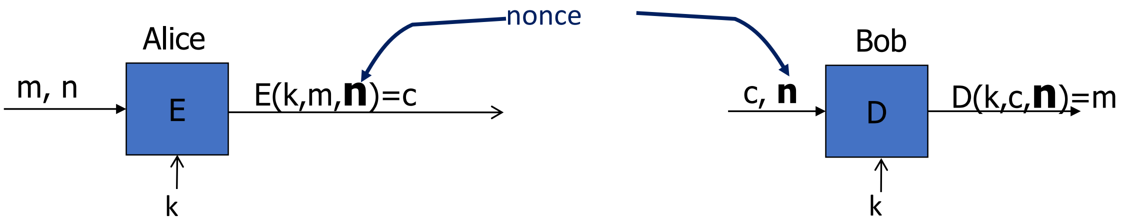
H.W. Prove that (E, D) where

$$E(k, m) = (r, \text{PRF}(k, r) \oplus m)$$

$$D(k, (c_1, c_2)) = \text{PRF}(k, c_1) \oplus c_2$$

is adaptive many-time secure.

Solution 2: nonce-based Encryption



$E(k, m[1], 00...01)$

$E(k,(m[2], 0....10))$

- nonce n: a value that changes from msg to msg.
  (k,n) pair never used more than once

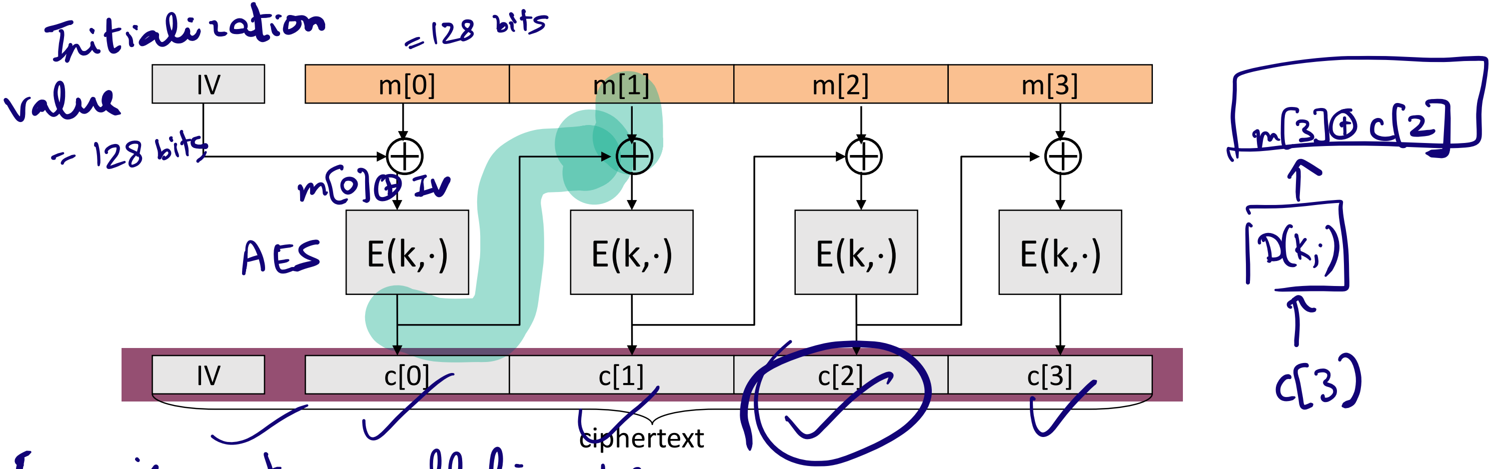
- method 1: nonce is a **counter** (e.g. packet counter) [SSL, IPSec]
  - used when encryptor keeps state from msg to msg
  - if decryptor has same state, need not send nonce with CT

- method 2: nonce is **random** [File Encryption]

$m = m[0] \| m[1] \| m[2] \| m[3] \ldots$

CBC (Cipher Block Chaining) mode

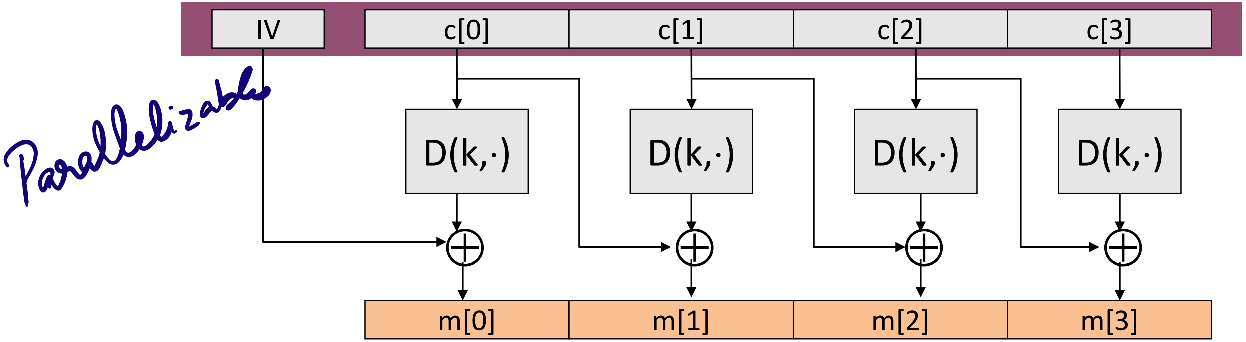
Let (E,D) be a PRP. $E_{CBC}(k,m)$: choose **random** Initialization Vector and do:



Enc is not parallelizable.

Decryption Circuit

In symbols: $c[0] = E\big(k, IV \oplus m[0]\big) \Rightarrow m[0] = D\big(k, c[0]\big) \oplus IV$



CPA Security of CBC

- <u>CBC Theorem</u>: For small enough L>0,
  If E is a secure PRP over (K,X) then
  E_{CBC} is CPA-secure over (K, X^L, X^{L+1}).

(adaptive, many-time)

Birthday attack

messages in vector

- In particular, security error in CBC = (2 x sec. error in PRP) + ($q^2L^2/|X|$)

blocks in a message

- What if IV was predictable? Is it still CPA-secure?

Bug in SSL/TLS 1.0: IV for record #i is last CT block of record #(i-1)

What happens if adversary can predict IV

Adaptive, many-msg games

Game$_0$

Ch | Adv

$k \leftarrow K$

$\xleftarrow{m_0^{(1)},\ m_1^{(1)}}$

$\xrightarrow{Enc(k, m_0^{(1)})}$

$\xleftarrow{m_0^{(2)},\ m_1^{(2)}}$

$\xrightarrow{Enc(k, m_0^{(2)})}$

Game$_1$

$\xleftarrow{m_0^{(1)},\ m_1^{(1)}}$

$\xrightarrow{Enc(k, m_1^{(1)})}$

$\xleftarrow{m_0^{(2)},\ m_1^{(2)}}$

$\xrightarrow{Enc(k, m_1^{(2)})}$

CBC- mode

What happens if adversary can predict IV

Adaptive, many - msg games

Game$_0$

Ch — Adv

$k \leftarrow K$

$0, 0$ ←

$IV_1, E(k, IV_1)$ →

$m_0^{(2)} = IV_1 \oplus IV_2$ | $m_1^{(2)} = 0$ ←

$IV_2, E(k, (IV_2 \oplus m))$ →

$= IV_1 \oplus IV_2$

$= E(k, IV_1)$

same?

Game$_1$

$0, 0$ ←

$IV_1, E(k, IV_1)$ →

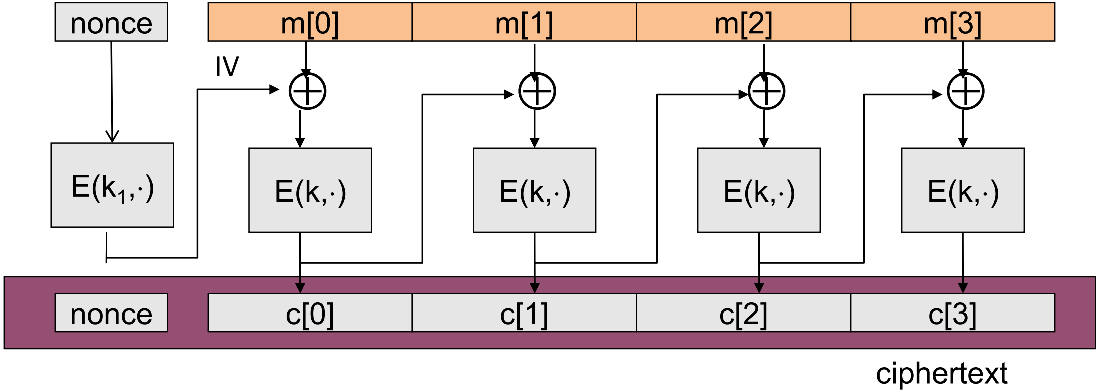
$m_0^{(2)} = IV_1 \oplus IV_2$ | $m_1^{(2)} = 0$ ←

$IV_2, E(k, IV_2)$ →

CBC (Cipher Block Chaining) mode: Version 2

- Cipher block chaining with <u>unique</u> nonce: key = (k,k_1)

unique nonce means: (key, n) pair is used for only one message



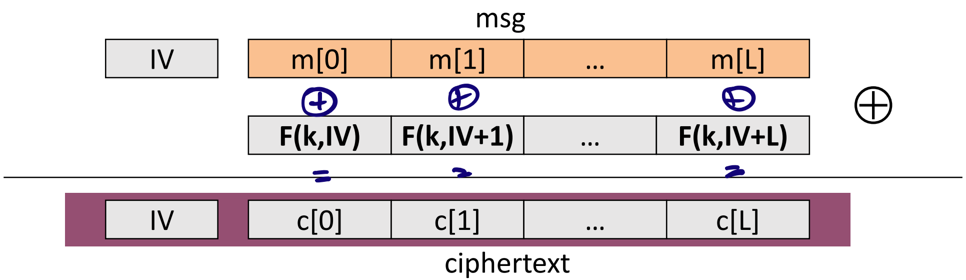
Random counter

Rand-ctr mode

parallelizable Enc.

Let F: K × $\{0,1\}^n \longrightarrow \{0,1\}^n$ be a secure *PRF*.

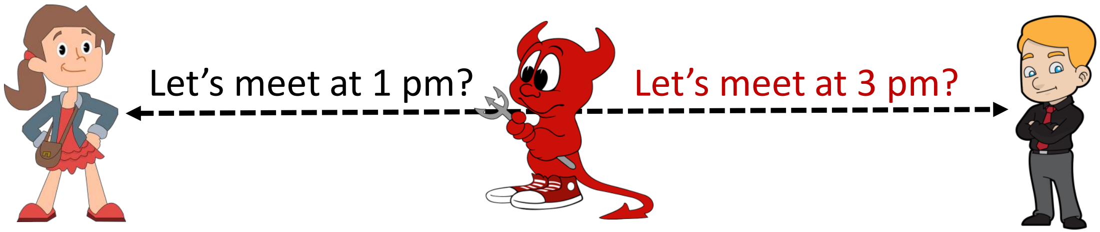
E(k,m): choose a random IV $\in \{0,1\}^n$ and do:



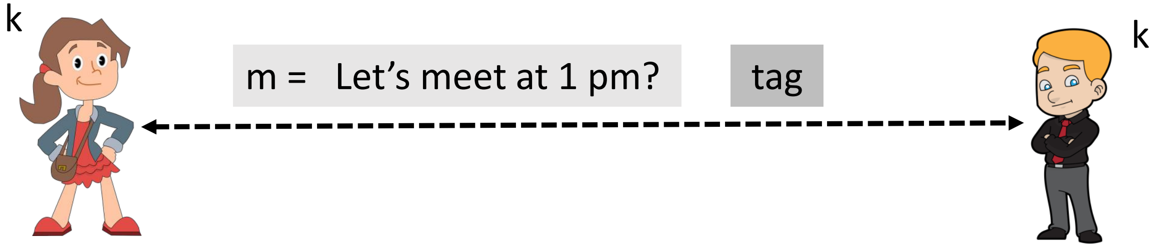
(1) Can use PRF instead of PRP and (2) is parallelizable.

Message Integrity: MACs

Goal: Integrity (not secrecy)



Shared key setting



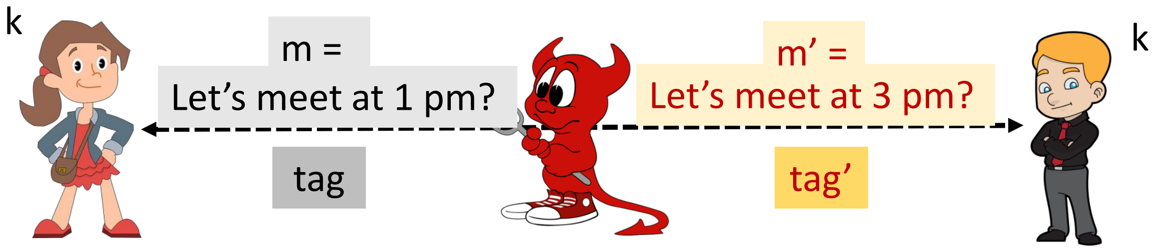
tag = S (k, m)

V (k, m, tag) = acc/rej

MAC = (S,V) is a pair of algorithms that satisfy:

1. CORRECTNESS. V(k, m, tag) = 1 when tag = S (k, m)

Shared key setting: security of MACs



MAC = (S,V) is a pair of algorithms that satisfy:

2. SECURITY. Attacker unable to compute (m’, tag’) different from (m, tag) such that

V (k, m, tag’) = 1

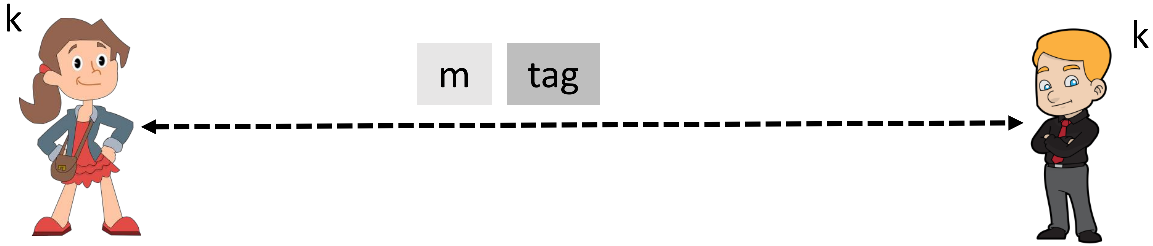
Shared key setting: security of MACs

Security, more generally:

- Attacker can demand tags (t_1, t_2, …. t_n) for messages (m_1, m_2, … m_n)
- Attacker wins if it outputs (m’, tag’) not in {(m_1, t_1), (m_2, t_2), …. (m_n, t_n)} such that V (k, m’, tag’) = 1

How do you build a secure MAC?

- Say *k* is a PRF key



tag = S (k, m)

V (k, m, tag) = acc/rej

How do you build a secure MAC?

- Say *k* is a PRF key
- S (k, m) = PRF (k, m)
- V (k, m, tag) = 1 *if and only if* tag = PRF (k, m)

- Correctness?

How do you build a secure MAC?

- S (k, m) = PRF (k, m)
- V (k, m, tag) = 1 *if and only if* tag = PRF (k, m)

- Security?

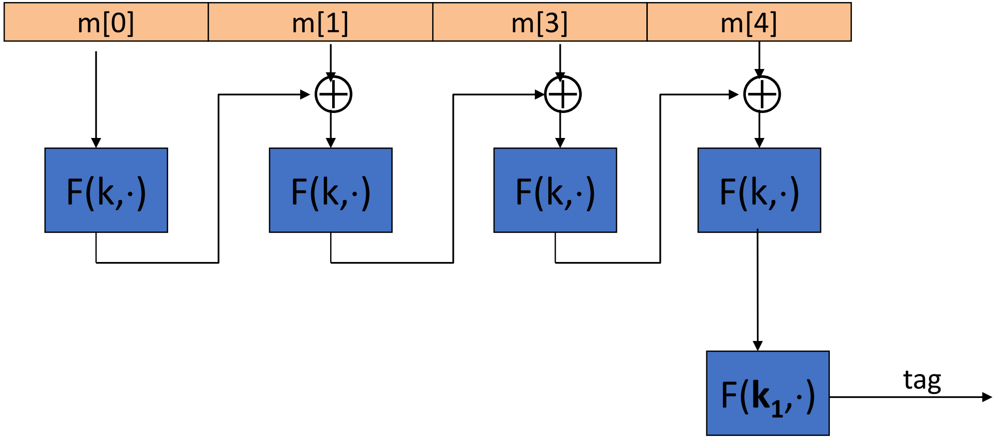
Examples

- AES: MAC for 64-bit messages

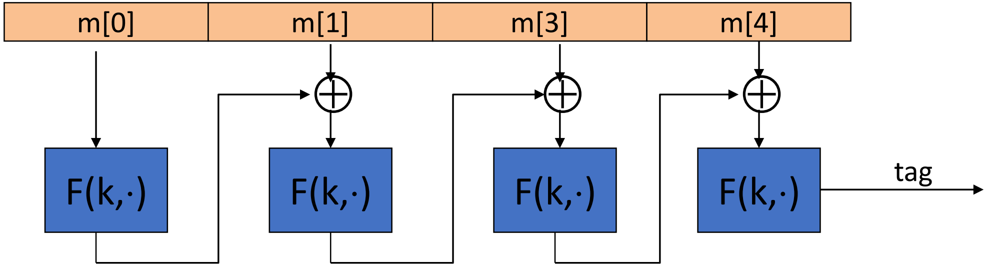
- What about larger messages?

- For larger messages, we use:
  - CBC-MAC
  - HMAC

Encrypted CBC-MAC

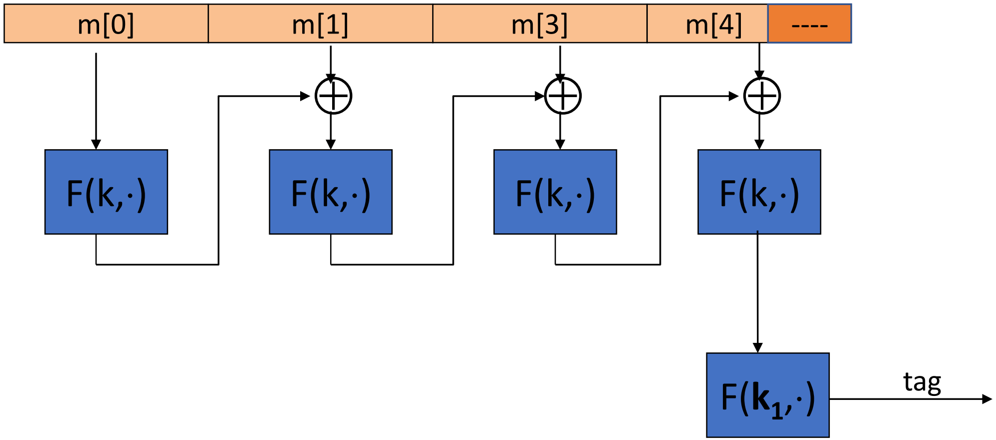


What happens if we remove encryption step?

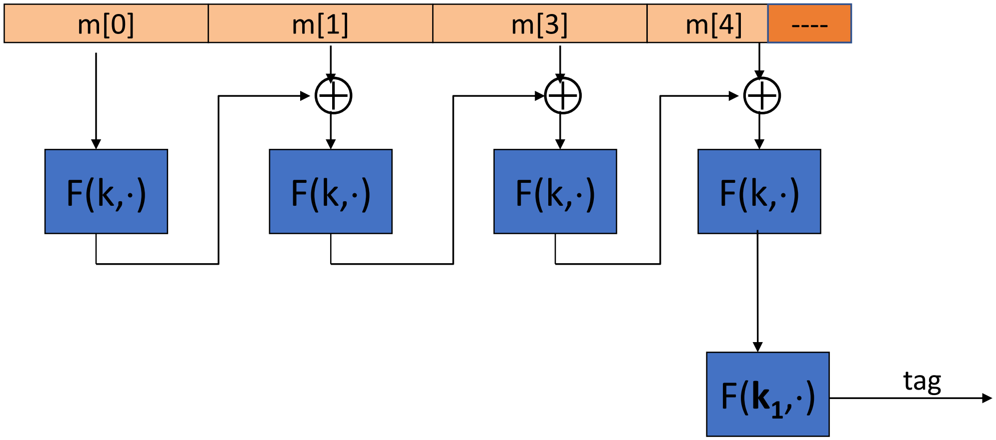


Obtain tag for m, then get t = F (k, m)
Output forgery (m’, t’) where m’ = (m||(t \xor m)) and t’ = t.

What if message length is not a multiple of block size?



First idea?



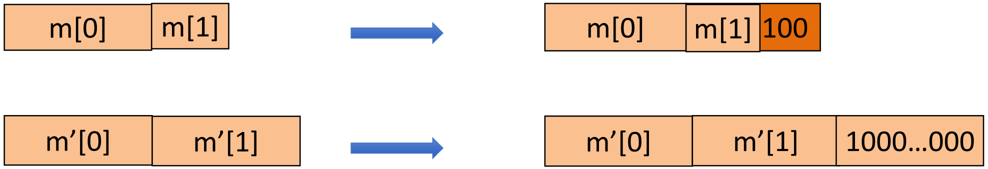
Need invertible padding!

For security, padding must be invertible !

$$m_0 \neq m_1 \quad \Rightarrow \quad pad(m_0) \neq pad(m_1)$$

Pad with "1000…00". Add new dummy block if needed.

- The "1" indicates beginning of pad.



Need invertible padding!

For security, padding must be invertible !

$$m_0 \neq m_1 \quad \Rightarrow \quad pad(m_0) \neq pad(m_1)$$

Pad with "1000…00". Add new dummy block if needed.

- The "1" indicates beginning of pad.