COS433/Math 473: Cryptography

Mark Zhandry

Princeton University

Spring 2017

Message Authentication Codes

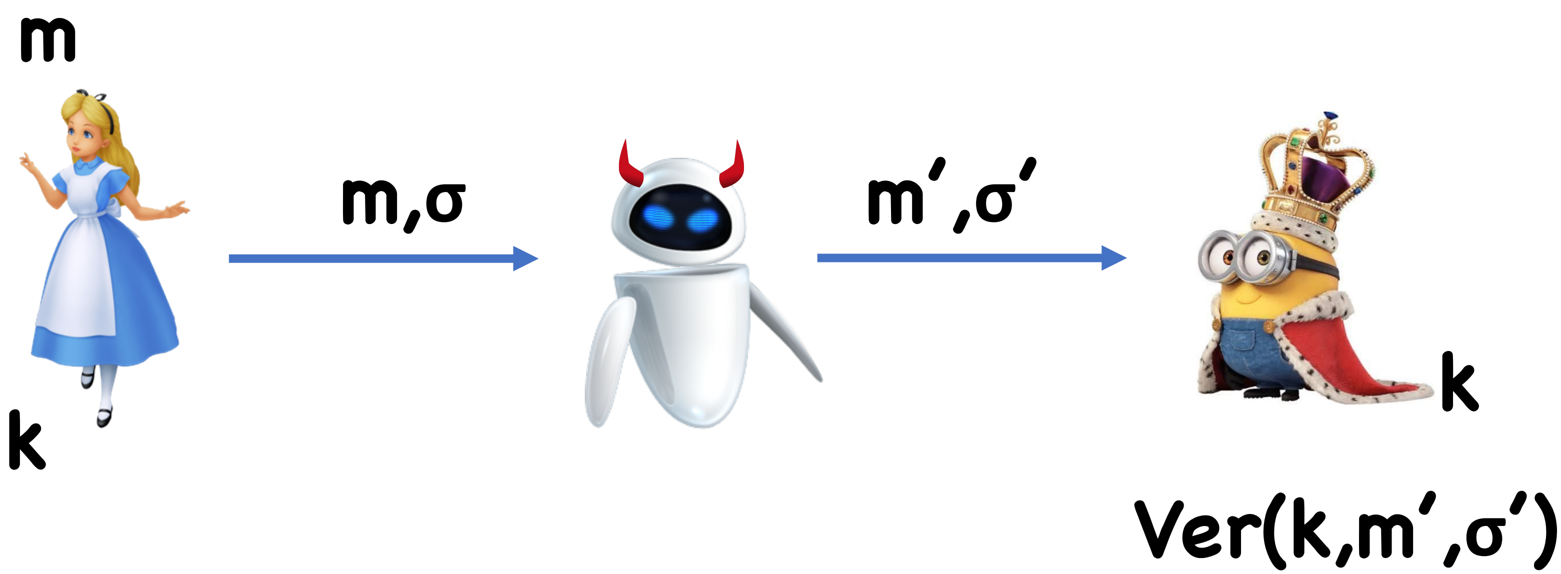
Syntax:

- Key space $\mathbf{K}_\lambda$
- Message space **M**
- Tag space $\mathbf{T}_\lambda$
- **MAC(k,m) → σ**
- **Ver(k,m,σ) → 0/1**

Correctness:

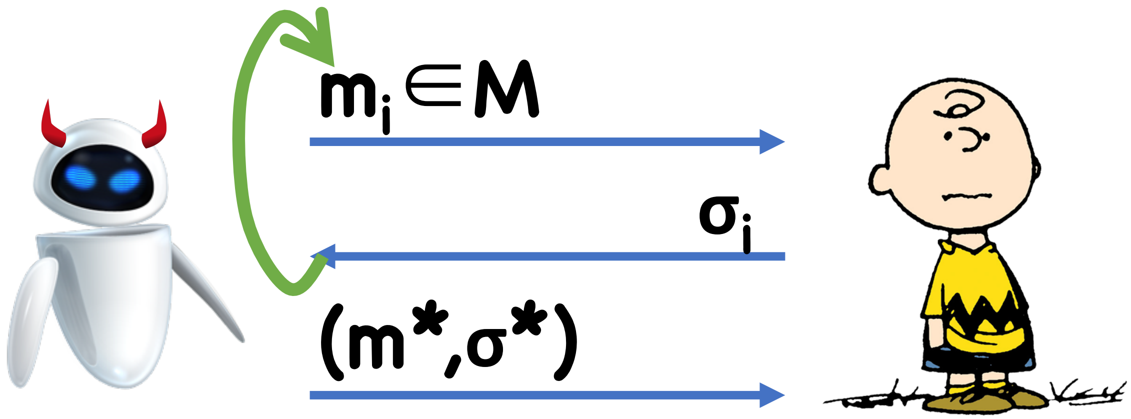
- $\forall$ **m,k, Ver(k,m, MAC(k,m)) = 1**

Message Authentication Codes



Goal: If Eve changed **m**, Bob should reject

Security For MACs

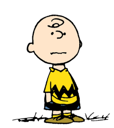


$k \leftarrow K_\lambda$

$\sigma \leftarrow MAC(k, m_i)$

Output 1 iff:

- $m^* \notin \{m_1, \ldots\}$
- $Ver(k, m^*, \sigma^*) = 1$

CMA-Adv( , λ) = Pr[ outputs 1]

Constructing MACs

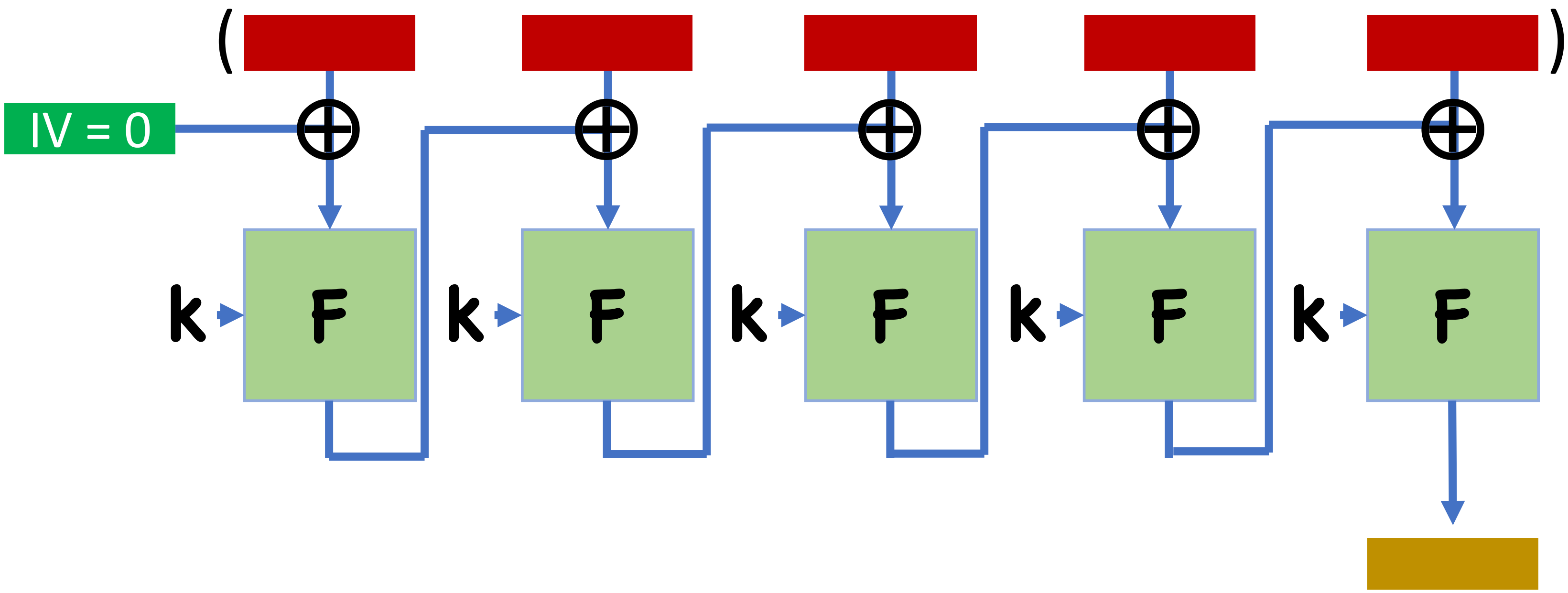
Use a PRF

$F: K \times M \rightarrow T$

$MAC(k,m) = F(k,m)$
$Ver(k,m,\sigma) = (F(k,m) == \sigma)$

Theorem: (MAC,Ver) is CMA secure assuming **$1/|T|$** is negligible

CBC-MAC



Theorem: CBC-MAC is a secure PRF for fixed-length messages

Today

Improving Efficiency of MACs

Authenticated Encryption: combining secrecy and integrity

Improving efficiency

Limitations of CBC-MAC

Many block cipher evaluations

Sequential

Carter Wegman MAC

$k' = (k,h)$ where:

- k is a PRF key for $F:K \times R \rightarrow Y$
- h is sampled from a pairwise independent function family

MAC(k',m):

- Choose a random $r \leftarrow R$
- Set $\sigma = (r, F(k,r) \oplus h(m))$

Theorem: The Carter Wegman MAC is strongly CMA secure

Proof

Assume toward contradiction a PPT 

Hybrids…

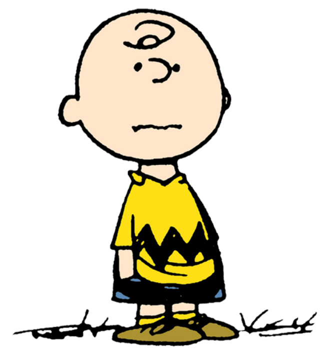
Proof

<u>Hybrid 0</u>

$m_i \in M$

$\sigma_i = (r_i, t_i)$

(m^*, r^*, t^*)



$k \leftarrow K$

h

$r_i \leftarrow R$

$t_i \leftarrow F(k,r) \oplus h(m)$

Output 1 iff:

- $(m^*, r^*, t^*) \notin \{(m_i, r_i, t_i)\}$
- $F(k, r^*) \oplus h(m^*) = t^*$

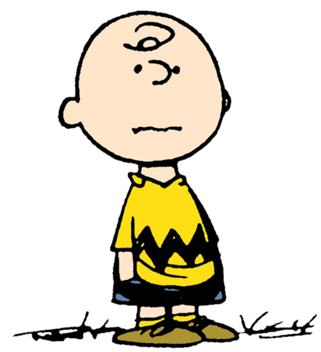
Proof

<u>Hybrid 1</u>

$m_i \in M$

$\sigma_i = (r_i, t_i)$

(m^*, r^*, t^*)



$k \leftarrow K$

h

(Distinct $\mathbf{r_i}$)

$r_i \leftarrow R$

$t_i \leftarrow F(k,r) \oplus h(m)$

Output 1 iff:

- $(m^*, r^*, t^*) \notin \{(m_i, r_i, t_i)\}$
- $F(k, r^*) \oplus h(m^*) = t^*$

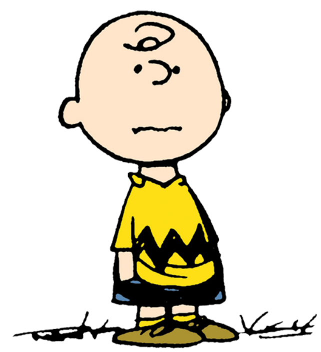
Proof

Hybrid 2

$m_i \in M$ →

← $\sigma_i = (r_i, t_i)$

(m^*, r^*, t^*) →



$H \leftarrow Funcs$

h

(Distinct r_i)

$r_i \leftarrow R$

$t_i \leftarrow H(r) \oplus h(m)$

Output 1 iff:

- $(m^*, r^*, t^*) \notin \{(m_i, r_i, t_i)\}$
- $H(r^*) \oplus h(m^*) = t^*$

Proof

Claim: In Hybrid 2, negligible success probability

Possibilities:

- $r^* \notin \{r_i\}$: then value of $H(r^*)$ hidden from adversary, so $Pr[H(r^*) \oplus h(m^*) = t^*]$ is $1/|Y|$

- $r^* = r_i$ for some i: then $m^* \neq m_i$ (why?)
  h completely hidden from adversary
  $Pr[H(r^*) \oplus h(m^*) = t^*]$
  $= Pr[h(m^*) = t^* \oplus t_i \oplus h(m_i)] = 1/|Y|$

Proof

Hybrid 1 and 2 are indistinguishable

- PRF security

Hybrid 0 and 1 are indistinguishable

- W.h.p. random $\mathbf{r}_i$ will be distinct

Therefore, negligible success probability in Hybrid 0

Efficiency of CW MAC

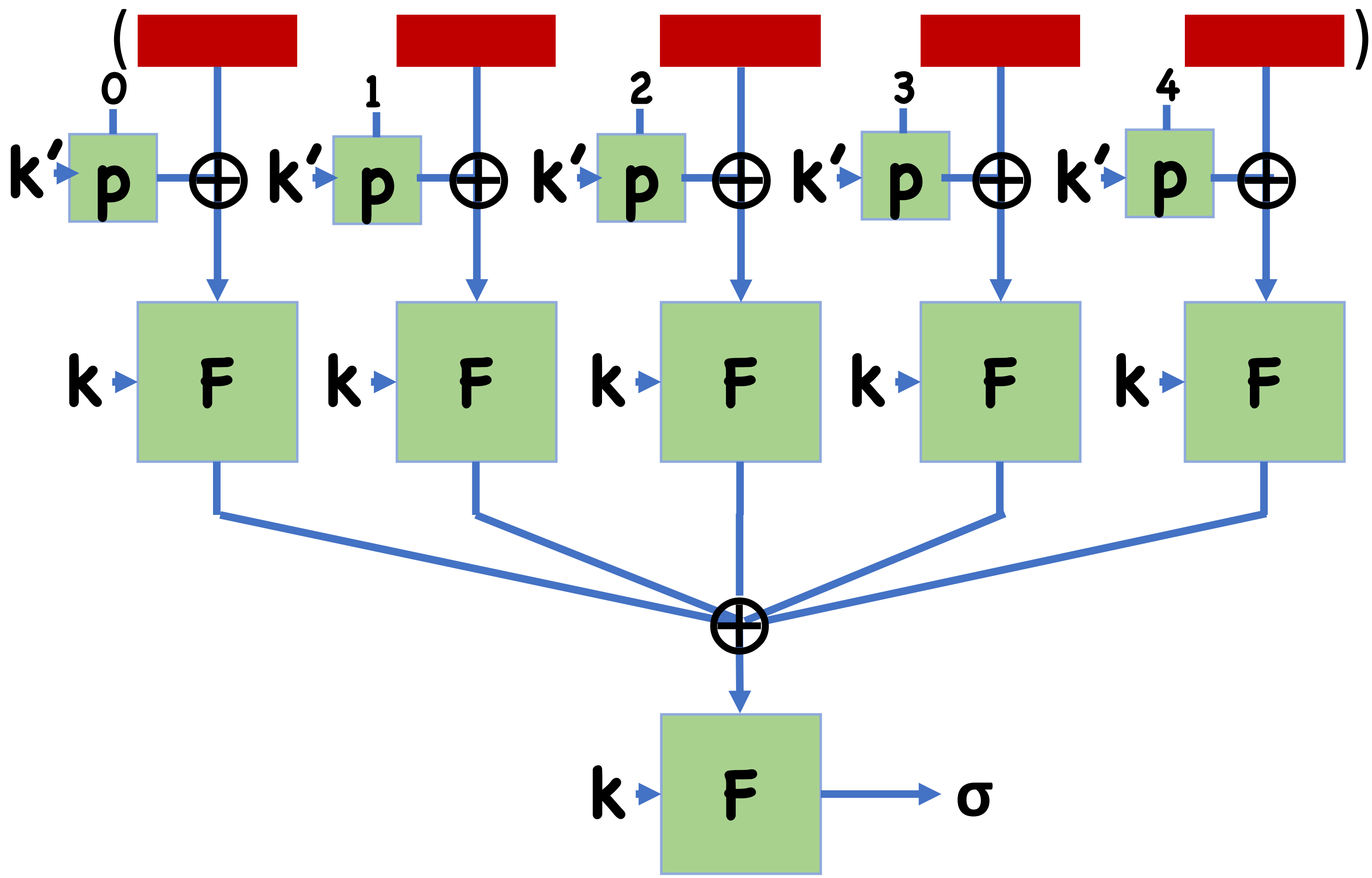
MAC(k',m):

- Choose a random **r←R**
- Set **σ = (r, F(k,r)⊕h(m))**

h much more efficient that PRFs

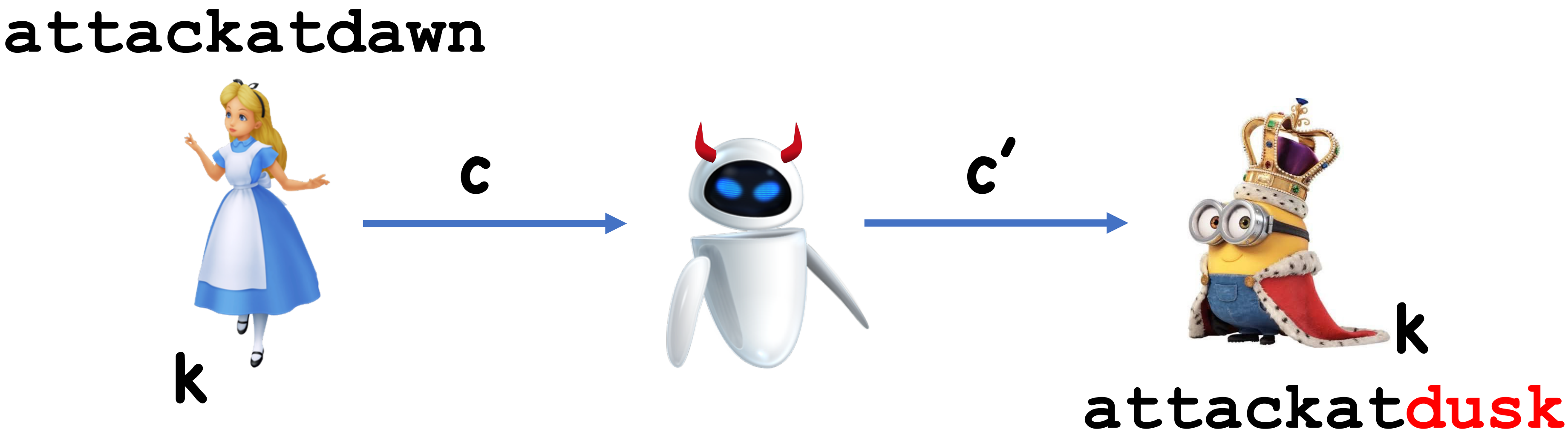
PRF applied only to small nonce **r**
h applied to large message **m**

PMAC: A Parallel MAC



Authenticated Encryption

Authenticated Encryption



Goal: Eve cannot learn nor change plaintext

- Authenticated Encryption will satisfy two security properties

Syntax

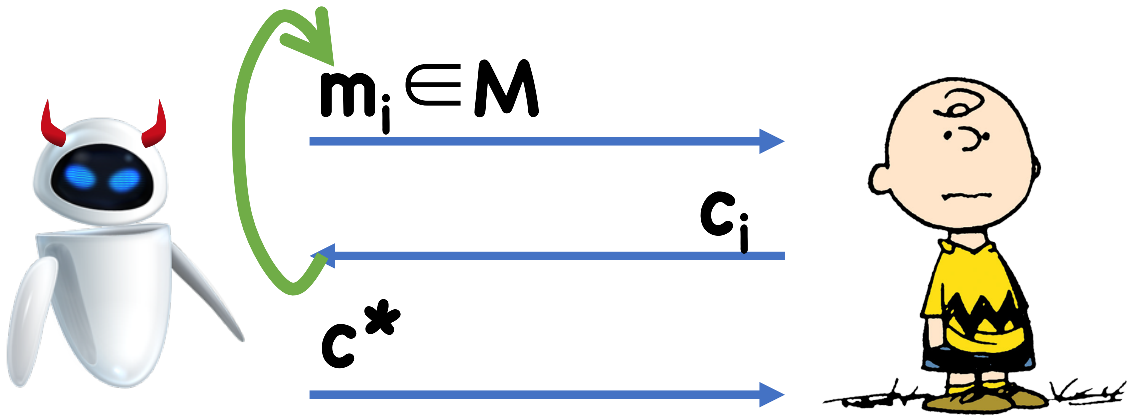
Syntax:

- $\mathbf{Enc: K \times M \rightarrow C}$
- $\mathbf{Dec: K \times C \rightarrow M \cup \{\perp\}}$

Correctness:

- For all $\mathbf{k \in K,\ m \in M,\ Dec(k,\ Enc(k,m)\ ) = m}$

Unforgeability



$k \leftarrow K_\lambda$

$c \leftarrow Enc(k,m_i)$

Output 1 iff:

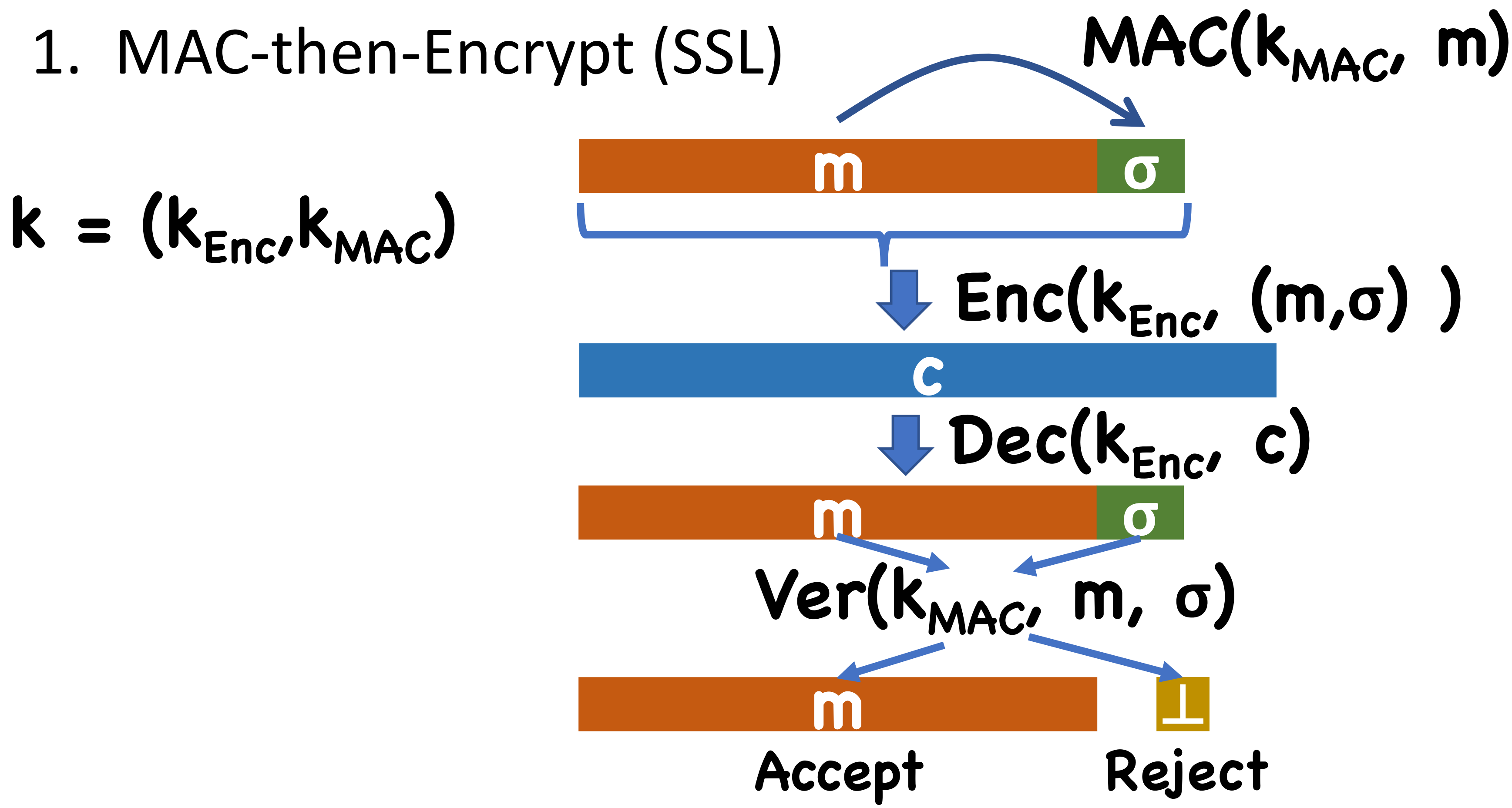
- $c^* \notin \{c_1,\ldots\}$
- $Dec(k,c^*) \neq \perp$

Definition: An encryption scheme **(Enc,Dec)** is an **authenticated encryption scheme** if it is unforgeable and CPA secure

Constructing Authenticated Encryption

Three possible generic constructions:

1. MAC-then-Encrypt (SSL)

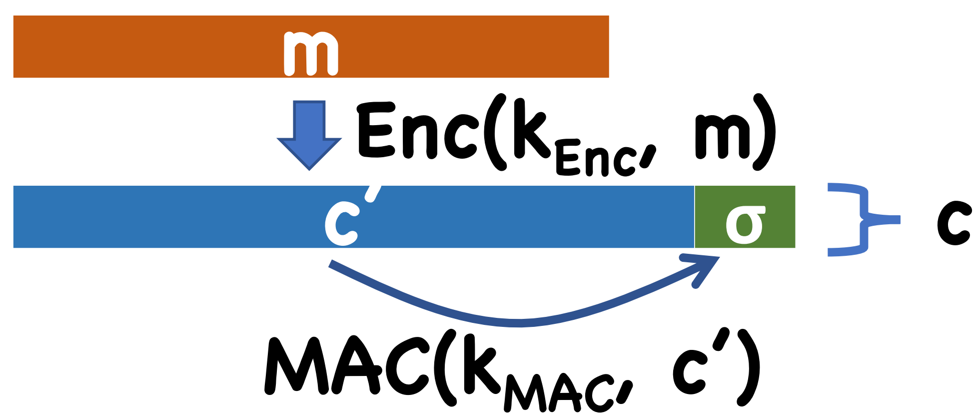


Constructing Authenticated Encryption

Three possible generic constructions:

2. Encrypt-then-MAC (IPsec)

$k = (k_{Enc}, k_{MAC})$

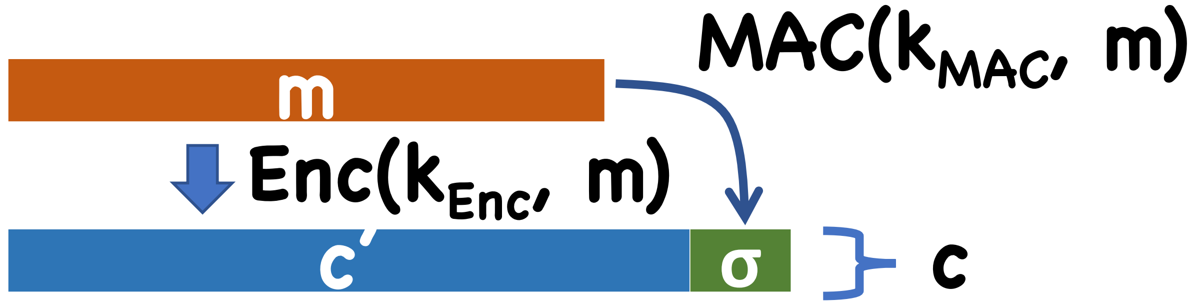


Constructing Authenticated Encryption

Three possible generic constructions:

3. Encrypt-and-MAC (SSH)

$k = (k_{Enc}, k_{MAC})$



Constructing Authenticated Encryption

1. MAC-then-Encrypt

2. Encrypt-then-MAC

3. Encrypt-and-MAC

Which one(s) **always** provides authenticated encryption (assuming strongly secure MAC)?

Constructing Authenticated Encryption

MAC-then-Encrypt?

Constructing Authenticated Encryption

Encrypt-then-MAC?

Constructing Authenticated Encryption

Encrypt-and-MAC?

Constructing Authenticated Encryption

Just because MAC-then-Encrypt and Encrypt-and-MAC are insecure for *some* MACs/encryption schemes, they may be secure in some settings

Ex: MAC-then-Encrypt with CTR or CBC encryption

- For CTR, any one-time MAC is actually sufficient

Theorem: MAC-then-Encrypt with any one-time MAC and CTR-mode encryption is an authenticated encryption scheme

Proof

CPA security: straightforward

- CPA security of encryption scheme guarantees message + mac is hidden

Proof

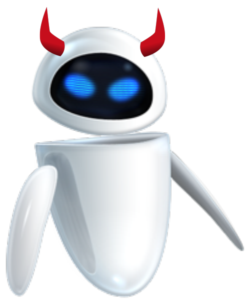
Integrity: assume towards contradiction a PPT ciphertext forger 

Hybrids…

Proof

Hybrid 0:

$k_{MAC} \leftarrow K_{MAC}$
$k_{PRF} \leftarrow K_{PRF}$



$m_i \in M$ →

← (r_i, c_i)

(r^*, c^*) →

$\sigma_i \leftarrow MAC(k_{MAC}, m_i)$
$r_i \leftarrow R$
$c_i \leftarrow F(k_{PRF}, r) \oplus (m_i, \sigma_i)$

Output 1 iff:

- $(r^*, c^*) \notin \{(r_1, c_1), \ldots\}$
- $Ver(k_{MAC}, m^*, \sigma^*) = 1$ where $(m^*, \sigma^*) \leftarrow F(k_{PRF}, r^*) \oplus c^*$

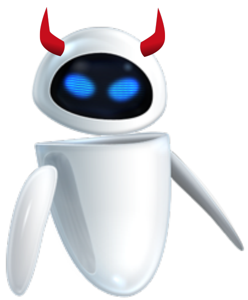
Standard forgery experiment

Proof

Hybrid 1:

$k_{MAC} \leftarrow K_{MAC}$

$H \leftarrow Funcs$



$m_i \in M$ →

← (r_i, c_i)

(r^*, c^*) →

$\sigma_i \leftarrow MAC(k_{MAC}, m_i)$

$r_i \leftarrow R$

$c_i \leftarrow H(r) \oplus (m_i, \sigma_i)$

Output 1 iff:

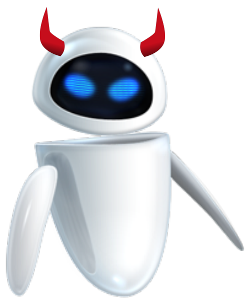
- $(r^*, c^*) \notin \{(r_1, c_1), \ldots\}$
- $Ver(k_{MAC}, m^*, \sigma^*) = 1$ where $(m^*, \sigma^*) \leftarrow H(r^*) \oplus c^*$

Proof

<u>Hybrid 2</u>:

$k_{MAC} \leftarrow K_{MAC}$

$H \leftarrow Funcs$



$m_i \in M$ →

← (r_i,c_i)

(r^*,c^*) →

$\sigma_i \leftarrow MAC(k_{MAC},m_i)$

$r_i \leftarrow R$ **(distinct)**

$c_i \leftarrow H(r) \oplus (m_i,\sigma_i)$

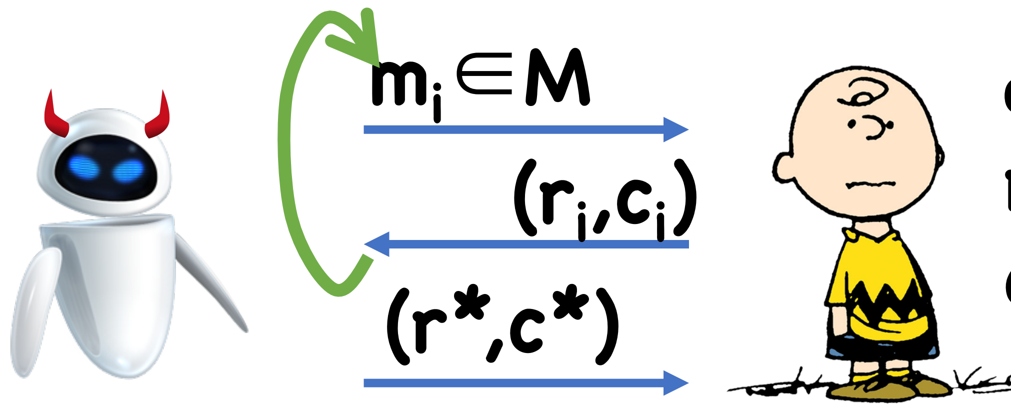
Output 1 iff:

- $(r^*,c^*) \notin \{(r_1,c_1),\ldots\}$
- $Ver(k_{MAC}, m^*,\sigma^*)=1$ where $(m^*,\sigma^*) \leftarrow H(r^*) \oplus c^*$

Proof

<u>Hybrid 3</u>:

$k_{MAC} \leftarrow K_{MAC}$
$H \leftarrow Funcs$



$\sigma_i \leftarrow MAC(k_{MAC}, m_i)$
$r_i \leftarrow R$ (distinct)
$c_i \leftarrow H(r) \oplus (m_i, \sigma_i)$

Output 1 iff:

- $(r^*, c^*) \notin \{(r_1, c_1), \ldots\}$
- $r^* \in \{r_1, \ldots\}$
- $Ver(k_{MAC}, m^*, \sigma^*) = 1$ where $(m^*, \sigma^*) \leftarrow H(r^*) \oplus c^*$

Proof

Hybrid 0 and Hybrid 1 are indistinguishable by PRF security

Hybrid 1 and Hybrid 2 are indistinguishable since the r's are distinct with overwhelming probability

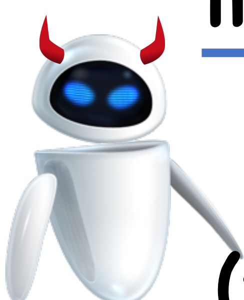
Hybrid 2 and Hybrid 3 are indistinguishable since if $\mathbf{r^*} \notin \{\mathbf{r_1},\ldots\}$, then $\mathbf{H(r^*)}$ hidden from adversary's view

- For any $\mathbf{c^*}$, $\mathbf{(m^*,\sigma^*)=H(r^*)\oplus c^*}$ truly random

    → forgery with negligible probability

Proof

Suppose non-negligible prob of forgery in Hyb 3



$\mathbf{m_i}$ →

← $\mathbf{(r_i,c_i)}$

$\mathbf{(r^*,c^*)}$ →

Pick random $\mathbf{i^* \in \{1,\dots,q\}}$

If $\mathbf{i \neq i^*}$, choose random $\mathbf{(r_i,c_i)}$

If $\mathbf{i = i^*}$,

- Choose random $\mathbf{r_{i^*}}$, $\mathbf{k^*}$
- $\mathbf{c_{i^*} \leftarrow k^* \oplus (m_i,\sigma)}$

If $\mathbf{r^* \neq r_{i^*}}$, abort

If $\mathbf{r^* = r_{i^*}}$,

- $\mathbf{(m^*,\sigma^*) \leftarrow k^* \oplus c^*}$

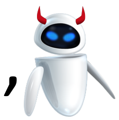
$\mathbf{m_i}$ →

← $\boldsymbol{\sigma}$

$\mathbf{(m^*,\sigma^*)}$ →

Proof

Analysis

- Regardless of which **i***  picks,  sees truly random ciphertexts (with distinct **r**)
- Therefore, **i*** independent of view of 
-  forges exactly when  forges AND guessed correct **i***
- ⇒ Prob  forges is non-negligible

Chosen Ciphertext Attacks

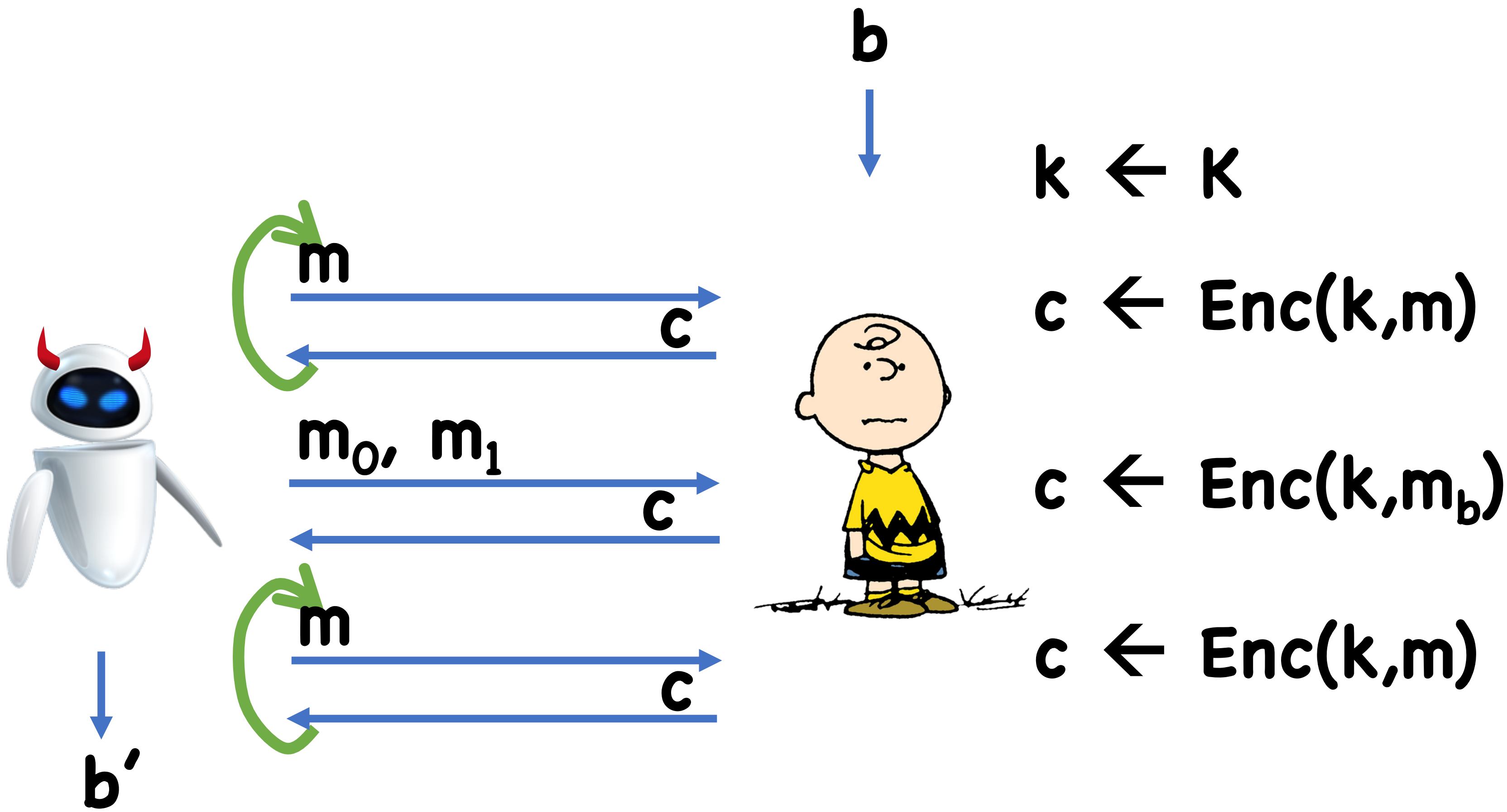
Chosen Ciphertext Attacks

Often, adversary can fool server into decrypting certain ciphertexts

Even if adversary only learns partial information (e.g. whether ciphertext decrypted successfully), can use info to decrypt entire message

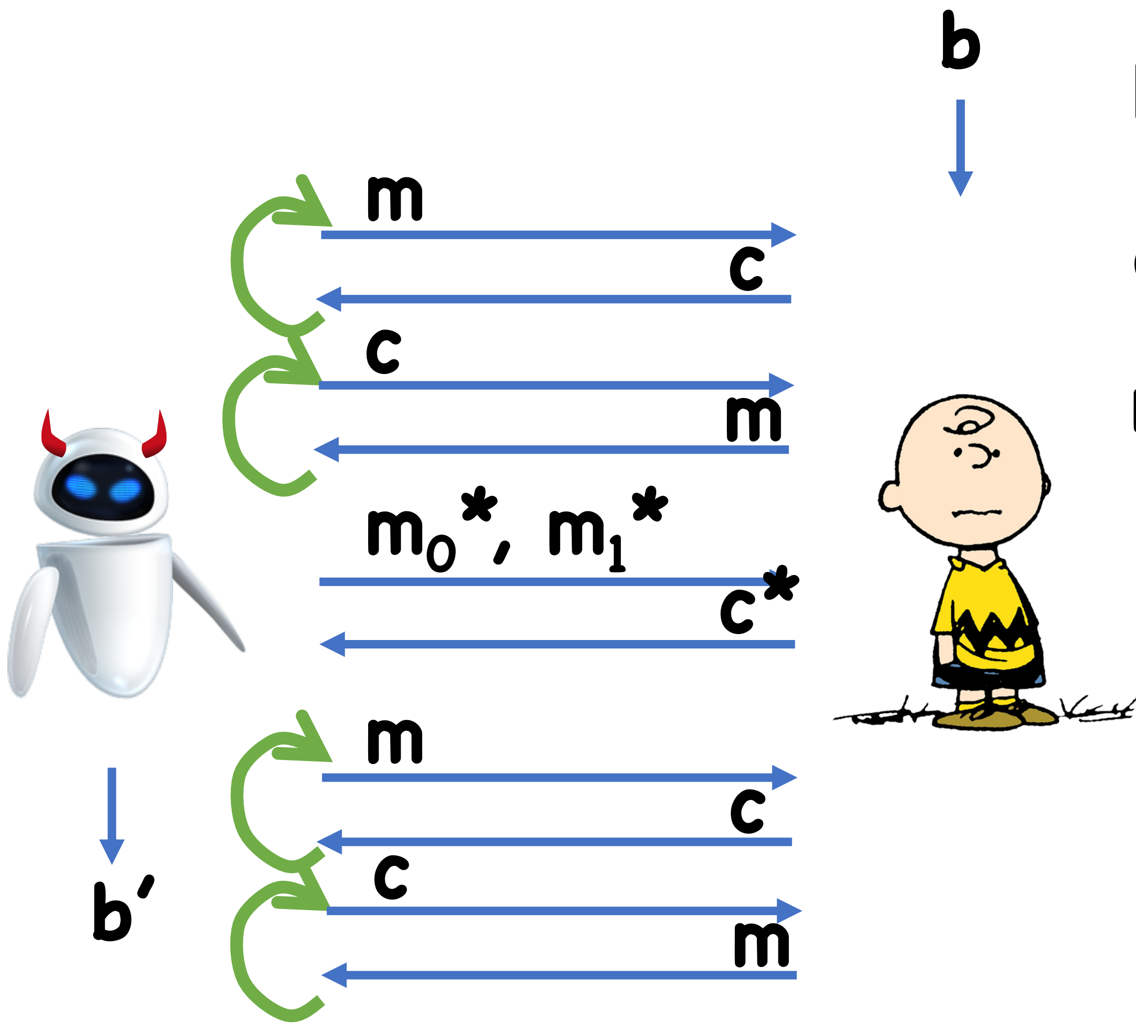
Therefore, want security even if adversary can mount decryption queries

Chosen Plaintext Security



CPA-Exp$_b$( , λ)

Chosen Ciphertext Security?



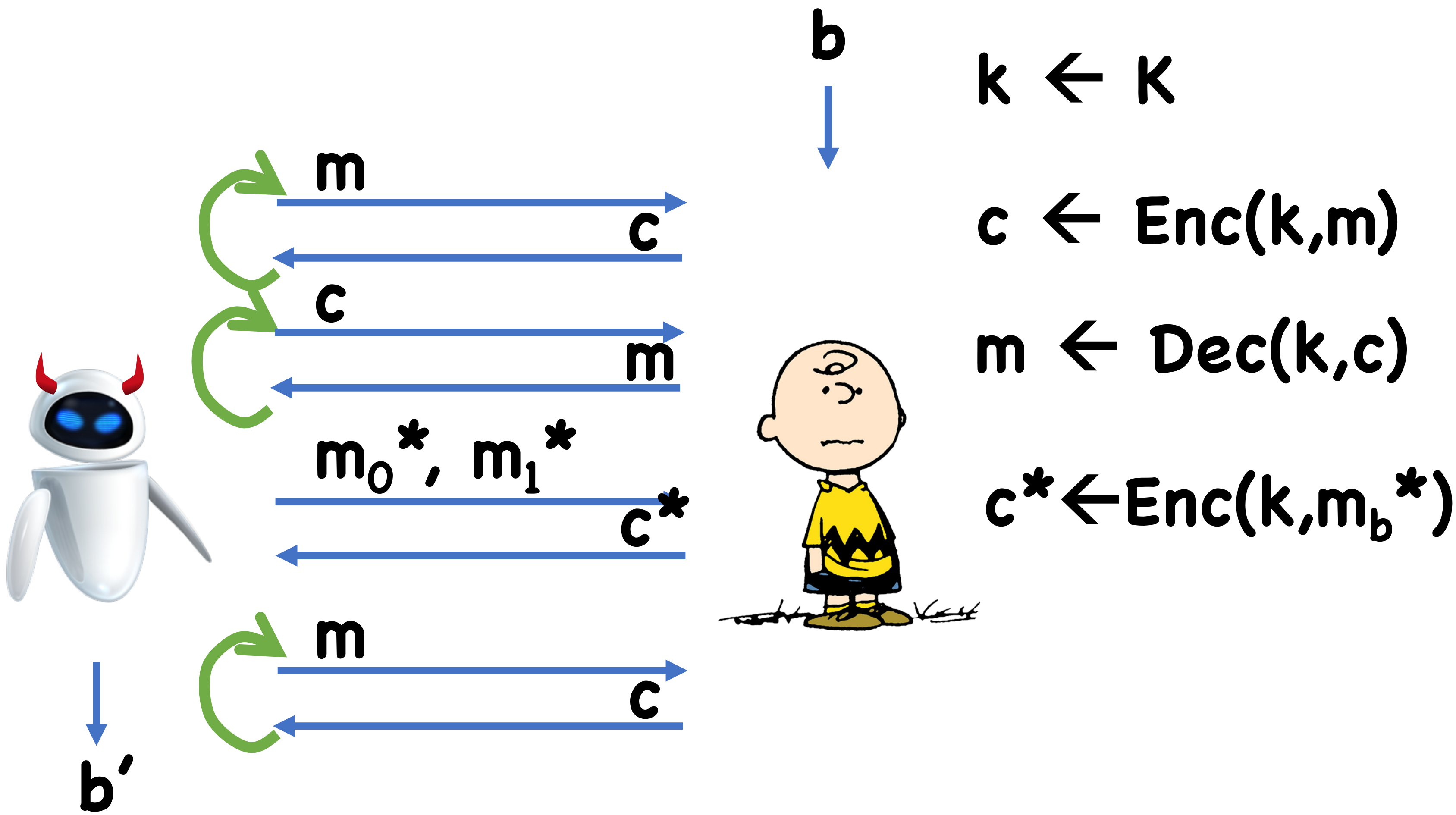
$k \leftarrow K$

$c \leftarrow Enc(k,m)$

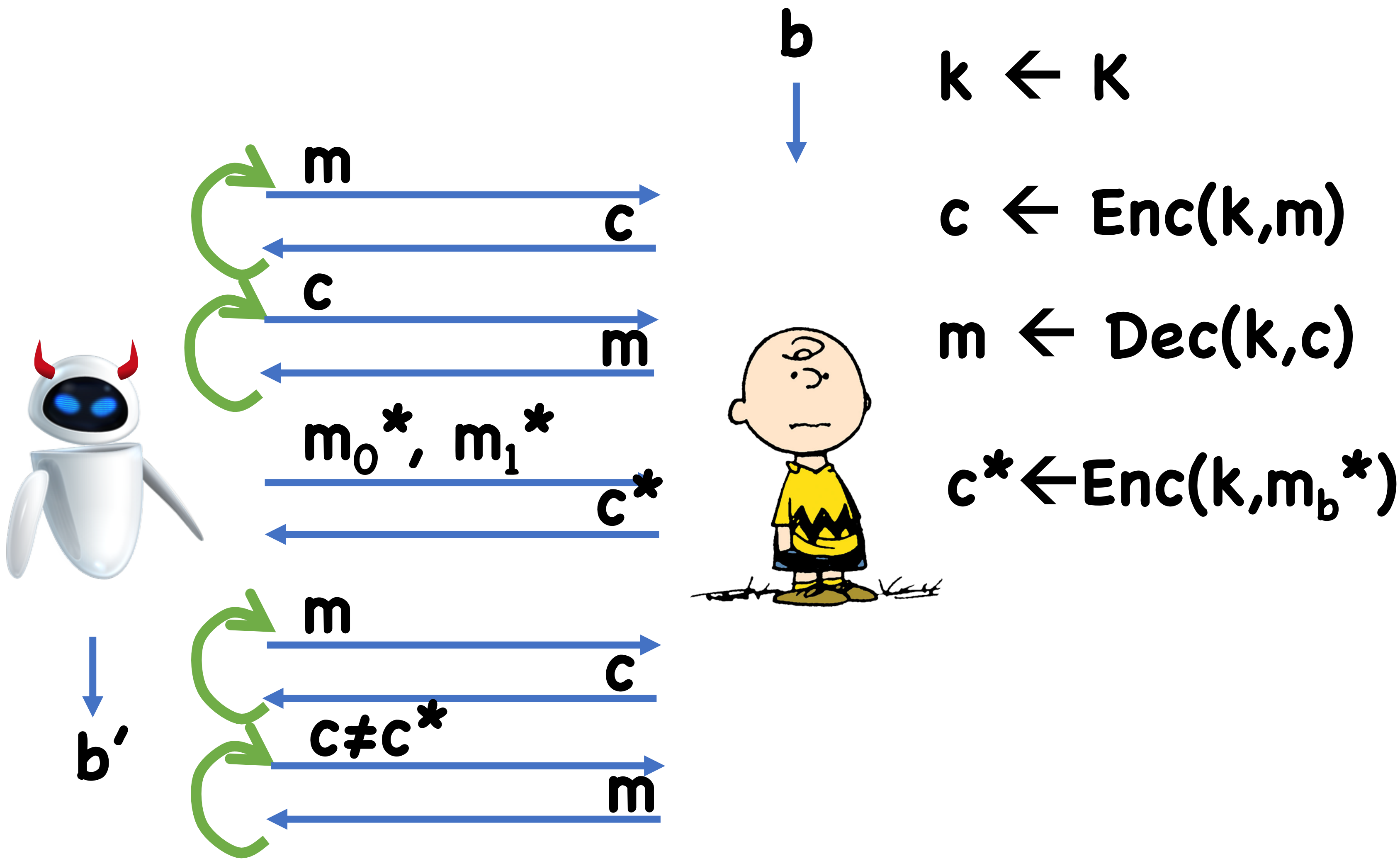
$m \leftarrow Dec(k,c)$

$c^* \leftarrow Enc(k,m_b^*)$

Lunch-time CCA (CCA1)



Full CCA (CCA2)



Theorem: If **(Enc,Dec)** is an authenticated encryption scheme, then it is also CCA secure

Proof Sketch

For any decryption query, two cases

1. Was the result of a CPA query
- In this case, we know the answer already!

2. Was not the result of an encryption query
- In this case, we have a ciphertext forgery

CCA vs Auth Enc

We know Auth Enc implies CCA security

What about the other direction?

For now, always strive for Authenticated Encryption

MAC-then-Encrypt with CBC

Even though MAC-then-Encrypt is secure for CBC encryption (which we did not prove), still hard to implement securely

Recall: need padding for CBC

Therefore, two possible sources of error

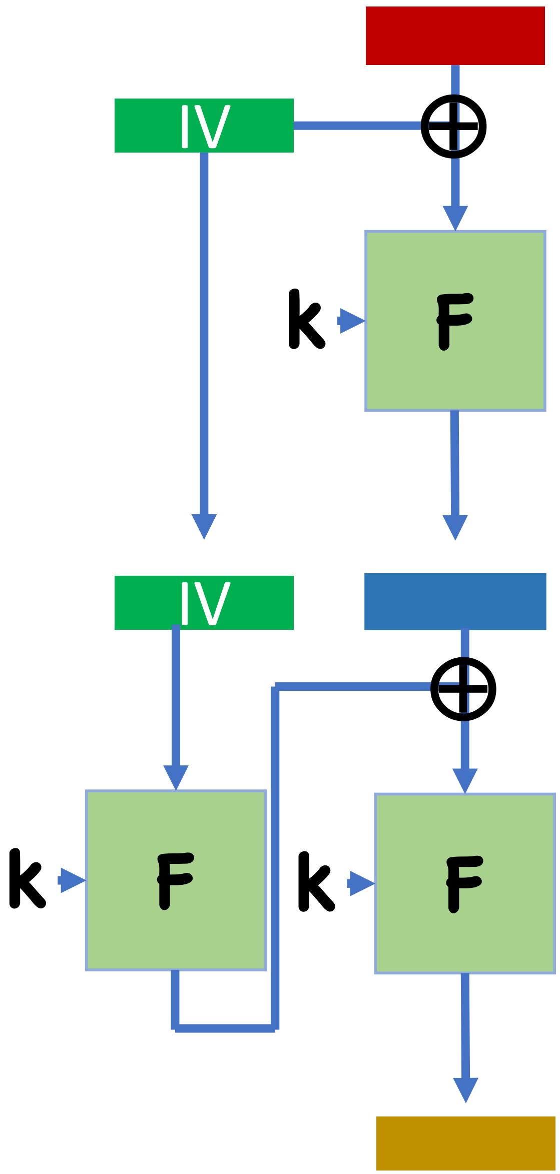
- Padding error
- MAC error

If possible to tell which one, then can attack

Using Same Key for Encrypt and MAC

Suppose we're combining CBC encryption and CBC-MAC

Can I use the same key for both?



Attack?

Using Same Key for Encrypt and MAC

In general, do not use same key for multiple purposes

- Schemes may interact poorly when using the same key

However, some modes of operation do allow same key to be used for both authentication and encryption

CCM Mode

CCM = Counter Mode with CBC-MAC in Authenticate-then-Encrypt combination

Possible to show that using same key for authentication and encryption still provides security

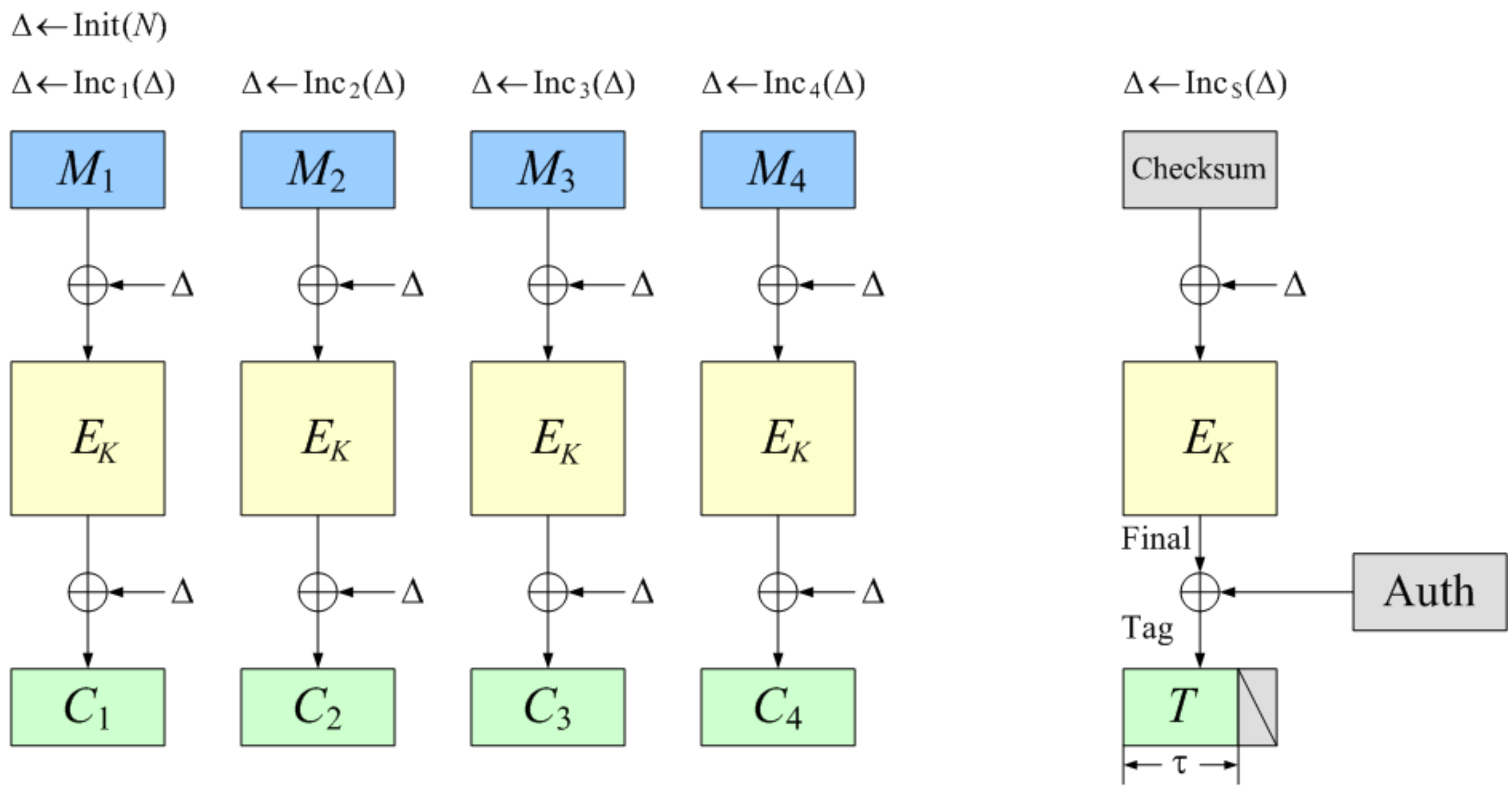
Efficiency

So far, all modes seen require two block cipher operations per block

- 1 for encryption
- 1 for authentication

Ideally, would have only 1 block cipher op per block

OCB Mode



Δ←Init(N)
Δ←Inc1(Δ)
Δ←Inc2(Δ)
Δ←Inc3(Δ)
Δ←Inc4(Δ)
Δ←IncS(Δ)
M1
M2
M3
M4
Checksum
Δ
EK
Final
Auth
Tag
C1
C2
C3
C4
T
τ

OCB Mode

Twice as fast as other block cipher modes of operation

However, not used much in practice

- Patents!

Other Modes

GCM: Roughly CTR mode then Carter-Wegman MAC

EAX: CTR mode then CMAC (variant of CBC-MAC)

Deterministic Encryption

Deterministic Encryption

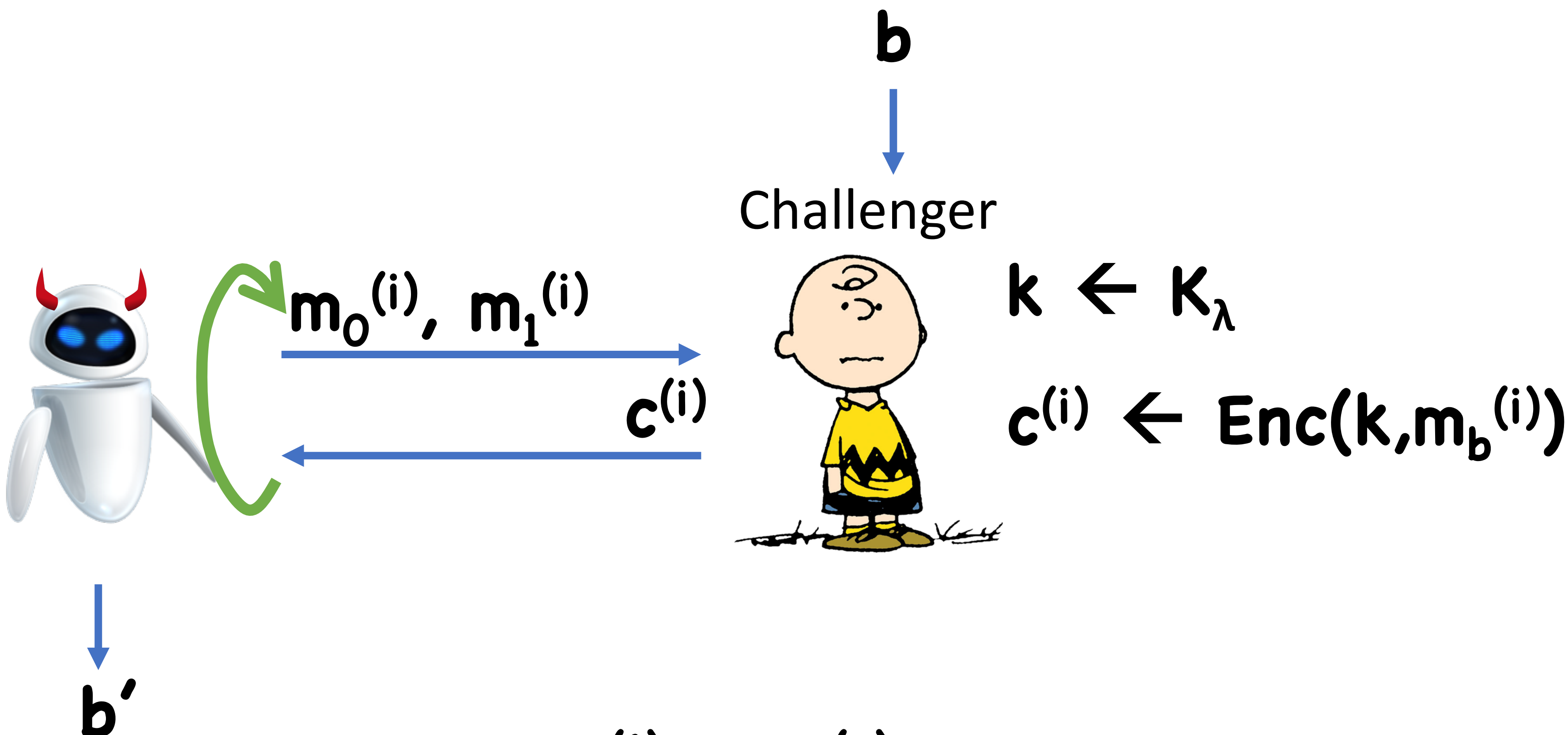
So far, we have insisted on CPA/CCA/Auth Enc security, which implies scheme must be randomized

However, sometimes deterministic encryption is necessary

- E.g. encrypting database records

How to resolve discrepancy?

Deterministic CPA Security



Where $\mathbf{m_1^{(1)},\ldots,m_1^{(q)}}$ are distinct
and $\mathbf{m_1^{(1)},\ldots,m_1^{(q)}}$ are distinct

Achieving Det. CPA Security

Idea? used fixed det. IV

- CTR mode?
- CBC mode?

Better options:

- Derive IV as **IV = PRF(k',m)**
  - If using Auth Enc, get Det. Auth Enc
- Use "large" PRP: **c = PRP(k,m)**
  - Can get Det. Auth Enc by padding message

Next Time

Collision resistant hashing

Reminder: Starting at 3pm, midterm will be posted on Blackboard (though not on course webpage)

- Due 1pm on Wednesday