COS433/Math 473: Cryptography

Mark Zhandry

Princeton University

Spring 2020

Announcements/Reminders

HW2 Due Feb 27^{th}
HW3 Due March 5^{th}

PR1 Due March 10^{th}

Previously on COS 433…

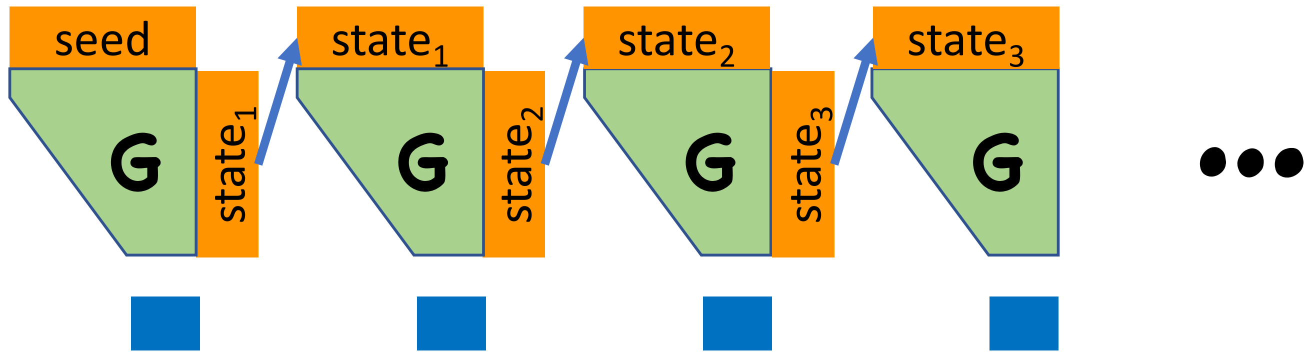
Length Extension for PRGs

Suppose I give you a PRG $\mathbf{G:\{0,1\}^{\lambda} \rightarrow \{0,1\}^{\lambda+1}}$

On it's own, not very useful: can only compress keys by 1 bit

But, we can use it to build PRGs with *arbitrarily-long* outputs!

Extending the Stretch of a PRG

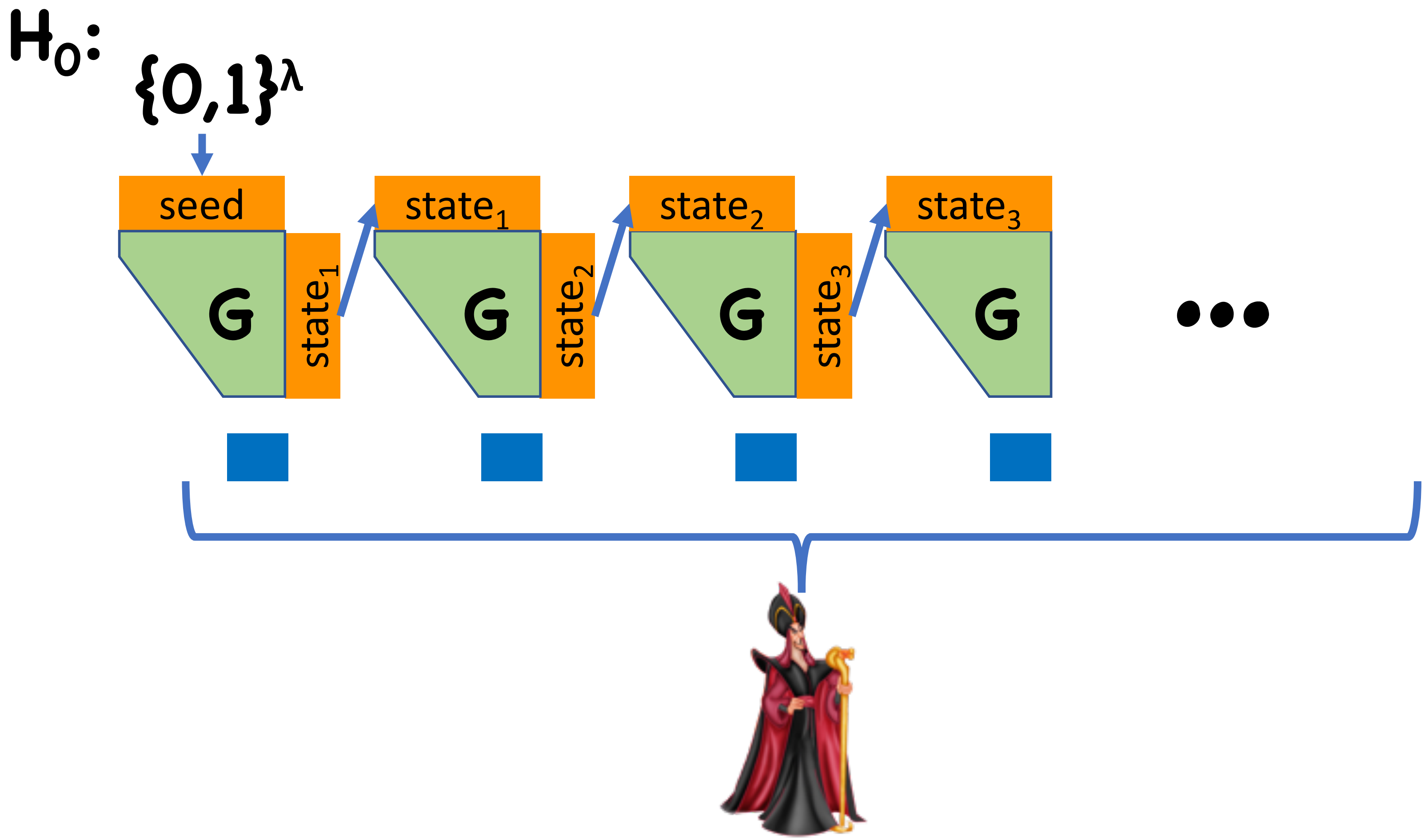


Security Proof

Assume towards contradiction  that breaks big PRG

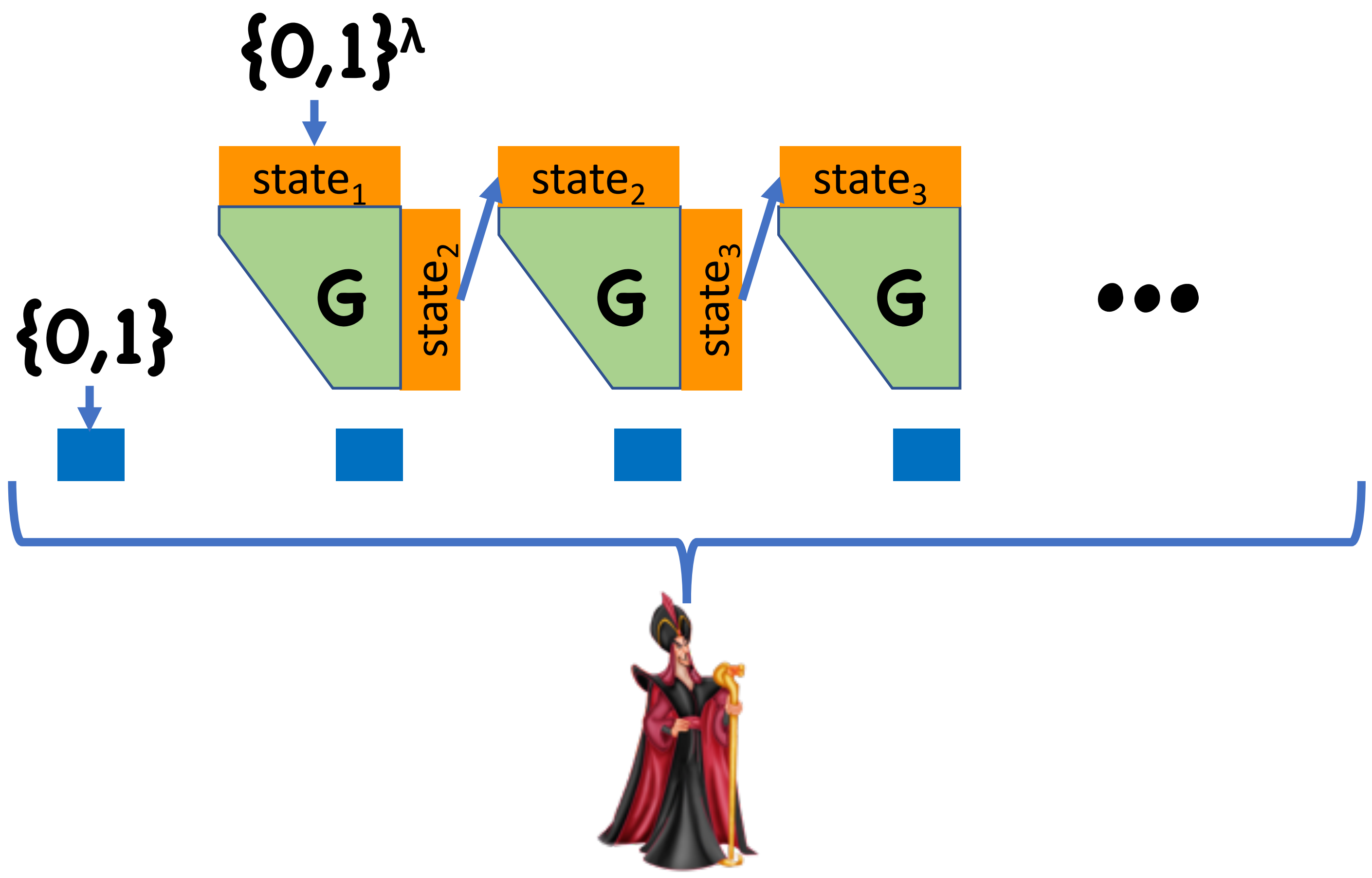
Goal: build adversary  that breaks **G**

Proof by Hybrids



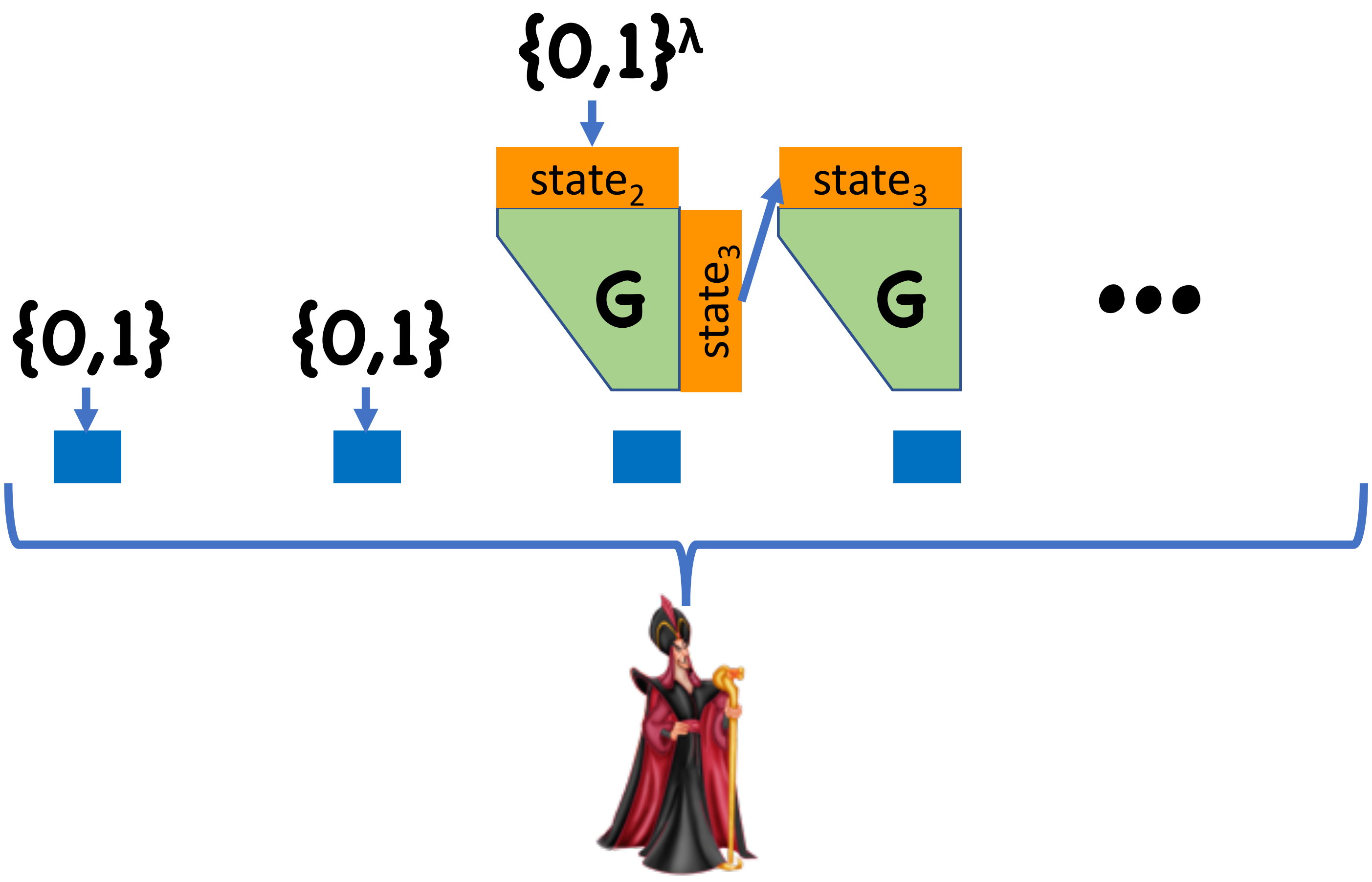
Security Proof

H_1:



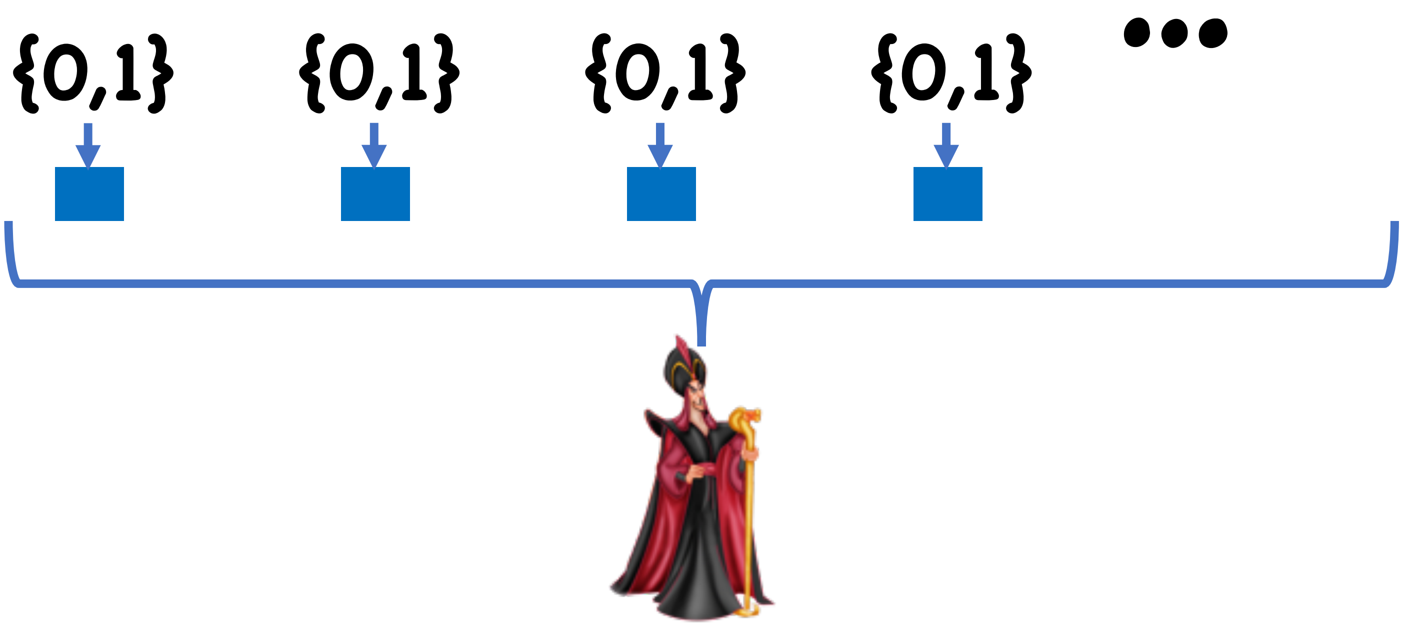
Security Proof

H_2:



Security Proof

H_t:



Security Proof

$\mathbf{H_0}$ corresponds to pseudorandom $\mathbf{x}$
$\mathbf{H_t}$ corresponds to truly random $\mathbf{x}$

Let $\mathbf{q_i} = \mathbf{Pr[}$  $\mathbf{(x)=1 : x \leftarrow H_i\ ]}$

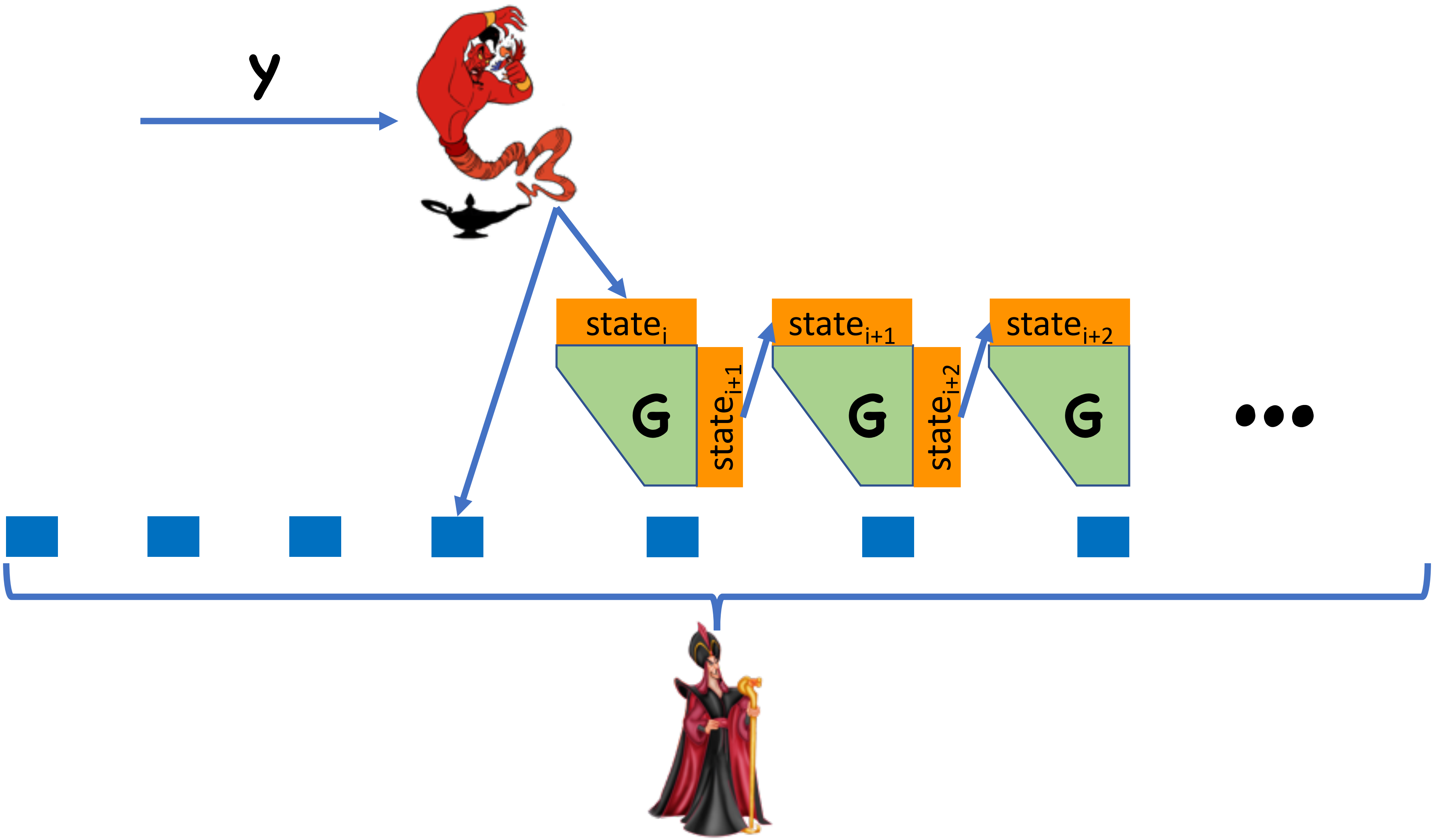
By assumption, $|\mathbf{q_t} - \mathbf{q_0}| > \varepsilon$

Triangle ineq:

$$|q_t - q_0| \leq |q_1 - q_0| + |q_2 - q_1| + \ldots + |q_t - q_{t-1}|$$

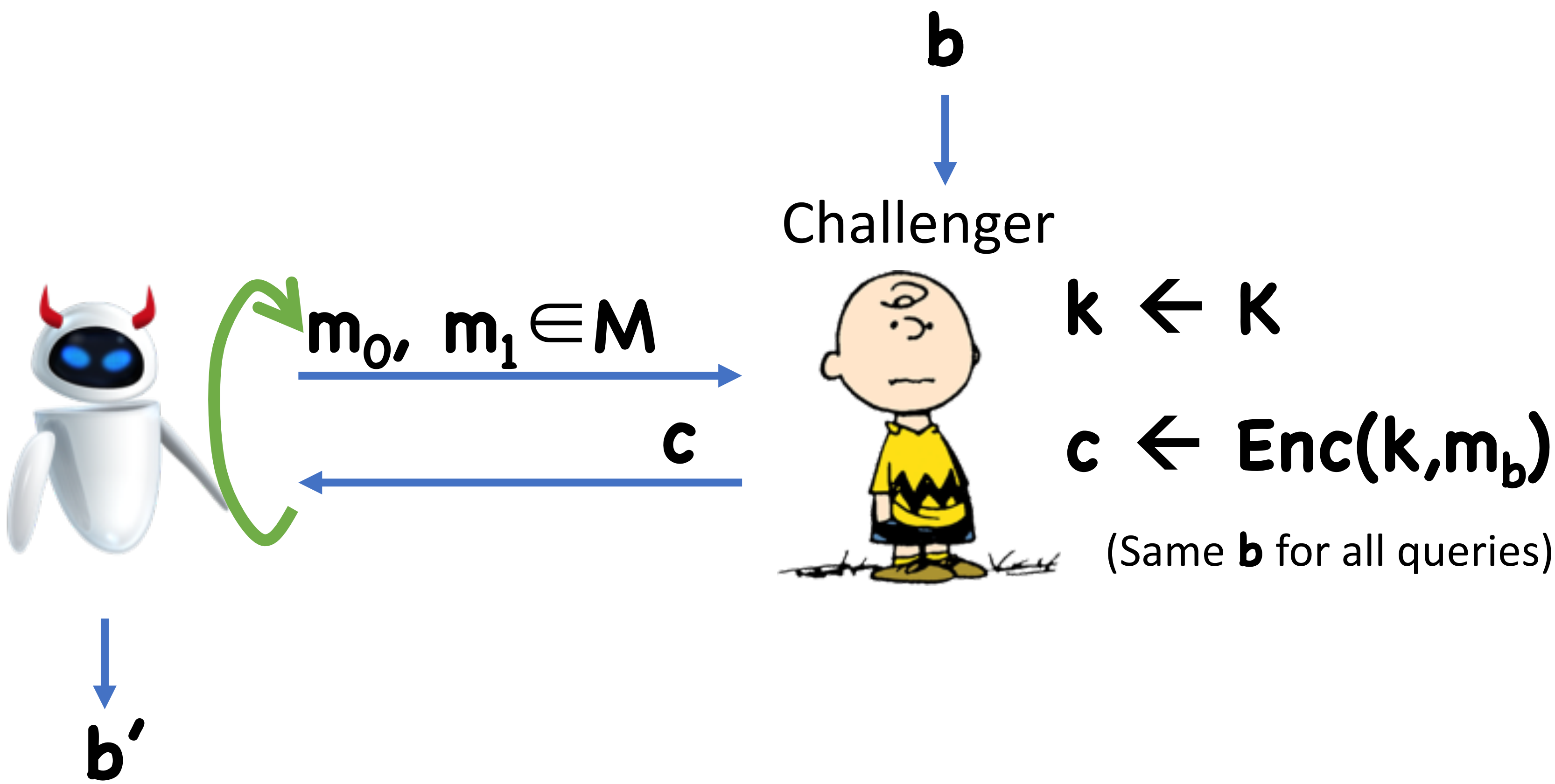
$$\Rightarrow\ \exists \mathbf{i} \text{ s.t. } |q_i - q_{i-1}| > \varepsilon/t$$

Security Proof



Today: Multiple Message Security

Left-or-Right Experiment



$\text{LoR-Exp}_b($  $, \lambda)$

LoR Security Definition

Definition: (Enc, Dec) has **Left-or-Right indistinguishability** if, for all  running in polynomial time, ∃ negligible **ε** such that:

$\Big| \Pr[1 \leftarrow \text{LoR-Exp}_0($  $, \lambda)]$

$- \Pr[1 \leftarrow \text{LoR-Exp}_1($  $, \lambda)] \Big| \leq \varepsilon(\lambda)$

Alternate Notion: CPA Security

What if adversary can additionally learn encryptions of messages of her choice?

Examples:

- Midway Island, WWII:
  - US cryptographers discover Japan is planning attack on a location referred to as “AF”
  - Guess that “AF” meant Midway Island
  - To confirm suspicion, sent message in clear that Midway Island was low on supplies
  - Japan intercepted, and sent message referencing “AF”

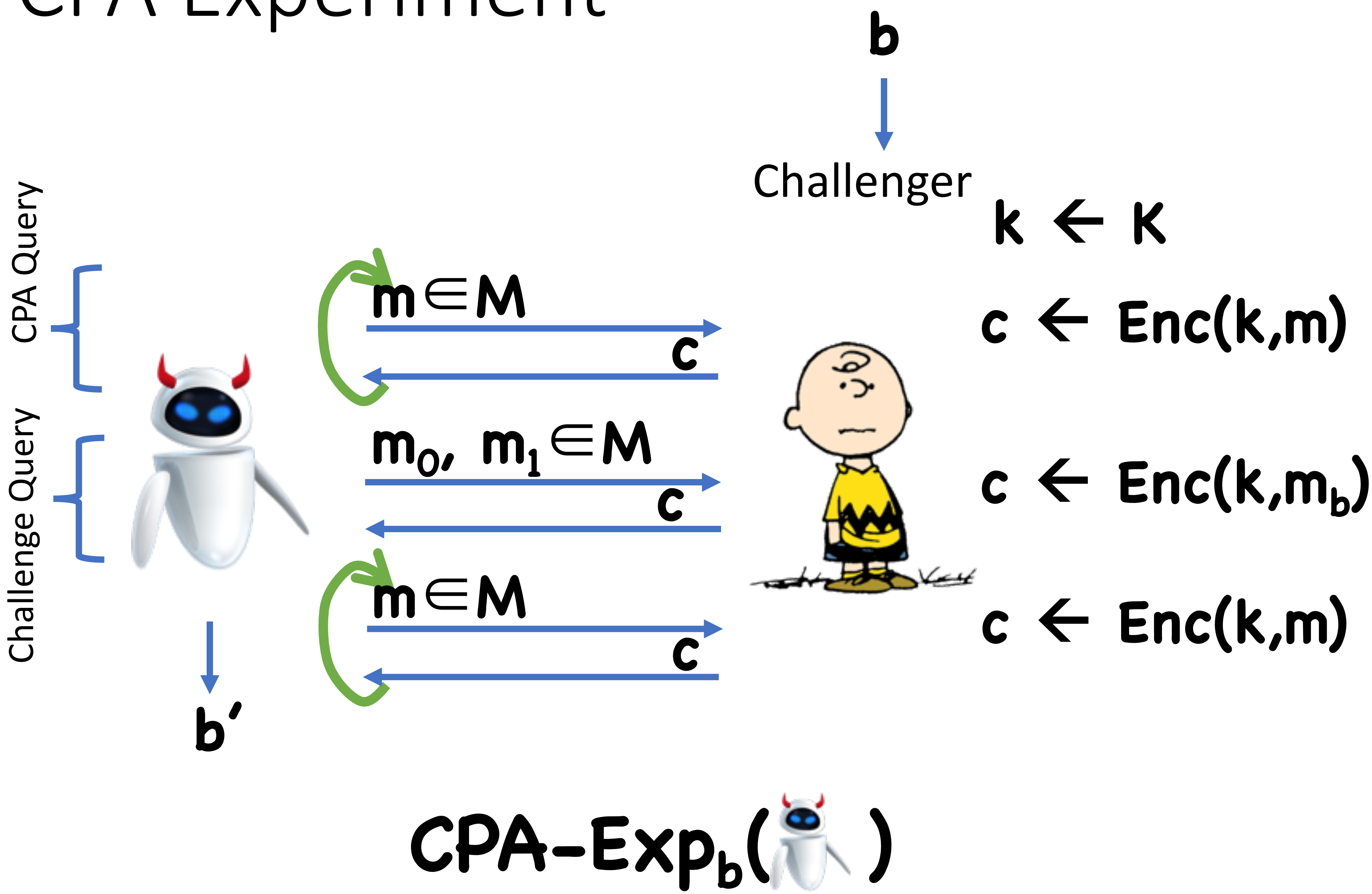
Alternate Notion: CPA Security

What if adversary can additionally learn encryptions of messages of her choice?

Examples:

- Mines, WWII:
  - Allies would lay mines at specific locations
  - Wait for Germans to discover mine
  - Germans would broadcast warning message about the mines, encrypted with Enigma
  - Would also send an “all clear” message once cleared

CPA Experiment



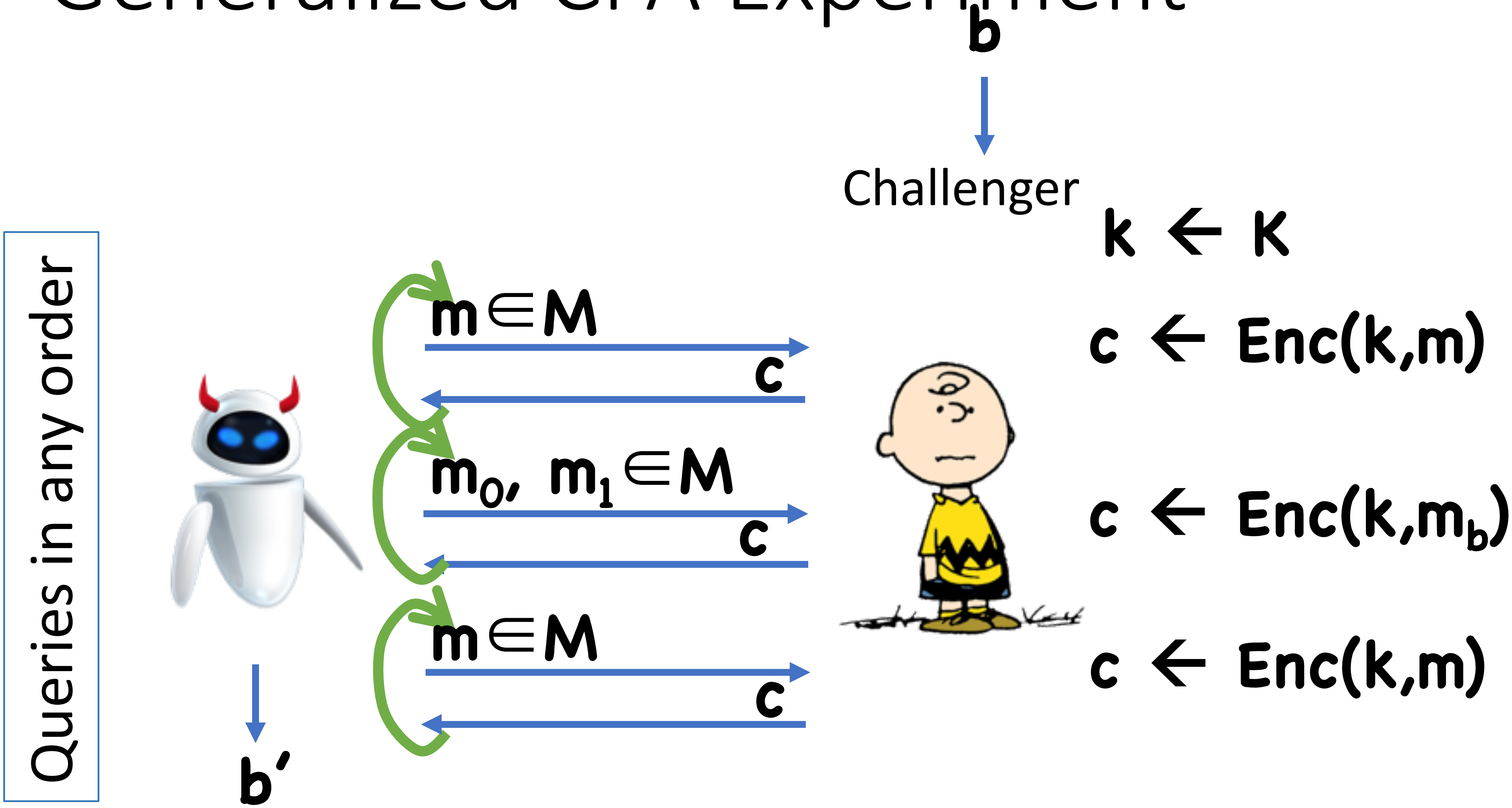
CPA Security Definition

Definition: (Enc, Dec) is **CPA Secure** if, for all  running in polynomial time, ∃ negligible **ε** such that:

| Pr[1←CPA-Exp$_0$( **, λ)]**

− Pr[1←CPA-Exp$_1$( **, λ)] | ≤ ε(λ)**

Generalized CPA Experiment



GCPA-Exp$_b$(, λ)

GCPA Security Definition

Definition: (Enc, Dec) is **Generalized CPA Secure** if, for all  running in polynomial time, ∃ negligible **ε** such that:

| Pr[1←GCPA-Exp$_0$(, λ)]

– Pr[1←GCPA-Exp$_1$(, λ)] | ≤ ε(λ)

Equivalences

Theorem:

Left-or-Right indistinguishability

$\Updownarrow$

CPA-security

$\Updownarrow$

Generalized CPA-security

Proof

We will prove:

Generalized CPA-security

$\Rightarrow$ CPA-security

$\Rightarrow$ LoR indistinguishability

$\Rightarrow$ Generalized CPA-security

Proof

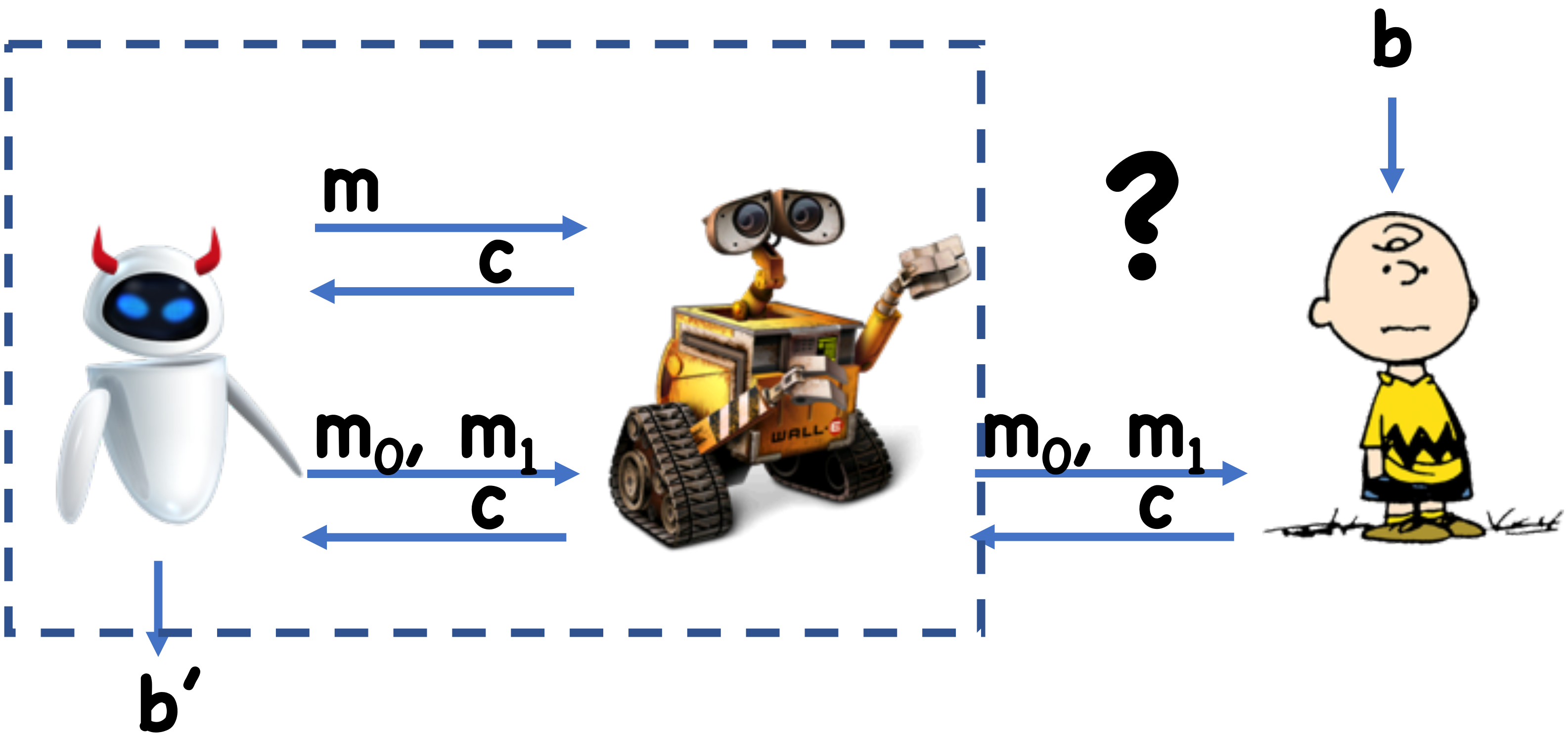
Generalized CPA-security $\Rightarrow$ CPA-security

- Trivial: any adversary in the CPA experiment is also an adversary for the generalized CPA experiment that just doesn’t take advantage of the ability to make multiple challenge/LoR queries

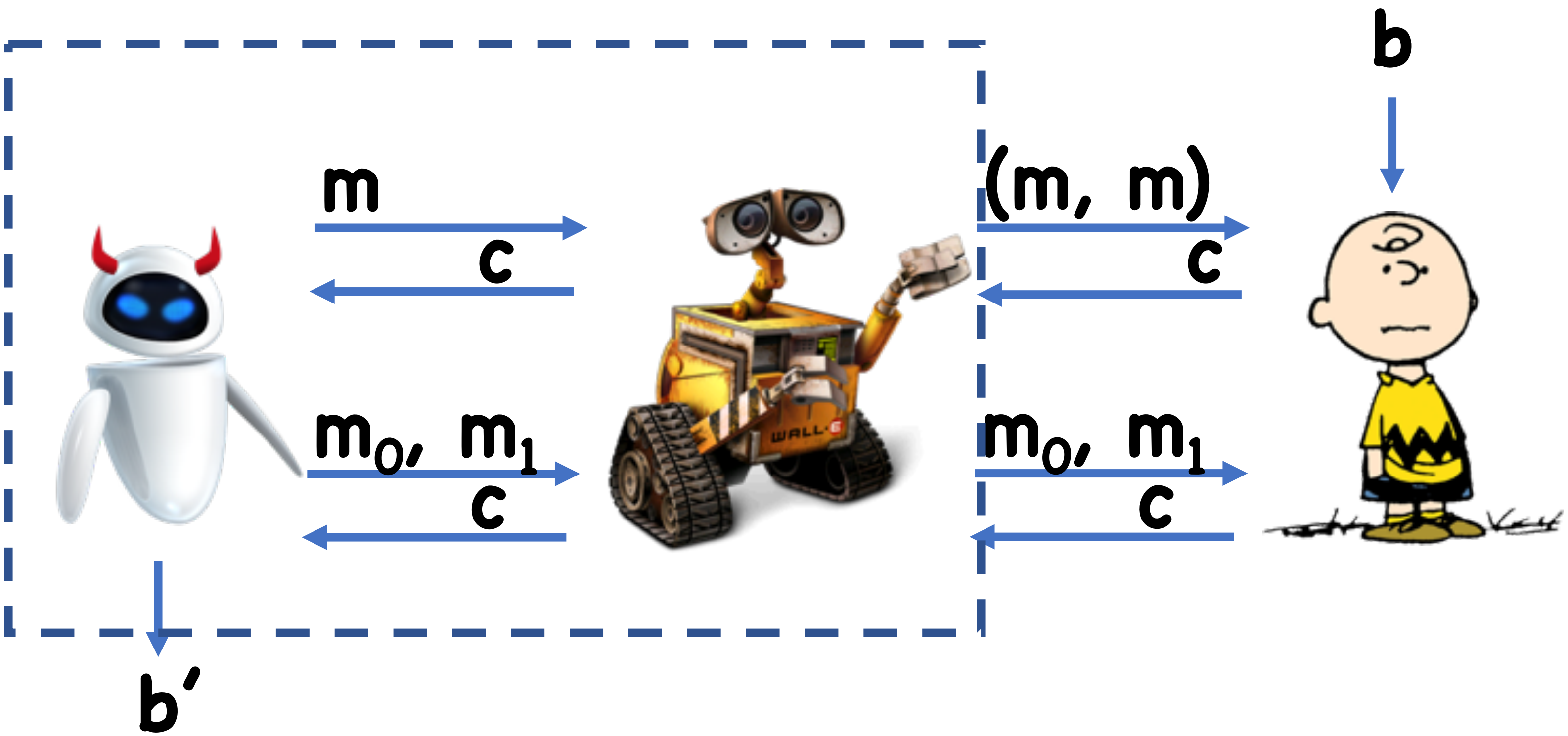
Proof

Left-or-Right $\Rightarrow$ Generalized CPA

- Assume towards contradiction that we have an adversary  for the generalized CPA experiment
- Construct an adversary  that runs  as a subroutine, and breaks the Left-or-Right indistinguishability



$\Pr[1 \leftarrow \text{LoR-Exp}_b($$, \lambda)] = \Pr[1 \leftarrow \text{GCPA-Exp}_b($$, \lambda)]$



$$\Pr[1 \leftarrow \text{LoR-Exp}_b($$$$, \lambda)] = \Pr[1 \leftarrow \text{GCPA-Exp}_b($$$$, \lambda)]$$

Proof

Left-or-Right $\Rightarrow$ Generalized CPA

$\Big|$ Pr[1←LoR-Exp$_0$( , λ)]

− Pr[1←LoR-Exp$_1$( , λ)] $\Big|$

= $\Big|$ Pr[1←GCPA-Exp$_0$( , λ)]

− Pr[1←GCPA-Exp$_1$( , λ)] $\Big|$ = ε

Proof

(regular) CPA $\Rightarrow$ Left-or-Right

- Assume towards contradiction that we have an adversary  for the **LoR Indistinguishability**
- Hybrids!

Hybrid **i**:

m_0, m_1

c

b'

$k \leftarrow K$

If at most **i** queries so far,

$c \leftarrow \text{Enc}(k, m_0)$

If more than **i** queries so far,

$c \leftarrow \text{Enc}(k, m_1)$

Proof

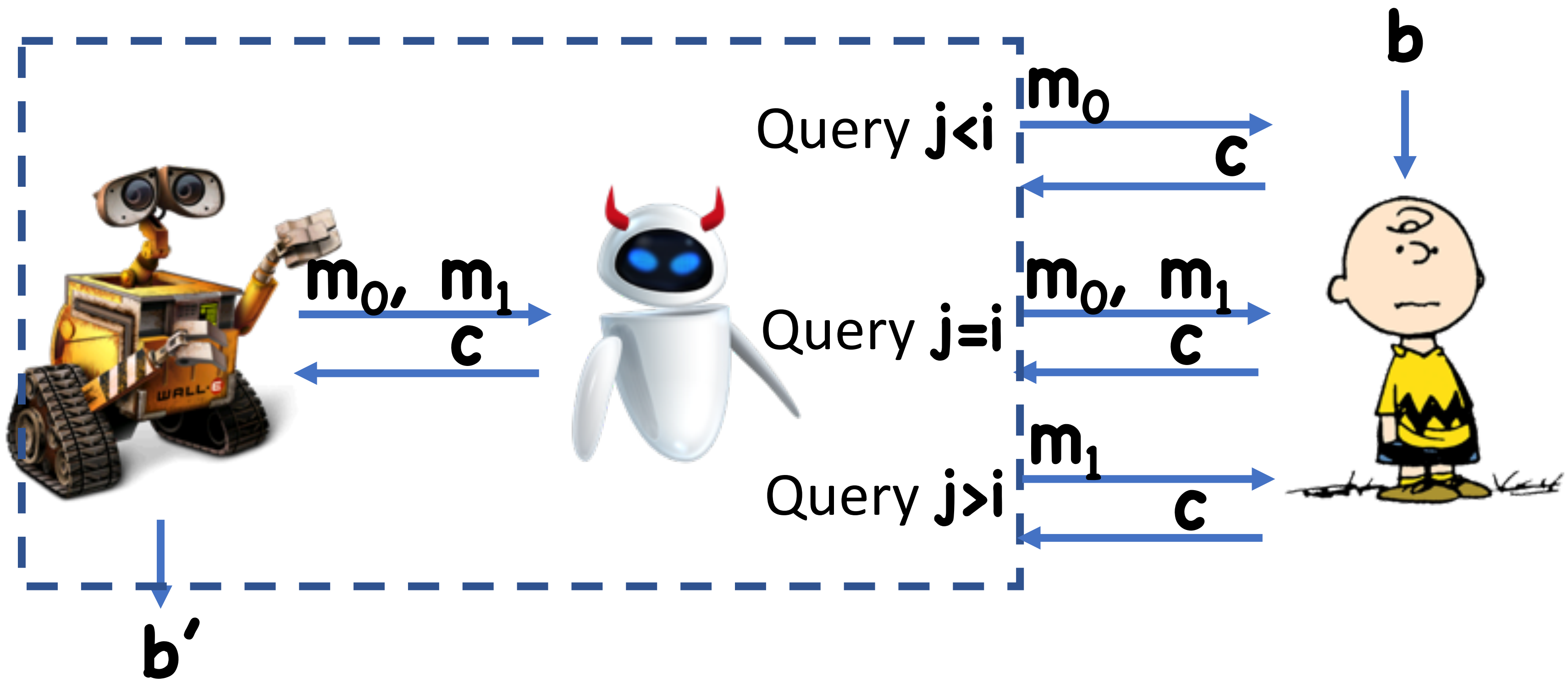
(regular) CPA $\Rightarrow$ Left-or-Right

- Hybrid **0** is identical to **LoR–Exp$_1$(****, λ)**

- Hybrid **q** is identical to **LoR–Exp$_0$(****, λ)**

- We know that  distinguishes Hybrid **q** and Hybrid **0** with advantage **ε**

  $\Rightarrow$ $\exists$ **i** s.t.  distinguishes Hybrid **i** and Hybrid **i–1** with advantage **ε/q**



Pr[1←CPA-Exp$_b$(, λ)] = Pr[1← in Hybrid **i-b**]

Proof

(regular) CPA $\Rightarrow$ Left-or-Right

| Pr[1←CPA-Exp$_0$(, λ)]

− Pr[1←CPA-Exp$_1$(, λ)] |

= | Pr[1← in Hybrid **i**]

− Pr[1← in Hybrid **i−1**] | ≥ ε/q

Equivalences

Theorem:

Left-or-Right indistinguishability

$\Updownarrow$

CPA-security

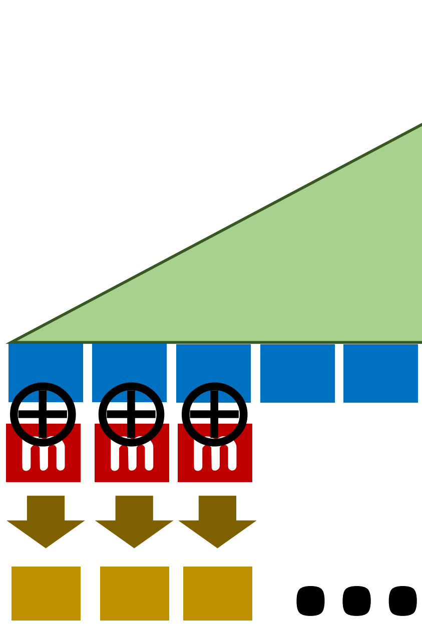
$\Updownarrow$

Generalized CPA-security

Therefore, you can use whichever notion you like best

Constructing CPA-secure Encryption

Starting point: stream ciphers = PRG + OTP for multiple messages

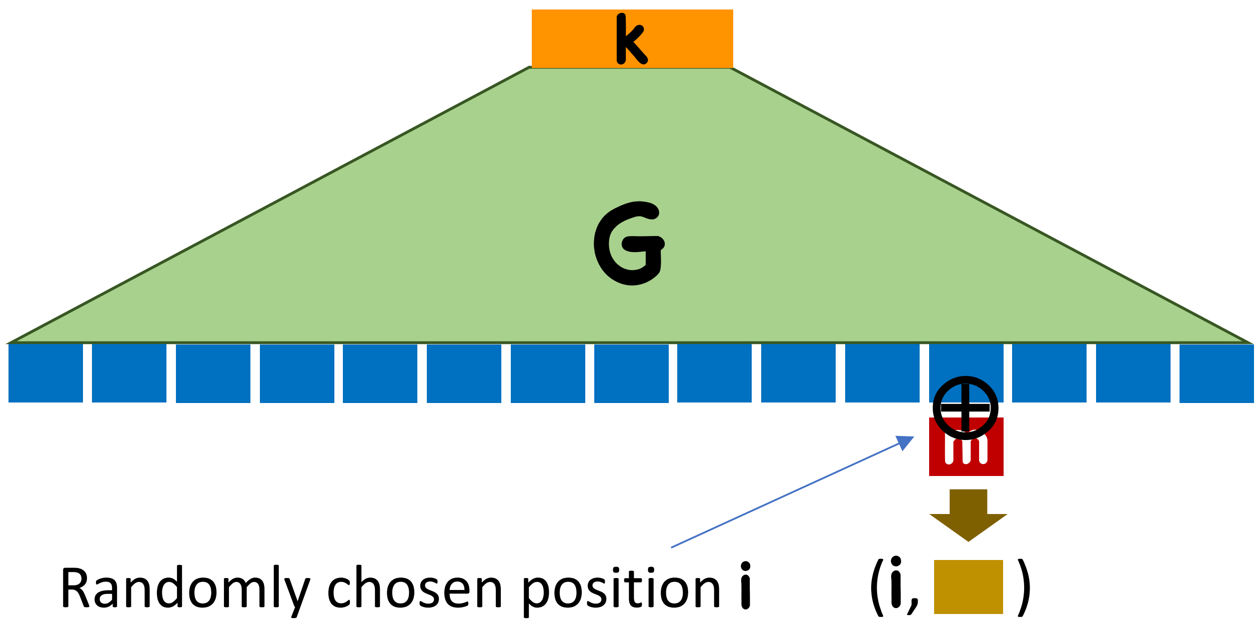


Problems:

- Stateful
- Need to synchronize with Bob

Constructing CPA-secure Encryption

Idea 1: Use random position to encrypt



Analysis

As long as the two encryptions never pick the same location, we will have security

Pr[Collision] = ?

Pr[Collision]

Consider event $E_{j,k} = (i_j = i_k)$

$\Rightarrow Pr[E_{j,k}] = 1/n$

$Pr[Collision] = Pr[E_{1,2} \text{ or } E_{1,3} \text{ or } \dots \text{ or } E_{j,k} \text{ or } \dots]$

Union bound:

$Pr[Collision] \leq \Sigma_{j,k} Pr[E_{j,k}] = \Sigma_{j,k}(1/n) = q(q-1)/2n$

Analysis

As long as the two encryptions never pick the same location, we will have security

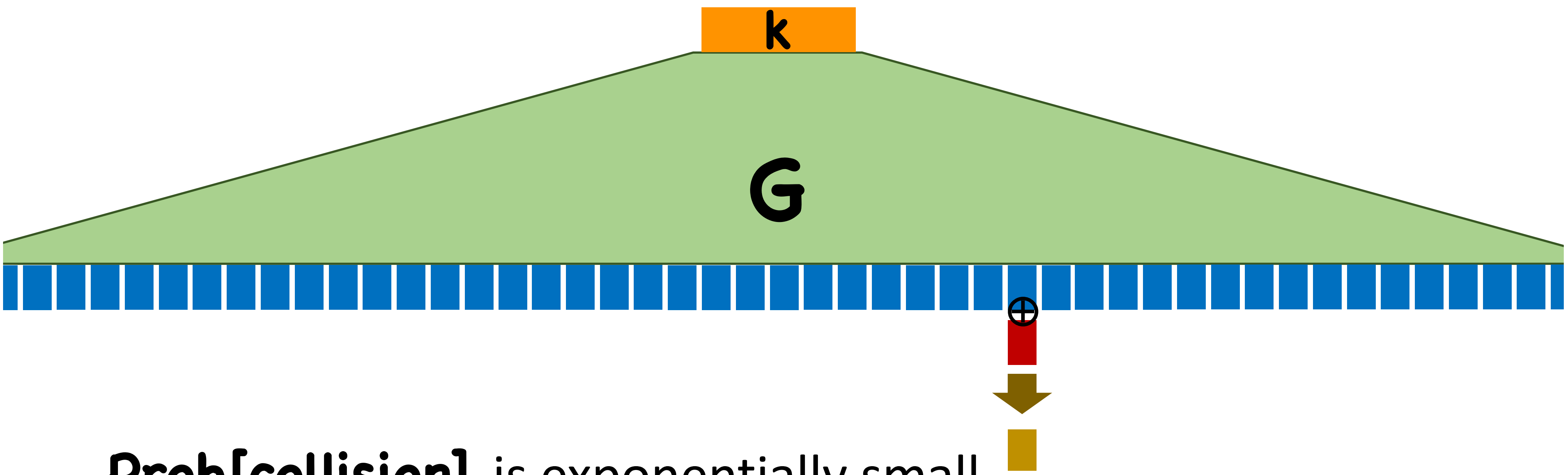
Pr[Collision] < $q^2/2n$, where
- **q =** number of messages encrypted
- **n =** number of blocks

If collision, then no security ("two-time pad")

So we get **LoR** security, with $\varepsilon' = \varepsilon + q^2/2n$

What if…

The PRG has **exponential** stretch



Prob[collision] is exponentially small
However, computing PRG takes exponential time

What if…

The PRG has **exponential** stretch

AND, it was possible to compute any 1 block of output of the PRG

- In polynomial time
- Without computing the entire output

In other words, given a key, can efficiently compute the function $F(k, x) = G(k)_x$

Pseudorandom Functions

Functions that "look like" random functions

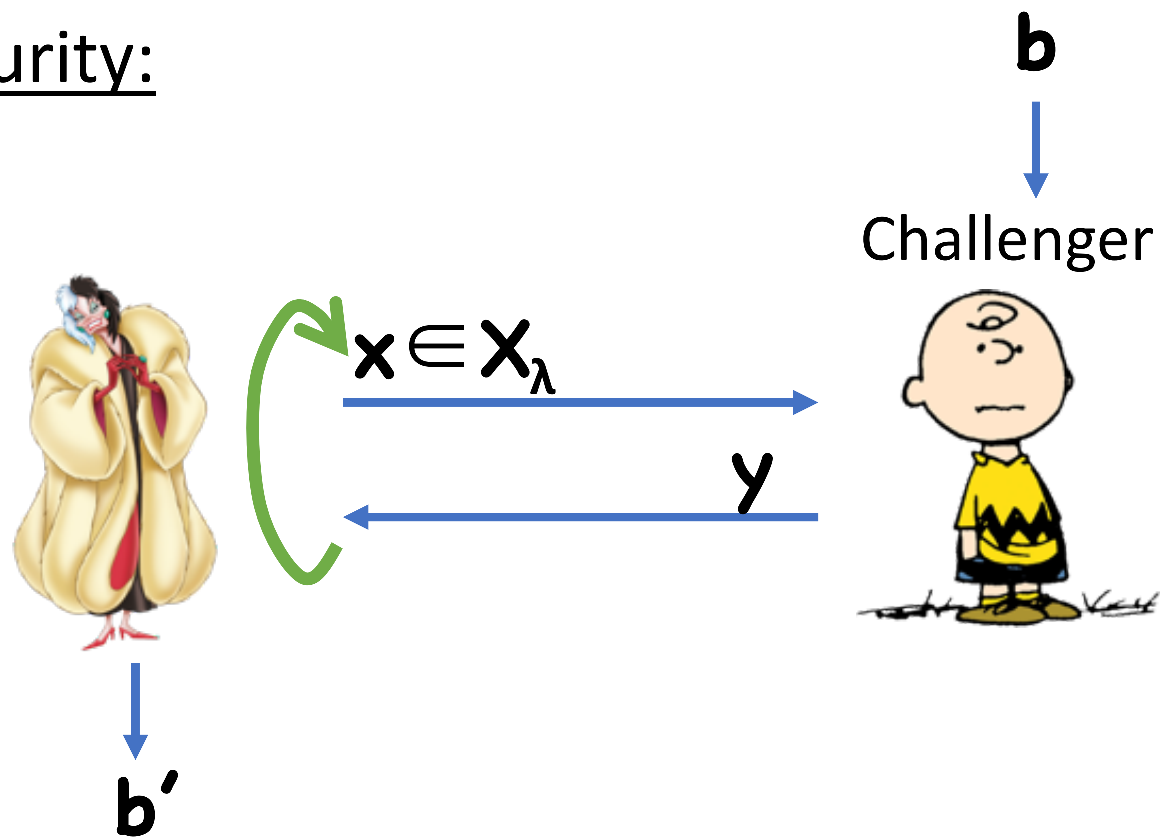
Syntax:

- Key space $\mathbf{K_\lambda}$
- Domain $\mathbf{X_\lambda}$
- Co-domain/range $\mathbf{Y_\lambda}$
- Function $\mathbf{F:K_\lambda \times X_\lambda \rightarrow Y_\lambda}$

Correctness: $\mathbf{F}$ is a function (deterministic)

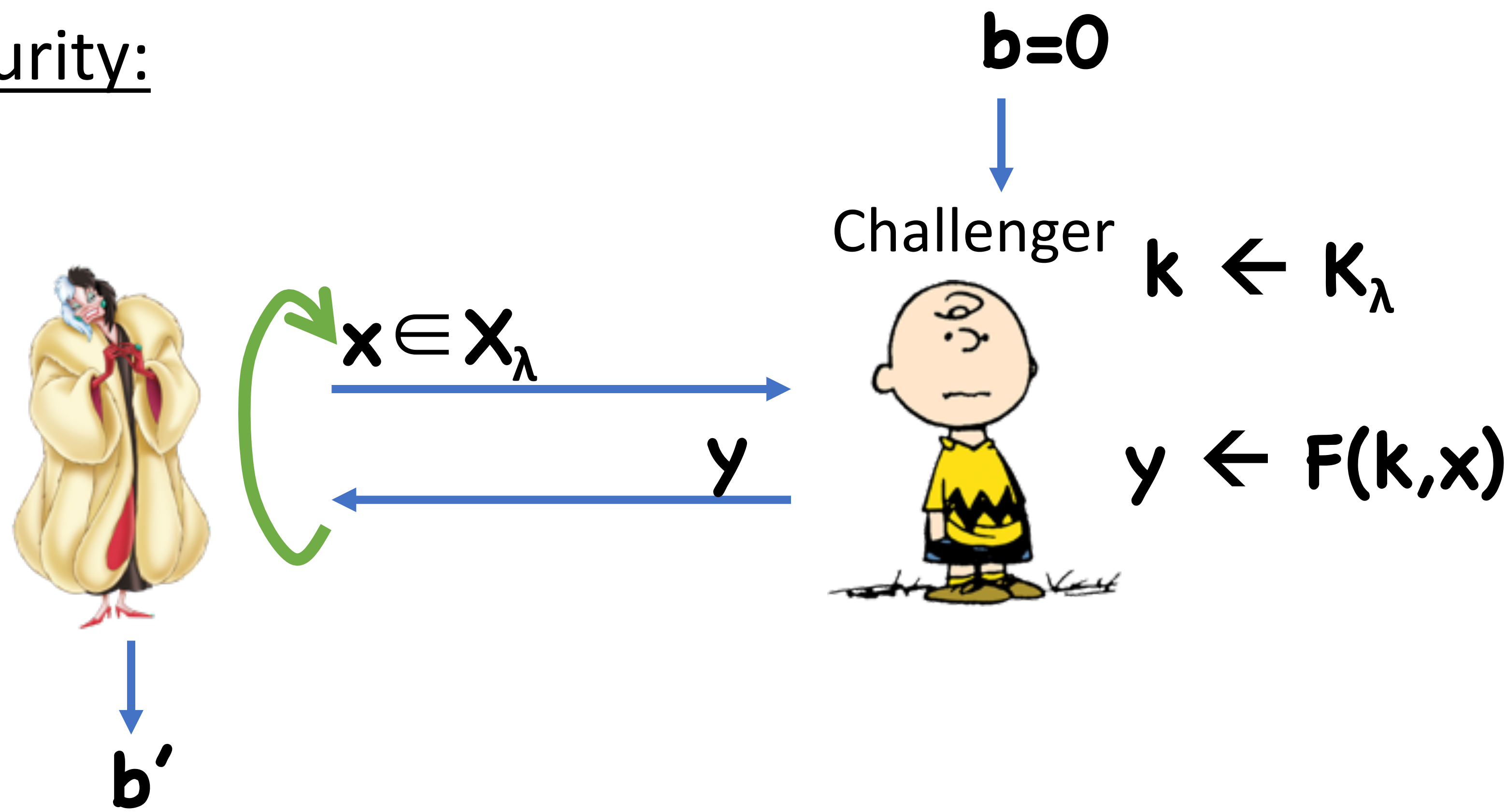
Pseudorandom Functions

Security:



Pseudorandom Functions

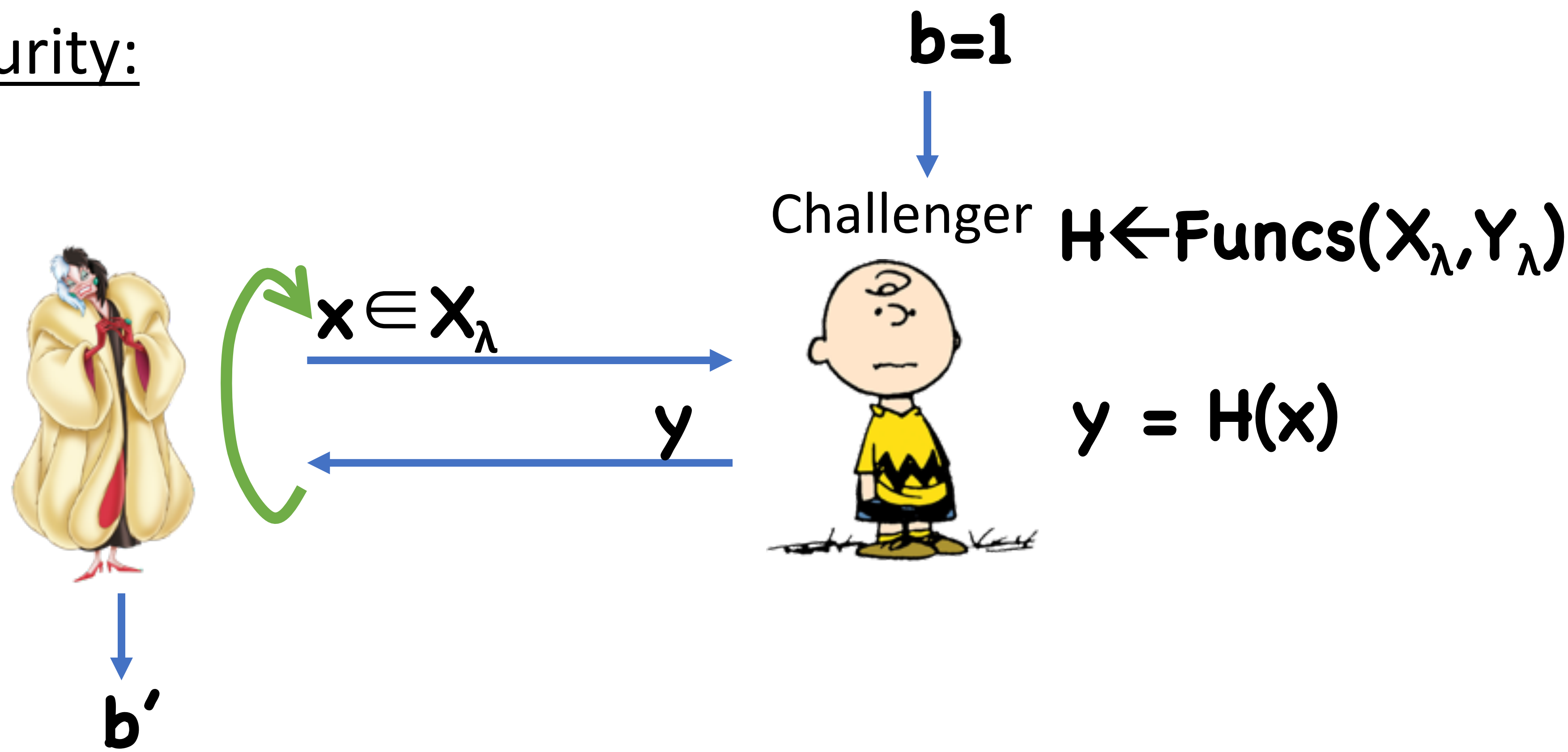
Security:



$\mathbf{PRF\text{-}Exp_0(}$  $\mathbf{, \lambda)}$

Pseudorandom Functions

Security:



$\text{PRF-Exp}_1($  $, \lambda)$

PRF Security Definition

Definition: $\mathbf{F}$ is a secure PRF if, for all  running in polynomial time, $\exists$ negligible $\boldsymbol{\varepsilon}$ such that:

$\big|\, \Pr[1 \leftarrow \text{PRF-Exp}_0($  $, \lambda)\,]$

$- \Pr[1 \leftarrow \text{PRF-Exp}_1($  $, \lambda)\,] \,\big| \leq \varepsilon(\lambda)$

Using PRFs to Build Encryption

Enc(k, m):

- Choose random $r \leftarrow X_\lambda$
- Compute $y \leftarrow F(k,r)$
- Compute $c \leftarrow y \oplus m$
- Output (r,c)

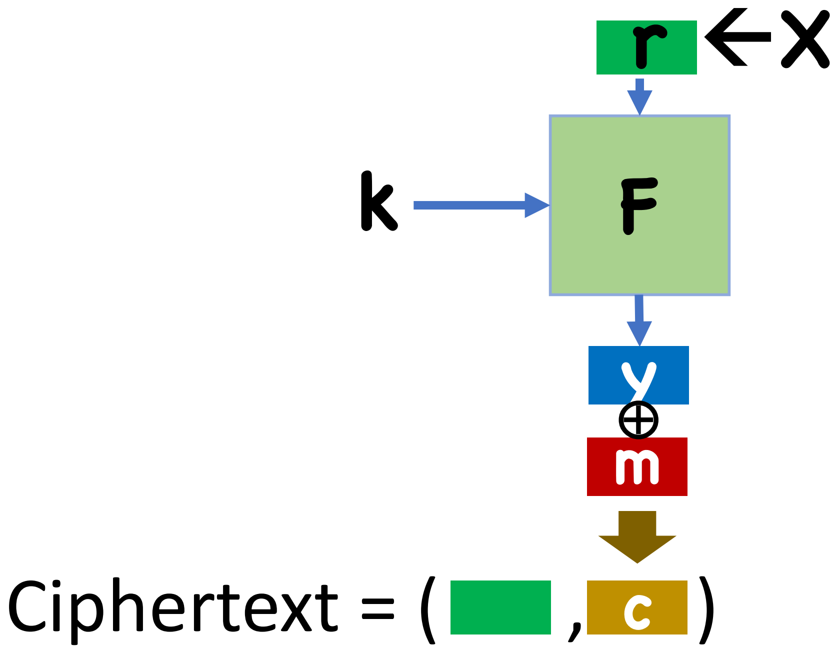
Dec(k, (r,c)):

- Compute $y' \leftarrow F(k,r)$
- Compute and output $m' \leftarrow c \oplus y'$

Correctness:

- $y'=y$ since F is deterministic
- $m' = c \oplus y = y \oplus m \oplus y = m$

Using PRFs to Build Encryption



Security

Theorem: If $\mathbf{F}$ is a secure PRF with domain $\mathbf{X_\lambda}$ and $|\mathbf{X_\lambda}|$ is superpoly, then **(Enc,Dec)** is **LoR** secure.

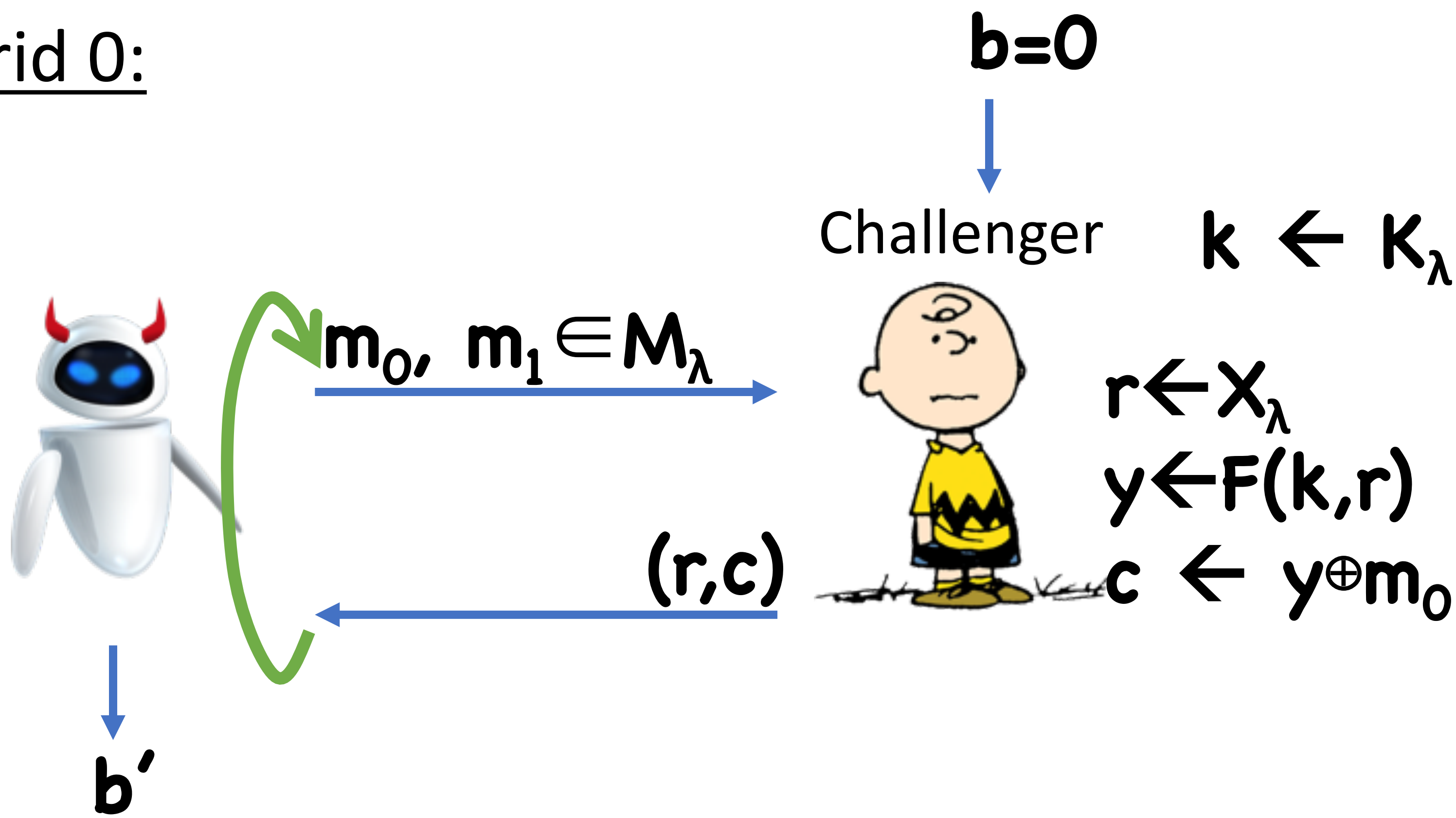
Proof

Assume toward contradiction that there exists a  breaking **(Enc,Dec)**

Hybrids…

Proof

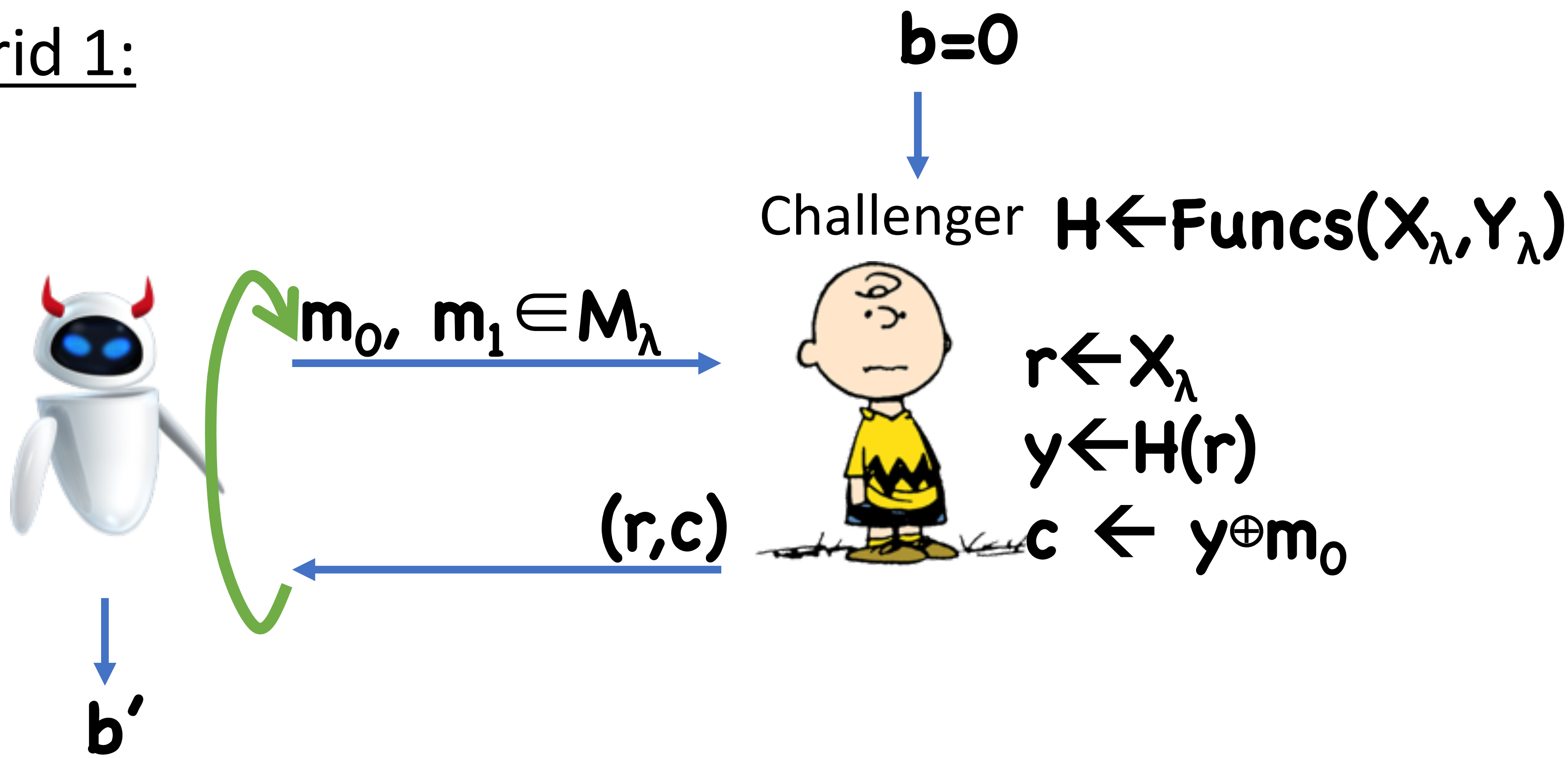
<u>Hybrid 0:</u>



LoR-Exp$_0$( **, λ)**

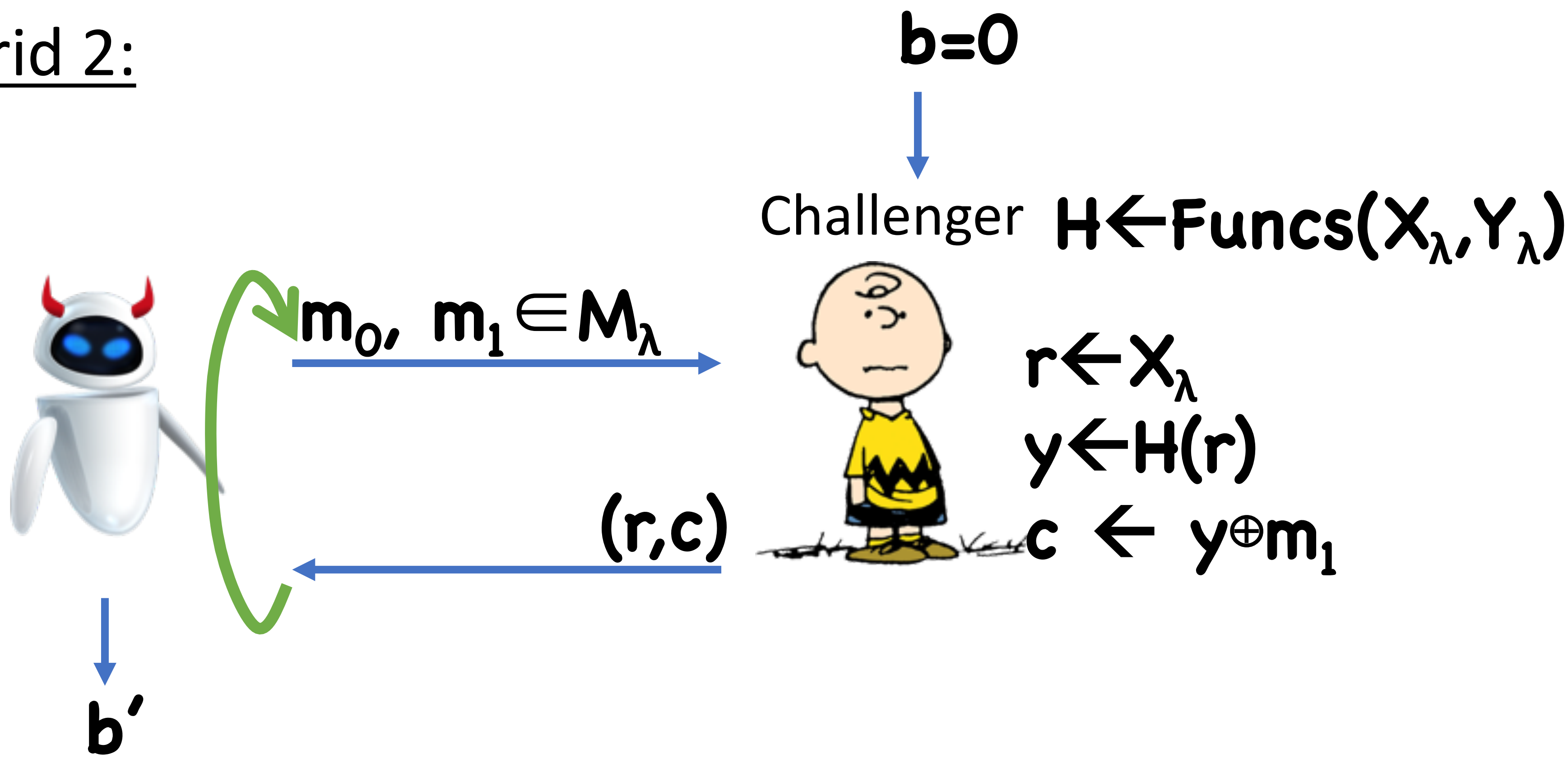
Proof

Hybrid 1:



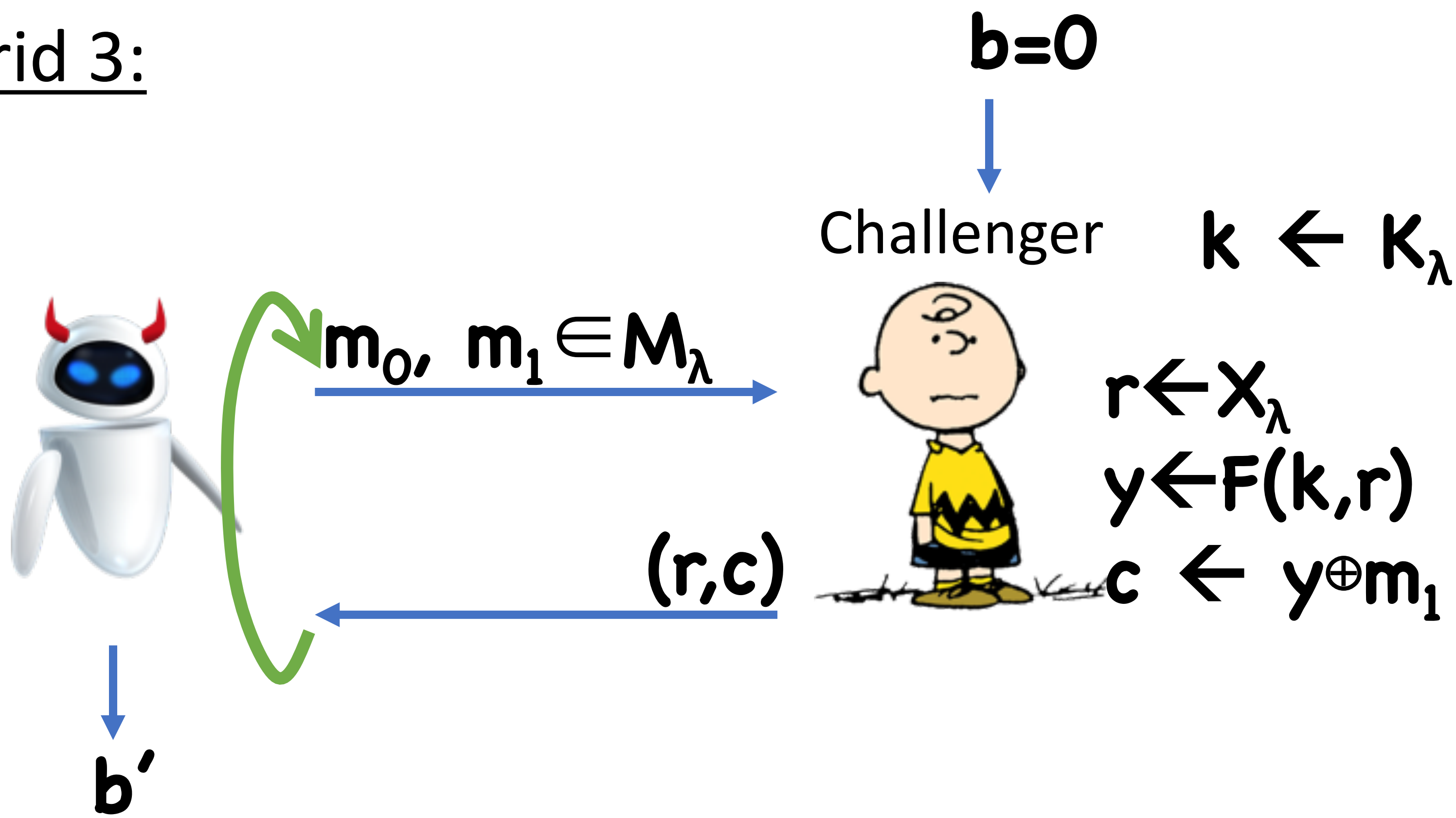
Proof

Hybrid 2:



Proof

Hybrid 3:



LoR-Exp$_1$( **, λ)**

Proof

Assume toward contradiction that there exists a  with advantage **ε** in breaking **(Enc,Dec)**

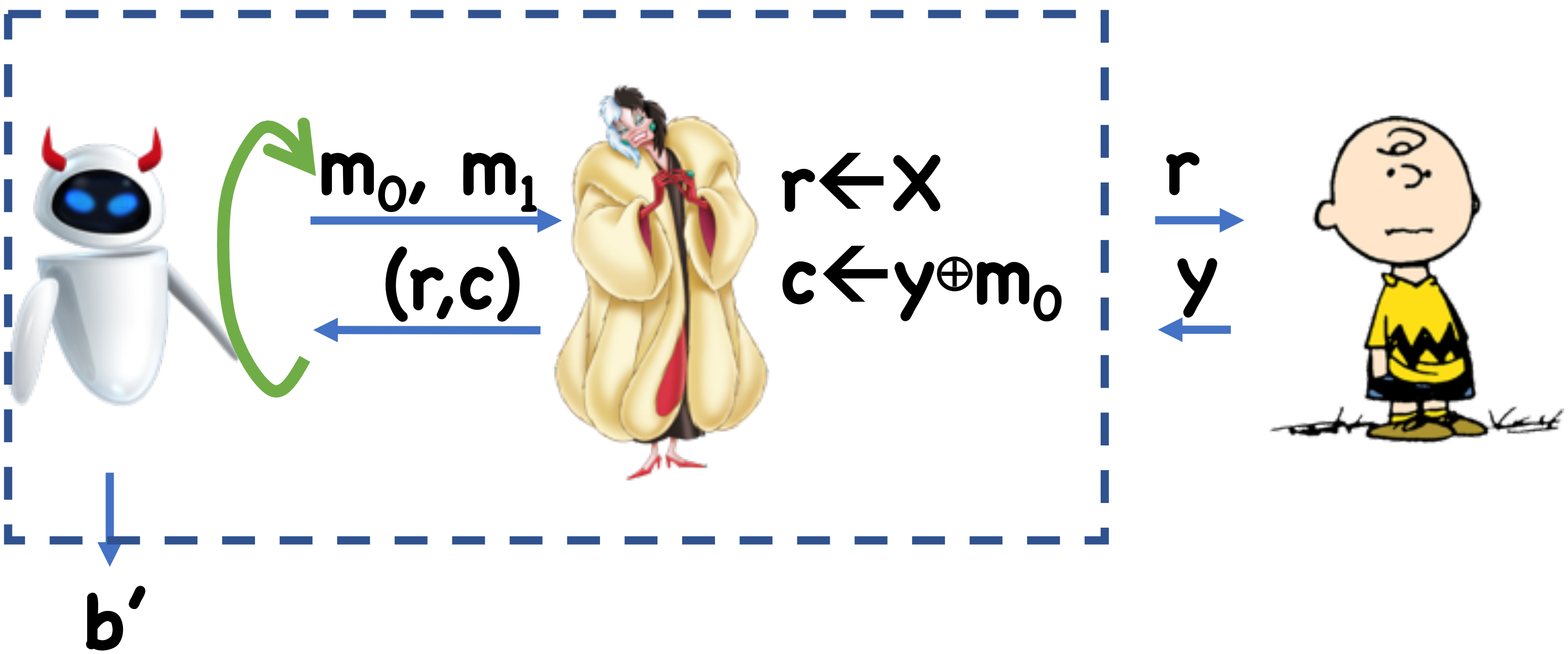
 distinguishes Hybrid 0 from Hybrid 3 with advantage **ε**, so either 

- Dist. Hybrid 0 from Hybrid 1 with adv. $\varepsilon - q^2/4|X|$
- Dist. Hybrid 1 from Hybrid 2 with adv. $q^2/2|X|$
- Dist. Hybrid 2 from Hybrid 3 with adv. $\varepsilon - q^2/4|X|$

Proof

Suppose  distinguishes Hybrid 0 from Hybrid 1

Construct 



Proof

Suppose  distinguishes Hybrid 0 from Hybrid 1

Construct 

- **PRF-Exp$_0$(****, λ)** corresponds to Hybrid 0
- **PRF-Exp$_1$(****, λ)** corresponds to Hybrid 1

Therefore,  has advantage **$\varepsilon - q^2/4|X|$**

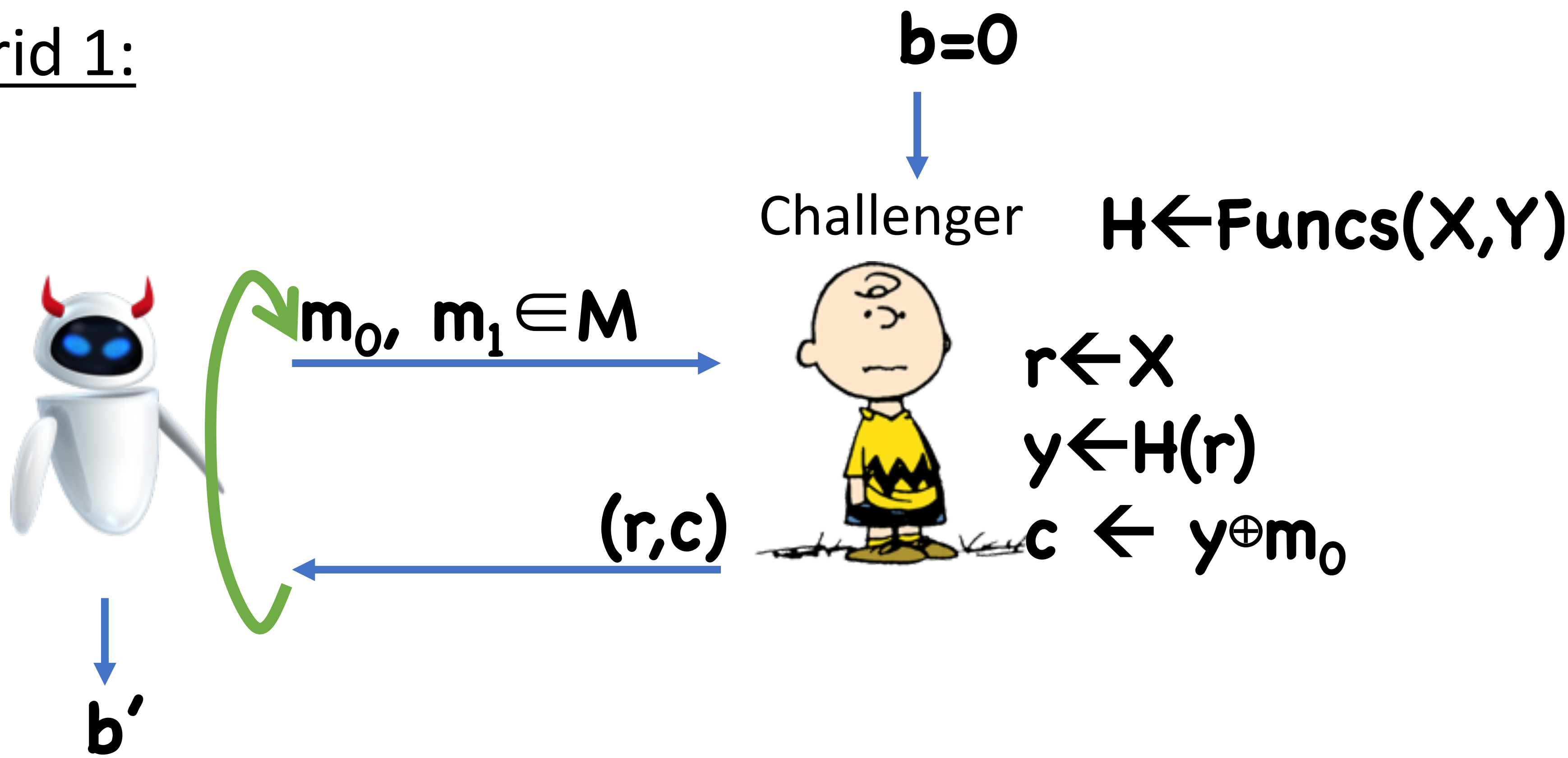
$\Rightarrow$ contradiction

Proof

Suppose  distinguishes Hybrid 1 from Hybrid 2

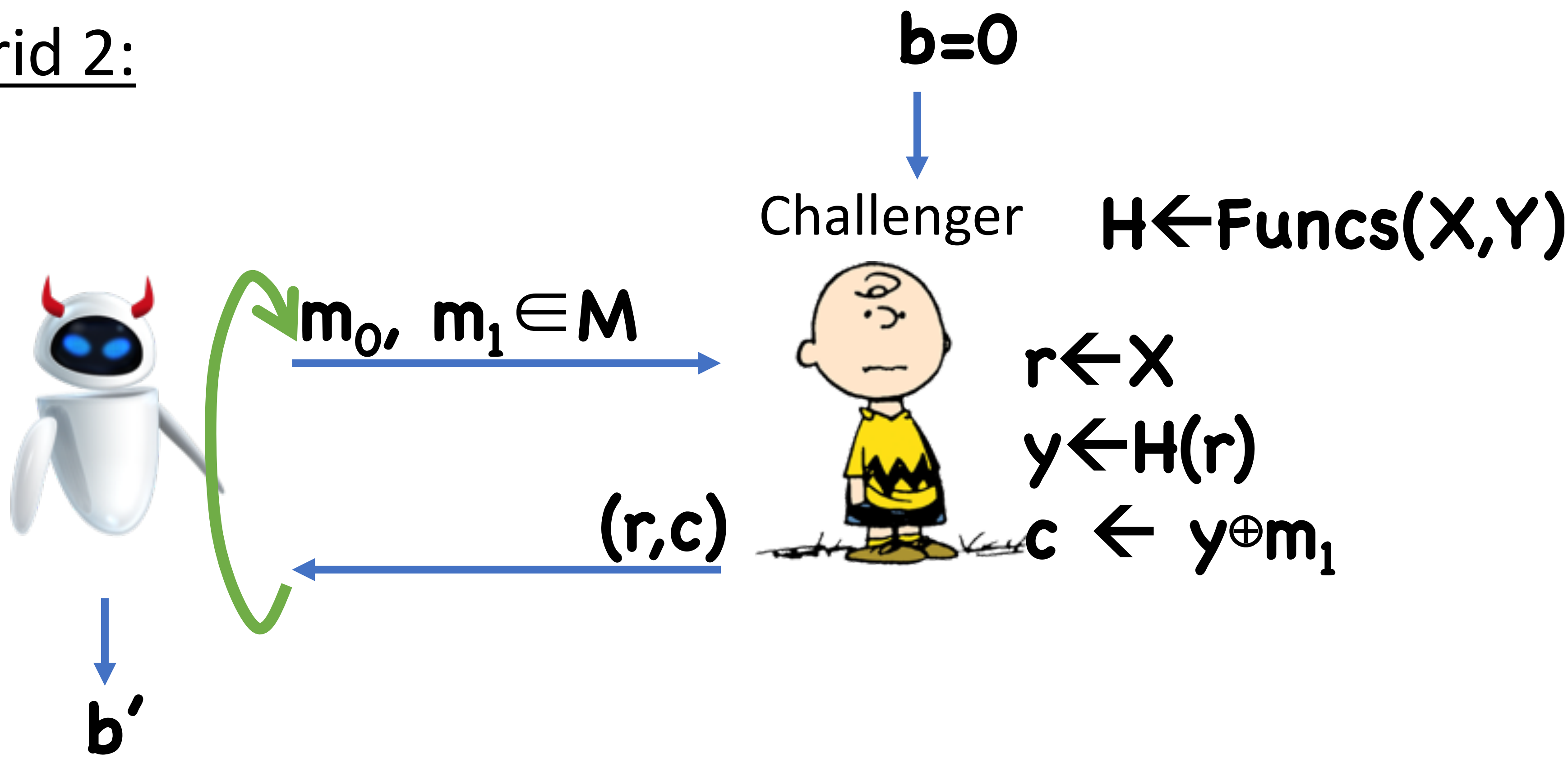
Proof

Hybrid 1:



Proof

Hybrid 2:



Proof

Suppose  distinguishes Hybrid 1 from Hybrid 2

As long as the **r**'s for every query are distinct, the **y**'s for each query will look like truly random strings

In this case, encrypting $\mathbf{m_0}$ vs $\mathbf{m_1}$ will be perfectly indistinguishable

- By OTP security

Proof

Suppose  distinguishes Hybrid 1 from Hybrid 2

Therefore, advantage is **≤Pr[**collision in the **r**'s**]**

$< q^2/2|X|$

Proof

Suppose  distinguishes Hybrid 2 from Hybrid 3

Almost identical to the 0/1 case…

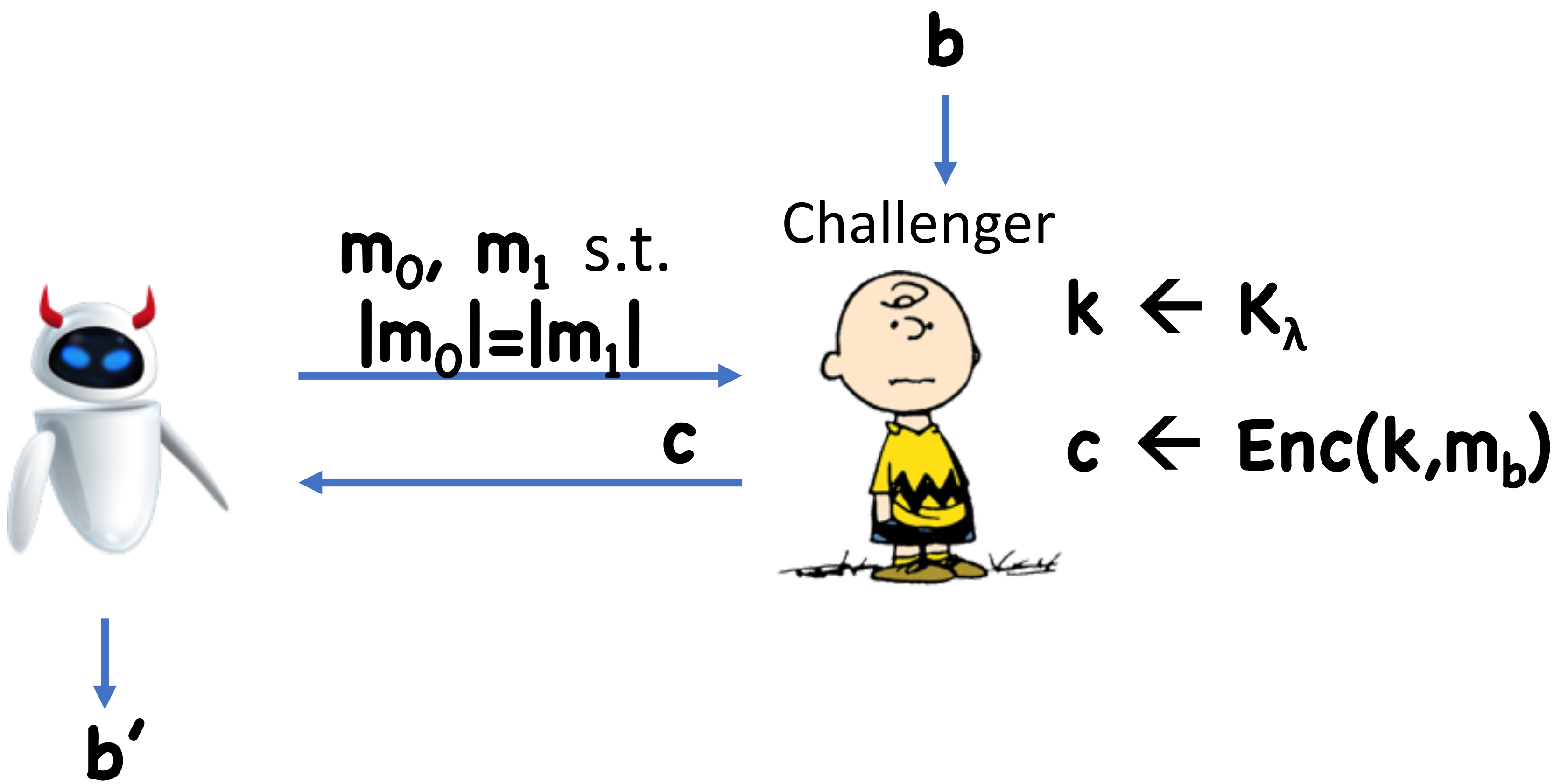
Using PRFs to Build Encryption

So far, scheme had fixed-length messages

- Namely, $\mathbf{M}_\lambda = \mathbf{Y}_\lambda$

Now suppose we want to handle arbitrary-length messages

Security for Arbitrary-Length Messages



IND-Exp$_b$(, λ)

Theorem: Given any CPA-secure **(Enc,Dec)** for fixed-length messages (even single bit), it is possible to construct a CPA-secure **(Enc,Dec)** for arbitrary-length messages

Construction

Let **(Enc,Dec)** be CPA-secure for single-bit messages

Enc'(k,m):

For $i=1,\dots,|m|$, run $c_i \leftarrow Enc(k, m_i)$

Output $(c_1, \dots, c_{|m|})$

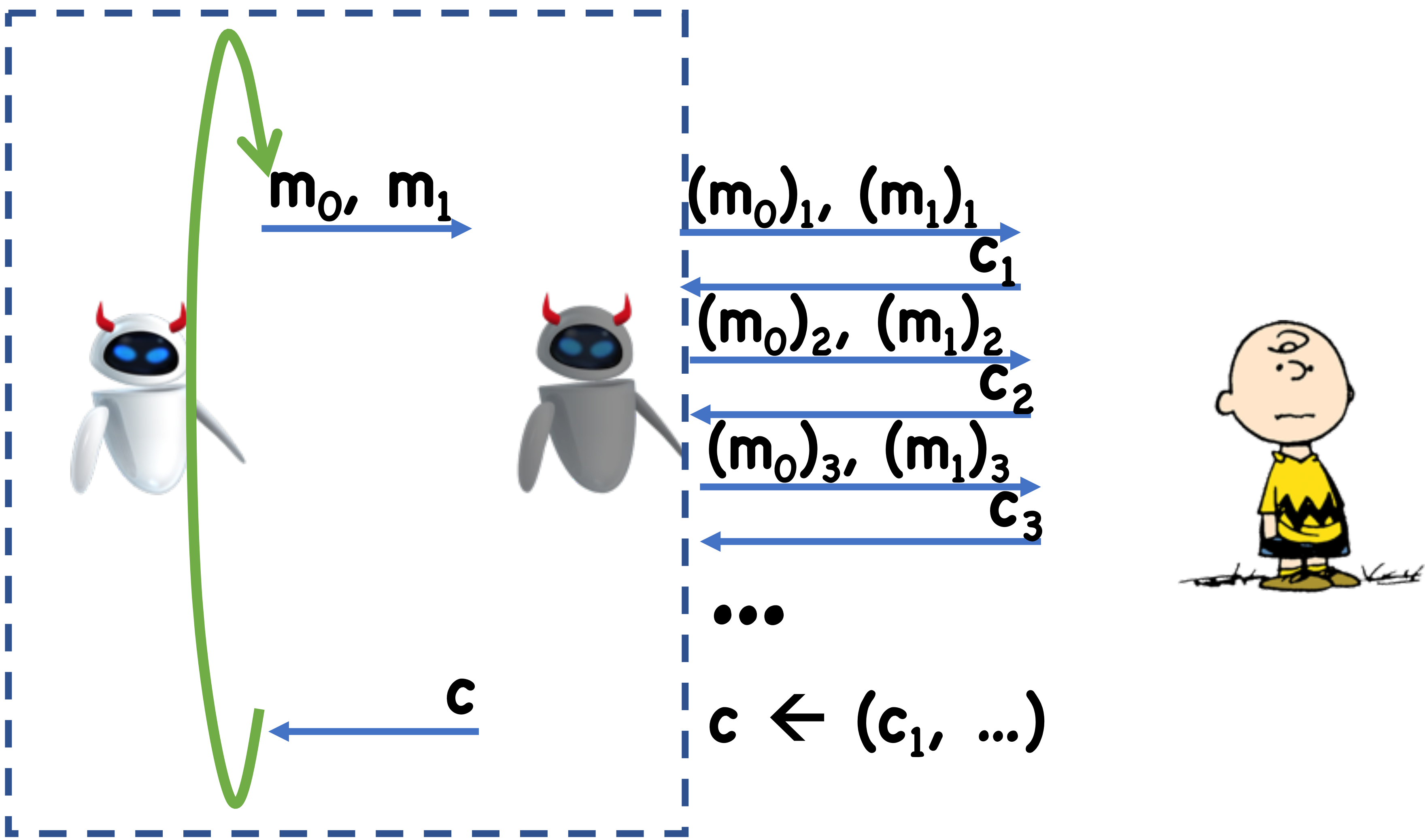
Dec'(k, (c₁, …, cₗ)):

For $i=1,\dots, l$, run $m_i \leftarrow Dec(k, c_i)$

Output $m = m_1 m_2 \dots, m_l$

Theorem: If **(Enc,Dec)** is LoR secure, then **(Enc',Dec')** is LoR secure

Proof (sketch)



Better Constructions Using PRFs

In PRF-based construction, encrypting single bit requires $\boldsymbol{\lambda+1}$ bits

$\Rightarrow$ encrypting $\boldsymbol{l}$-bit message requires $\boldsymbol{\approx\lambda l}$ bits

Ideally, ciphertexts would have size $\boldsymbol{\approx\lambda+l}$

Solution 1: Add PRG/Stream Cipher

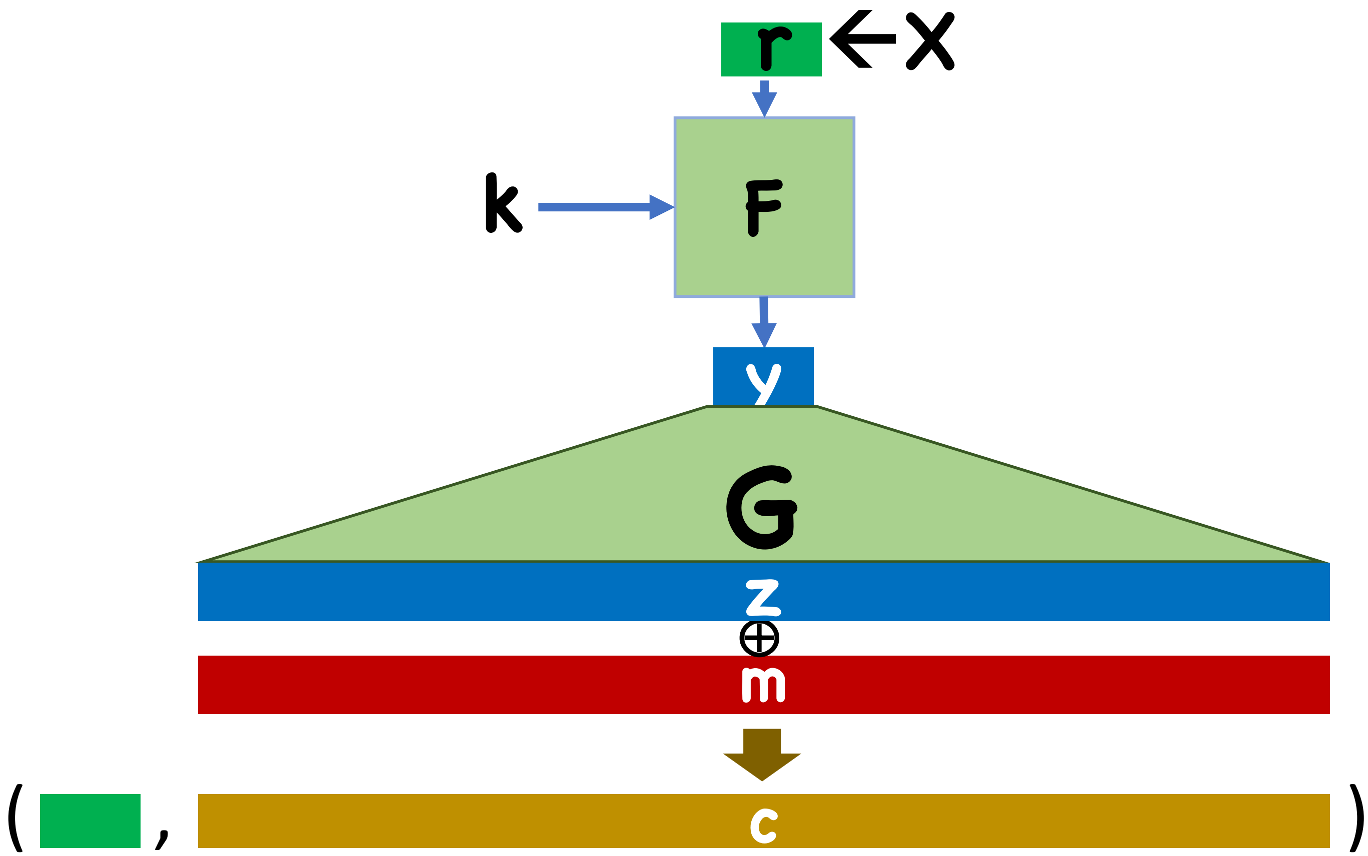
Enc(k, m):

- Choose random $r \leftarrow X$
- Compute $y \leftarrow F(k,r)$
- Get $|m|$ pseudorandom bits $z \leftarrow G(y)$
- Compute $c \leftarrow z \oplus m$
- Output (r,c)

Dec(k, (r,c)):

- Compute $y' \leftarrow F(k,r)$
- Compute $z' \leftarrow G(y')$
- Compute and output $m' \leftarrow c \oplus z'$

Solution 1: Add PRG/Stream Cipher



Solution 2: Counter Mode

Enc(k, m):

- Choose random $r \leftarrow \{0,1\}^{\lambda/2}$
- For $i=1,\dots,|m|$,
  - Compute $y_i \leftarrow F(k, r||i)$
  - Compute $c_i \leftarrow y_i \oplus m_i$
- Output (r,c) where $c=(c_1,\dots,c_{|m|})$

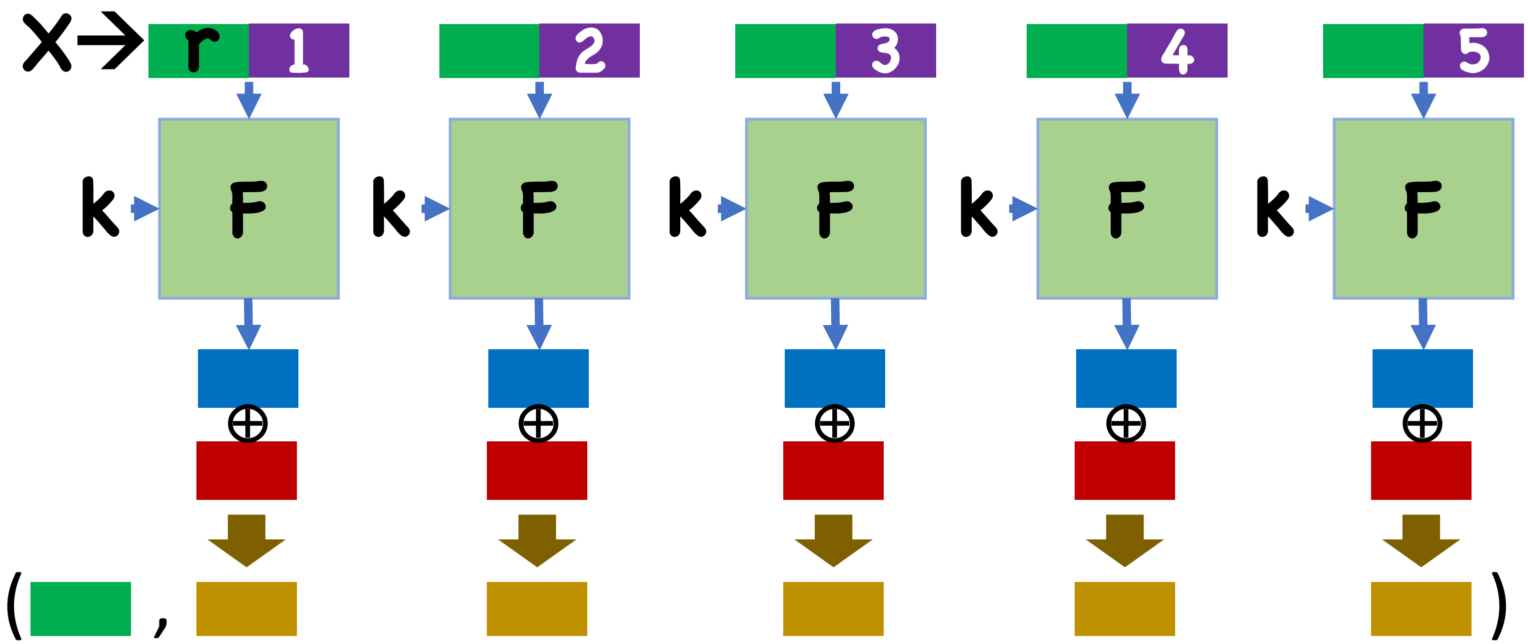
Write i as $\lambda/2$-bit string

Dec(k, (r,c)):

- For $i=1,\dots,l$,
  - Compute $y_i \leftarrow F(k, r||i)$
  - Compute $m_i \leftarrow y_i \oplus c_i$
- Output $m=m_1,\dots,m_l$

Handles any message of length at most $2^{\lambda/2}$

Solution 2: Counter Mode



Summary

PRFs = “random looking” functions

Can be used to build security for arbitrary length/number of messages with stateless scheme

Next time: block ciphers and other “modes” of operation

Reminders

HW2 Due Feb 27th
HW3 Due March 5th

PR1 Due March 10th