COS433/Math 473: Cryptography

Mark Zhandry

Princeton University

Spring 2017

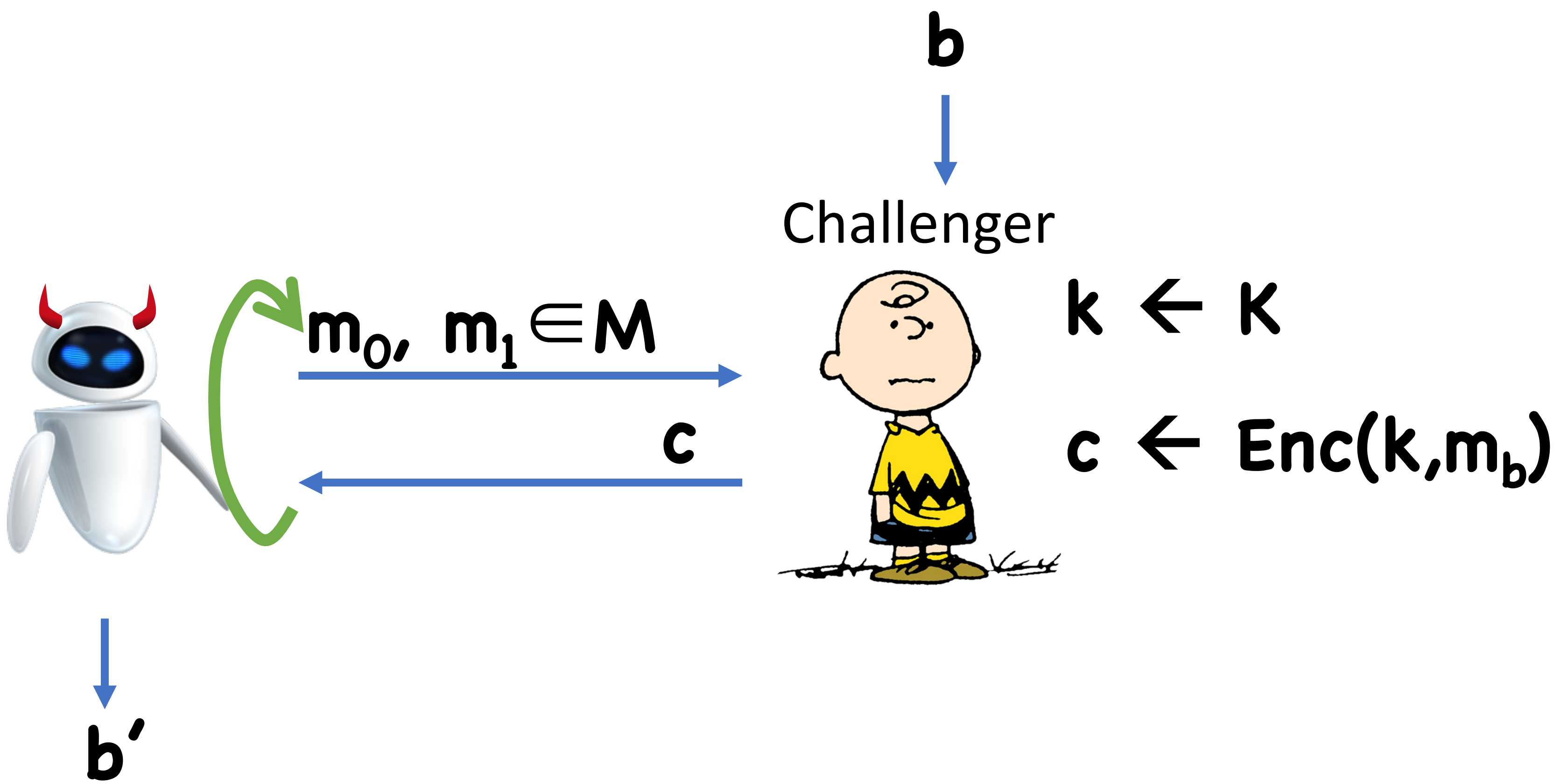
Project 1 – 2nd Bonus

Still at 40 decrypts…

- Tristan Pollner and Zachary Stier
- Prinstun Criptoe (Heather Newman, Iris Rukshin, Jacob Wachspress)

Previously on COS 433…

Left-or-Right Experiment



LoR-Exp$_b$( **)**

Pseudorandom Functions

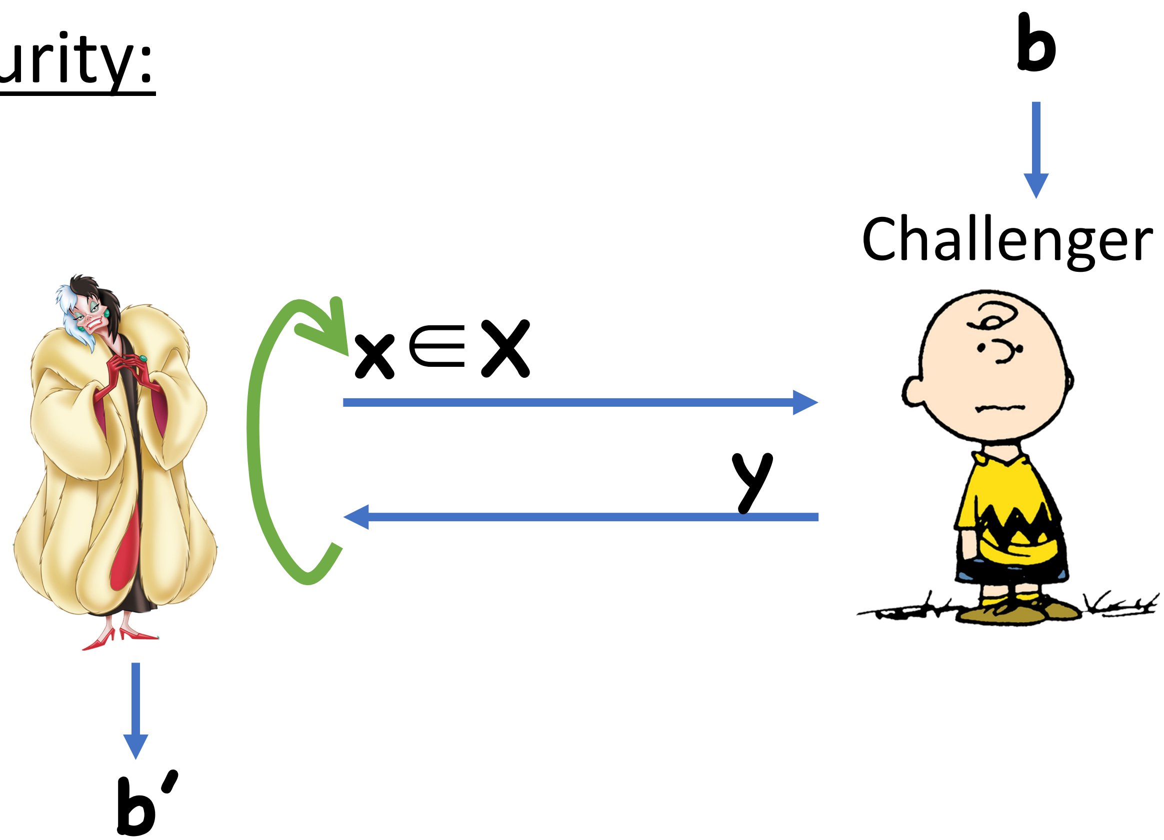
Functions that "look like" random functions

Syntax:

- Key space **K** (usually $\{0,1\}^\lambda$)
- Domain **X** (usually $\{0,1\}^m$)
- Co-domain/range **Y** (usually $\{0,1\}^n$)
- Function $F:K \times X \to Y$

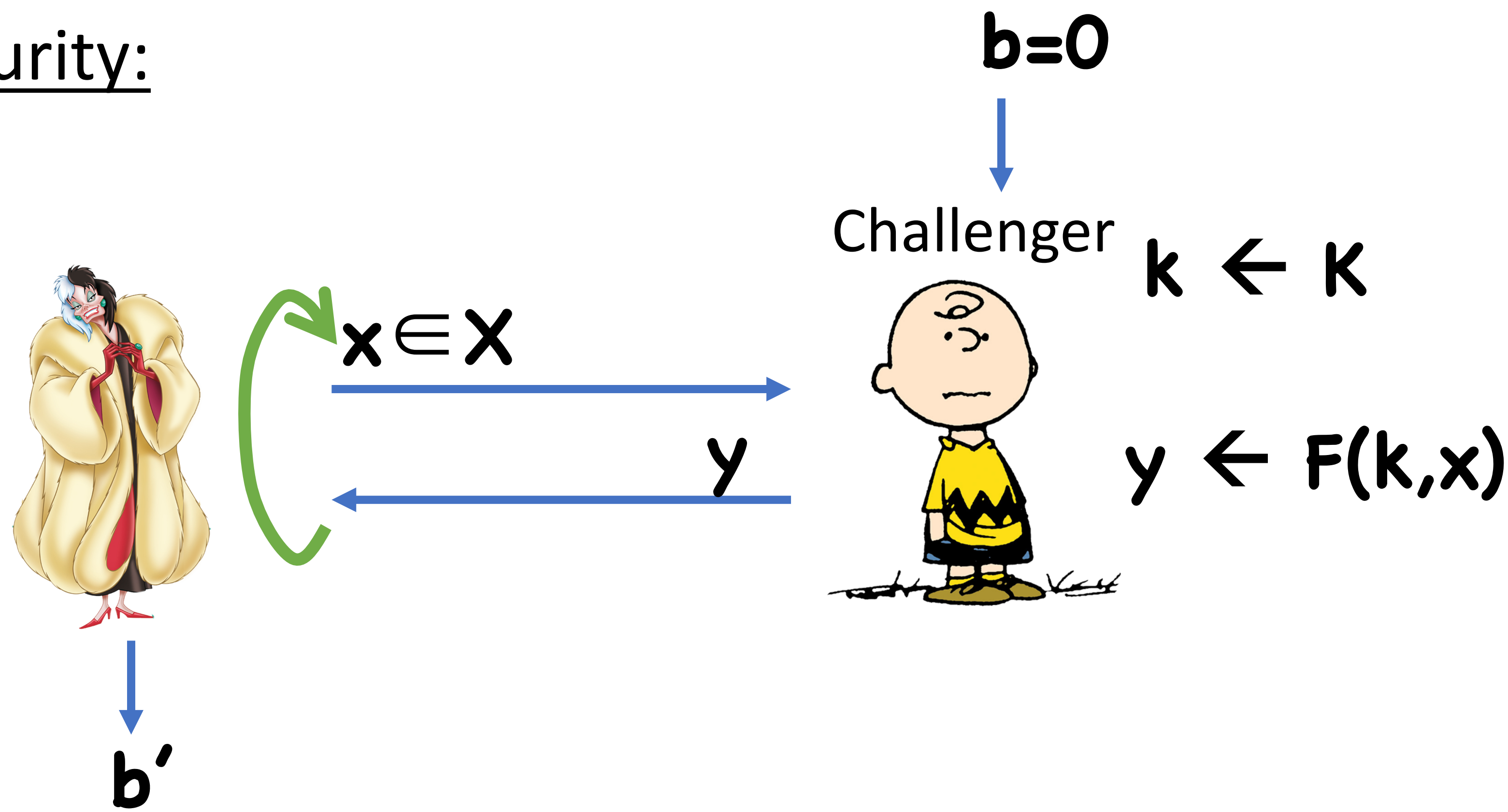
Pseudorandom Functions

Security:



Pseudorandom Functions

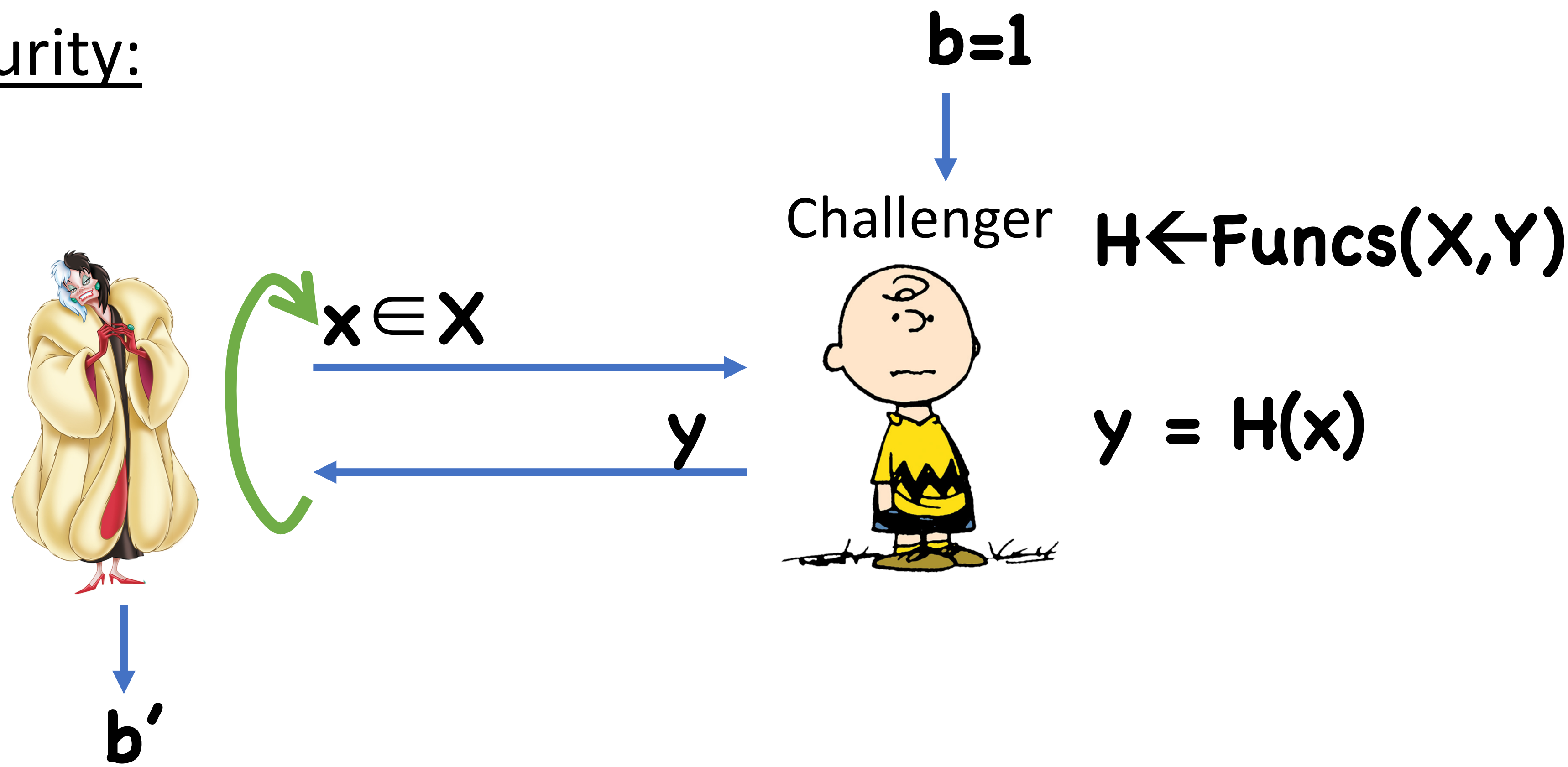
<u>Security:</u>



PRF-Exp_0()

Pseudorandom Functions

<u>Security:</u>



PRF-Exp$_1$( **)**

Using PRFs to Build Encryption

Enc(k, m):
- Choose random $r \leftarrow X$
- 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
- $m' = c \oplus y = y \oplus m \oplus y = m$

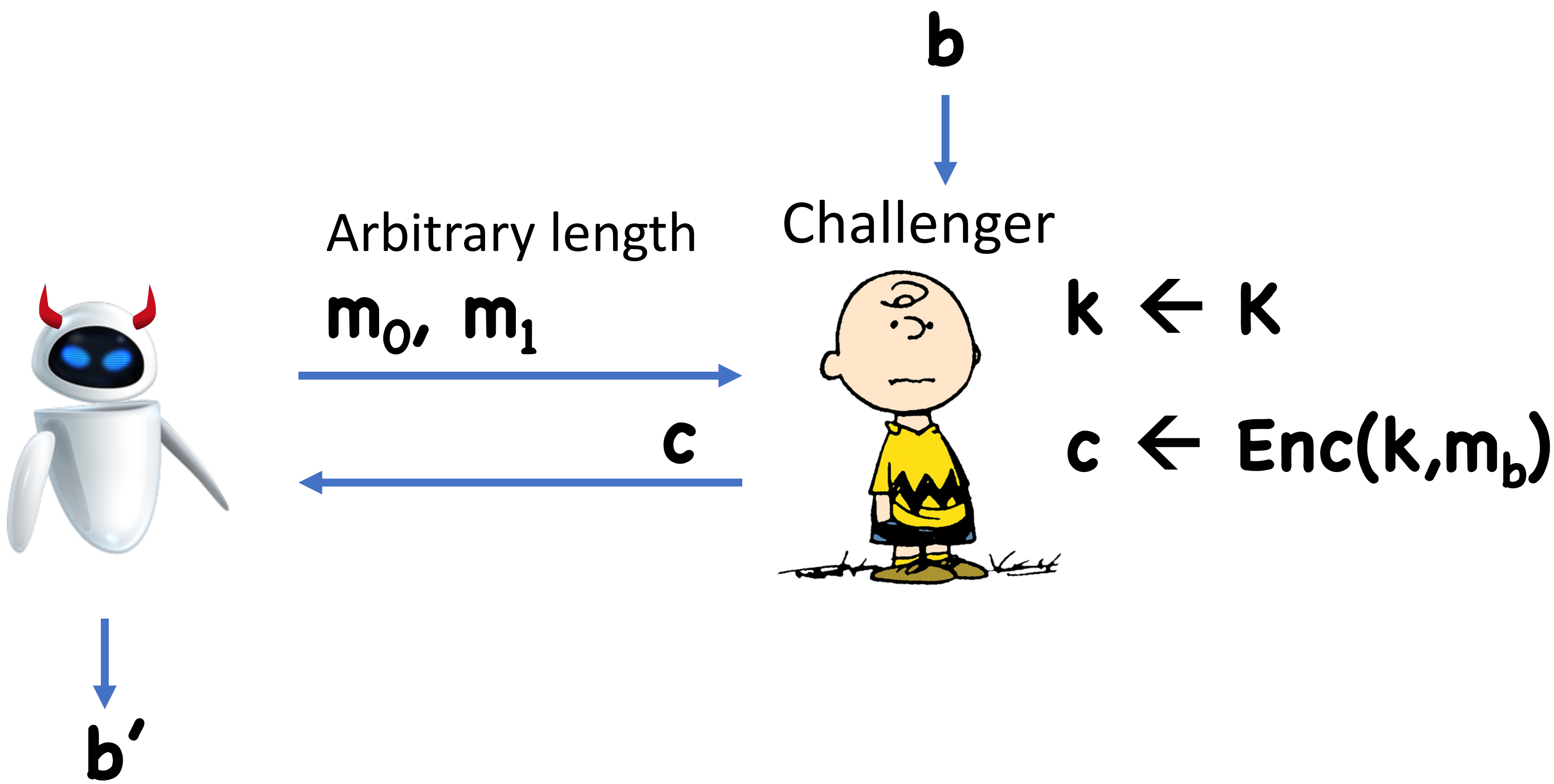
Today

Security for arbitrary-length messages

Block ciphers

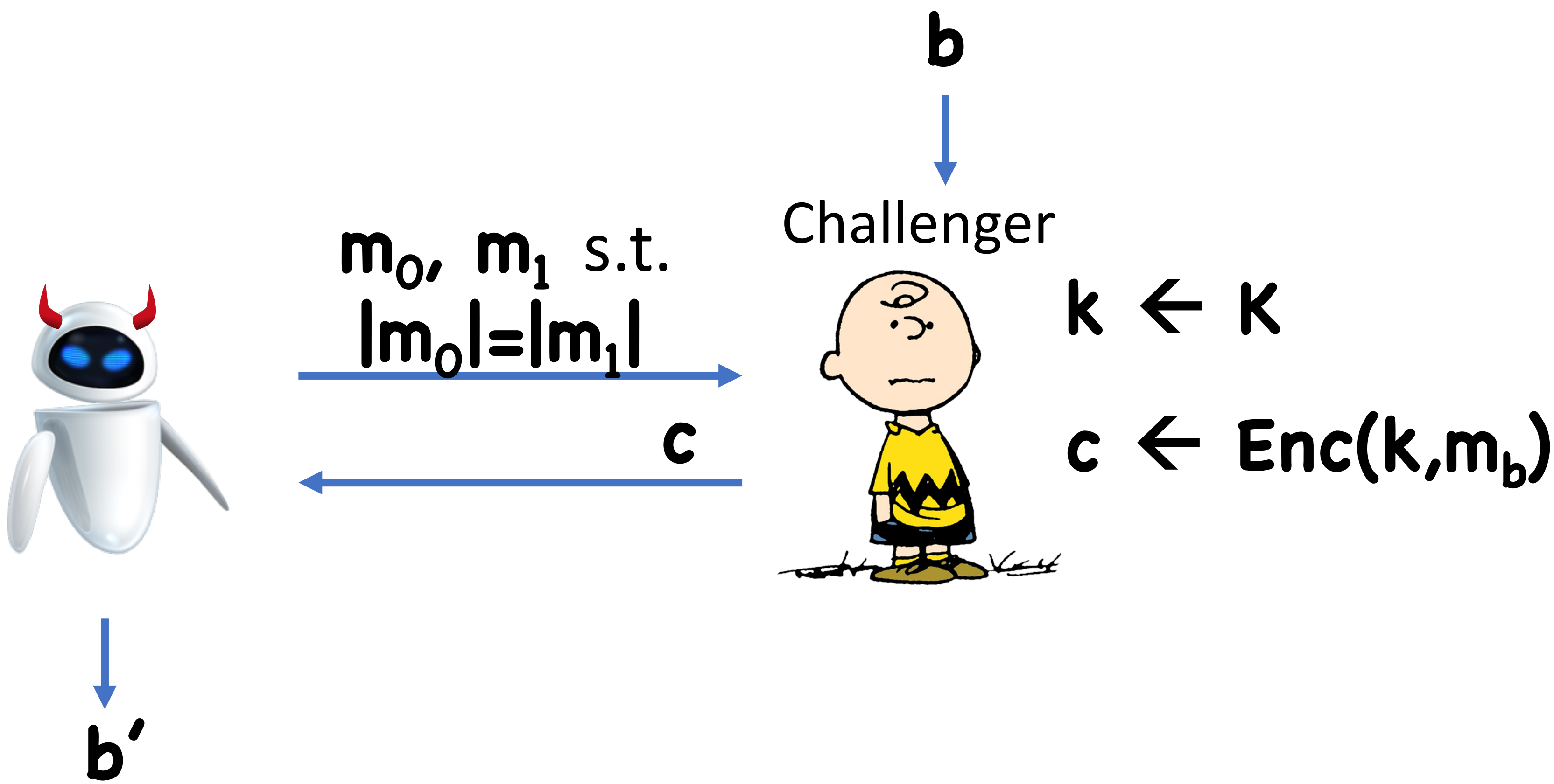
Modes of operation

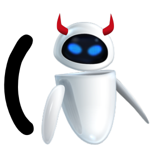
Security for Arbitrary-Length Messages



Impossible in general to hide message length

Security for Arbitrary-Length Messages



IND-Exp$_b$( **)**

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,…, |m|**, run $c_i \leftarrow \text{Enc}(k, m_i)$

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

Dec'(k, (c₁, …, c_l)):

For **i=1,…, l**, run $m_i \leftarrow \text{Dec}(k, c_i)$

Output $m = m_1 m_2 \ldots, m_l$

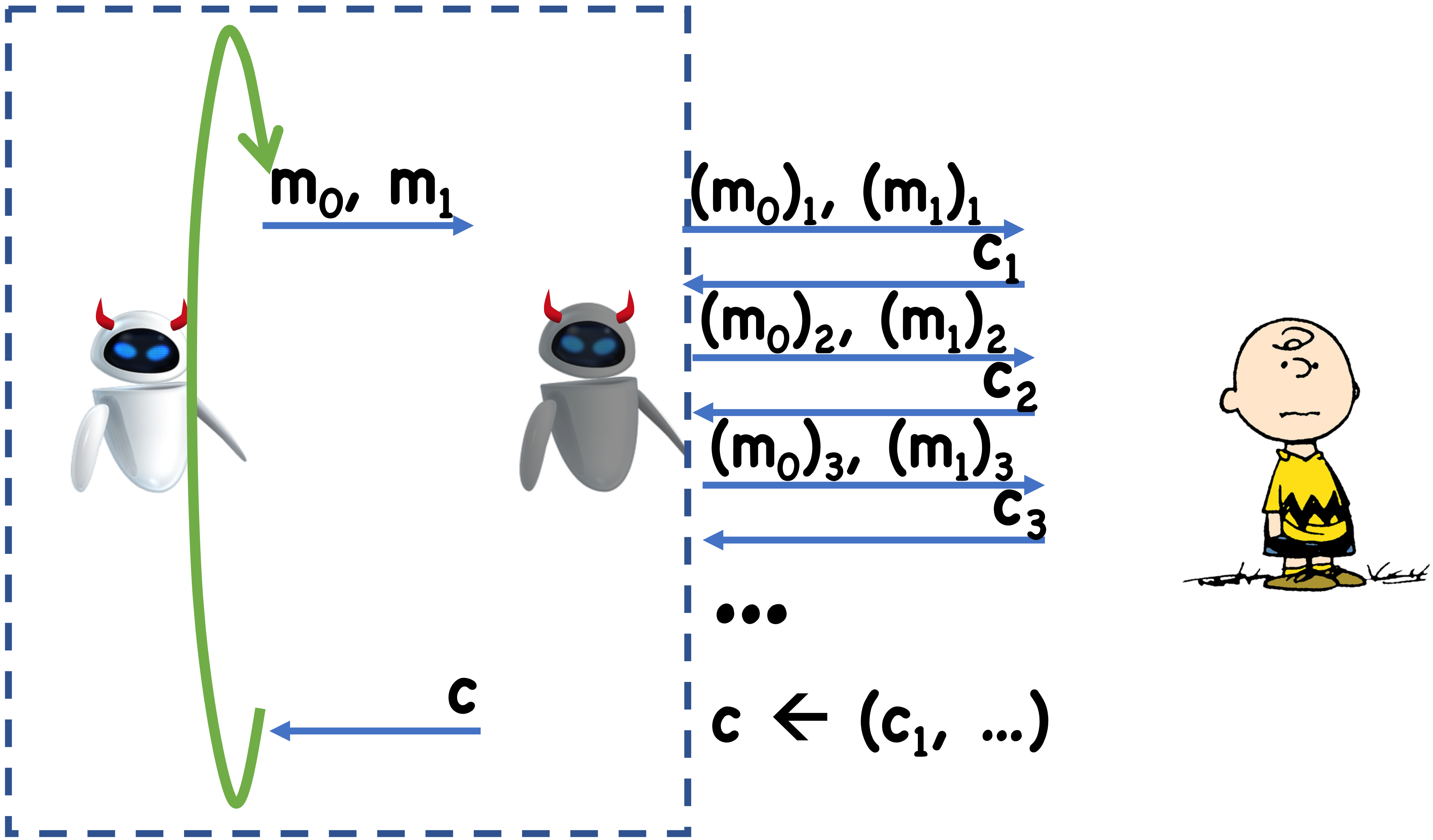
Theorem: If **(Enc,Dec)** is **(t,q,ε)**-LoR secure, then **(Enc',Dec')** is **(t–t',q/n,ε)**-LoR secure for messages of length up to **n**

Proof

Assume toward contradiction that there exists a  running in time at most **t−t'**, making **q/n** LoR queries on messages of length up to **n**, which has advantage **ε** in breaking **(Enc',Dec')**

Construct  that has advantage **ε** in breaking **(Enc,Dec)**

Proof (sketch)



Better Constructions Using PRFs

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

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

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

Solution 1: Add PRG/Stream Cipher

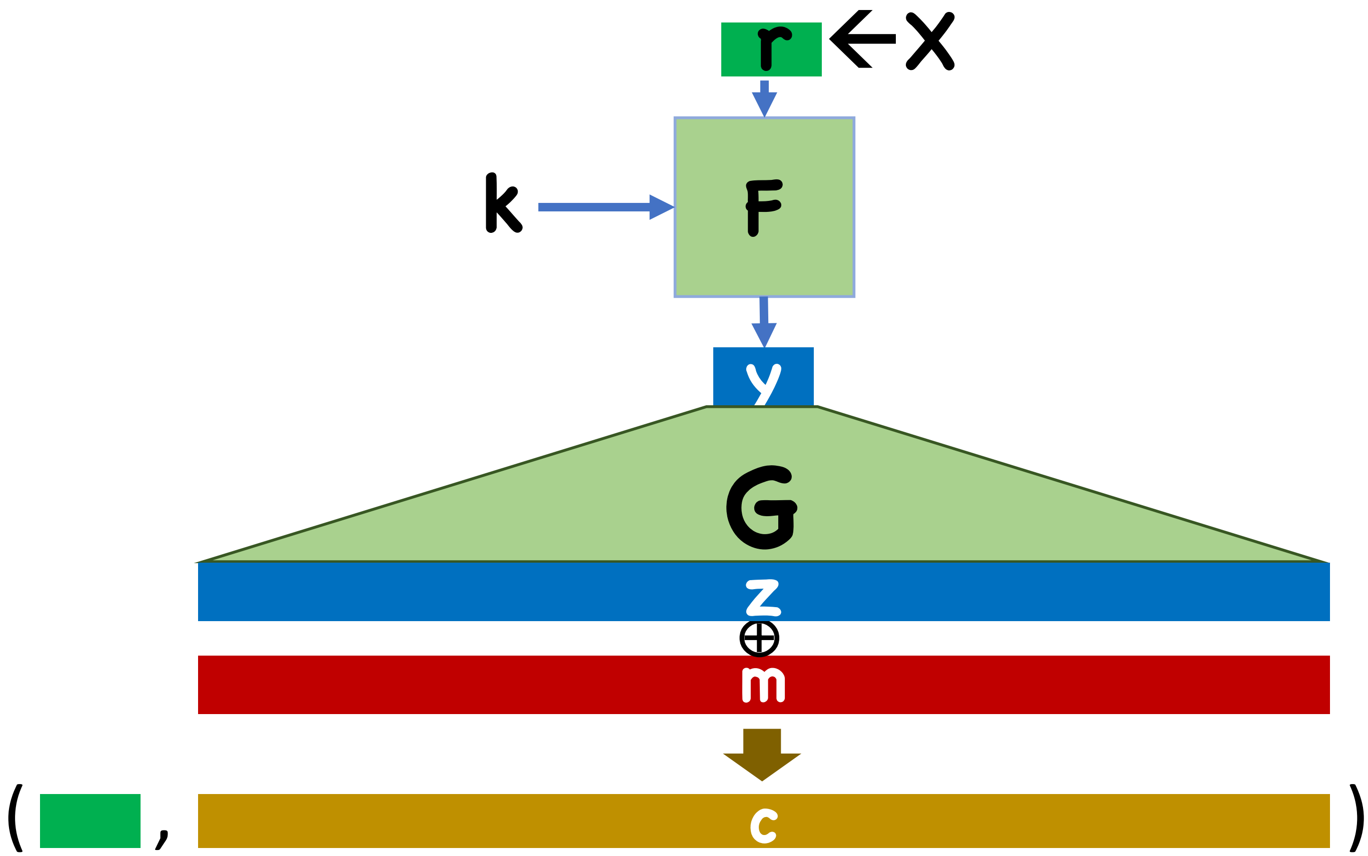
Enc(k, m):

- 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



Proof Sketch

Hybrid 0: $(m_0,m_1) \rightarrow (r,\ \ G(F(k,r))\oplus m_0)$

Hybrid 1: $(m_0,m_1) \rightarrow (r,\ \ G(s)\oplus m_0)$

Hybrid 2: $(m_0,m_1) \rightarrow (r,\ \ t\oplus m_0)$

Hybrid 3: $(m_0,m_1) \rightarrow (r,\ \ t\oplus m_1)$

Hybrid 4: $(m_0,m_1) \rightarrow (r,\ \ G(s)\oplus m_1)$

Hybrid 5: $(m_0,m_1) \rightarrow (r,\ \ G(F(k,r))\oplus m_1)$

Solution 2: Counter Mode

Enc(k, m):

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

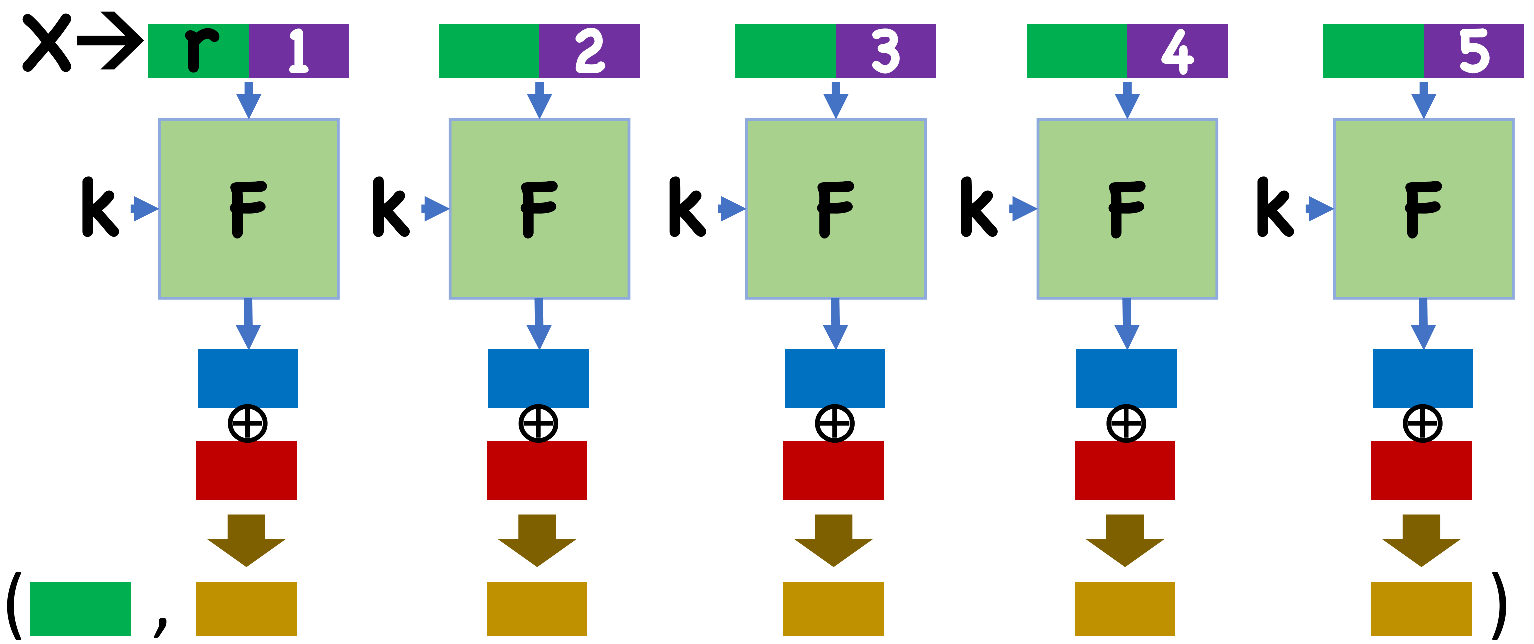
Write $\mathbf{i}$ as $\boldsymbol{\lambda/2}$-bit string

Dec(k, (r,c)):

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

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

Solution 2: Counter Mode



Block ciphers/Pseudorandom Permutations

Pseudorandom Permutations
(also known as block ciphers)

Functions that "look like" random **permutations**

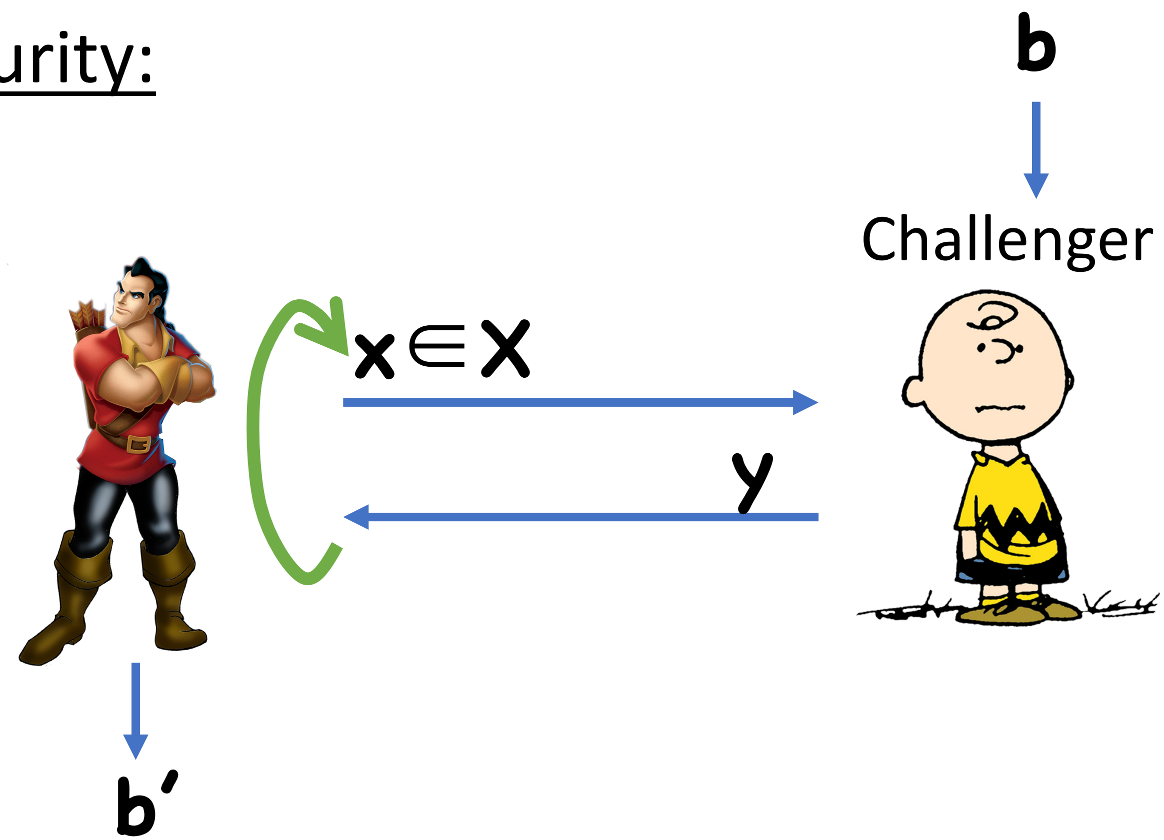
Syntax:

- Key space $\mathbf{K}$ (usually $\{0,1\}^\lambda$)
- Domain=Range= $\mathbf{X}$ (usually $\{0,1\}^n$)
- Function $\mathbf{F:K \times X \rightarrow X}$
- Function $\mathbf{F^{-1}:K \times X \rightarrow X}$

Correctness: $\forall \mathbf{k,x,\ F^{-1}(k,\ F(k,\ x)\ ) = x}$

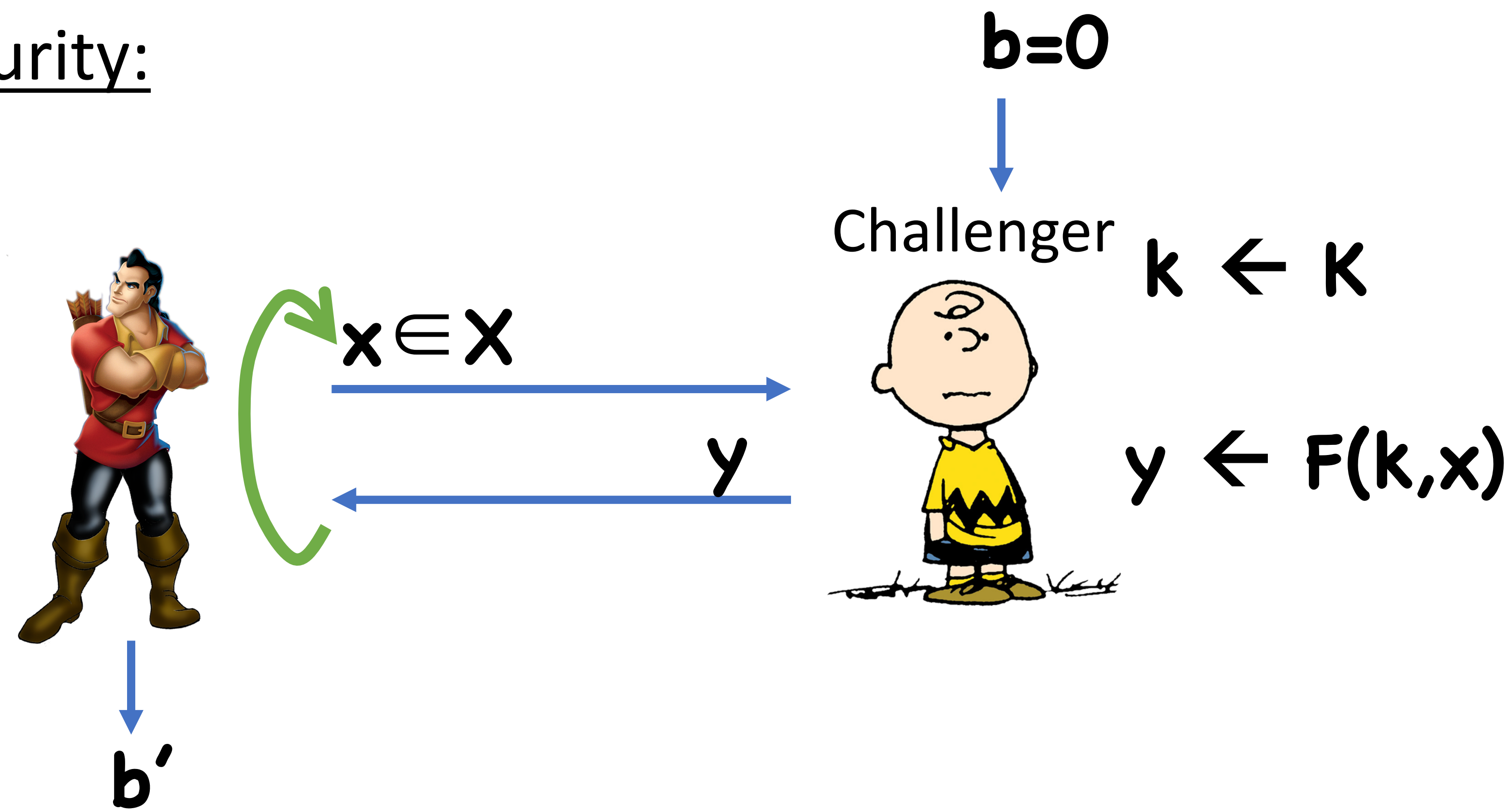
Pseudorandom Permutations

<u>Security:</u>



Pseudorandom Permutations

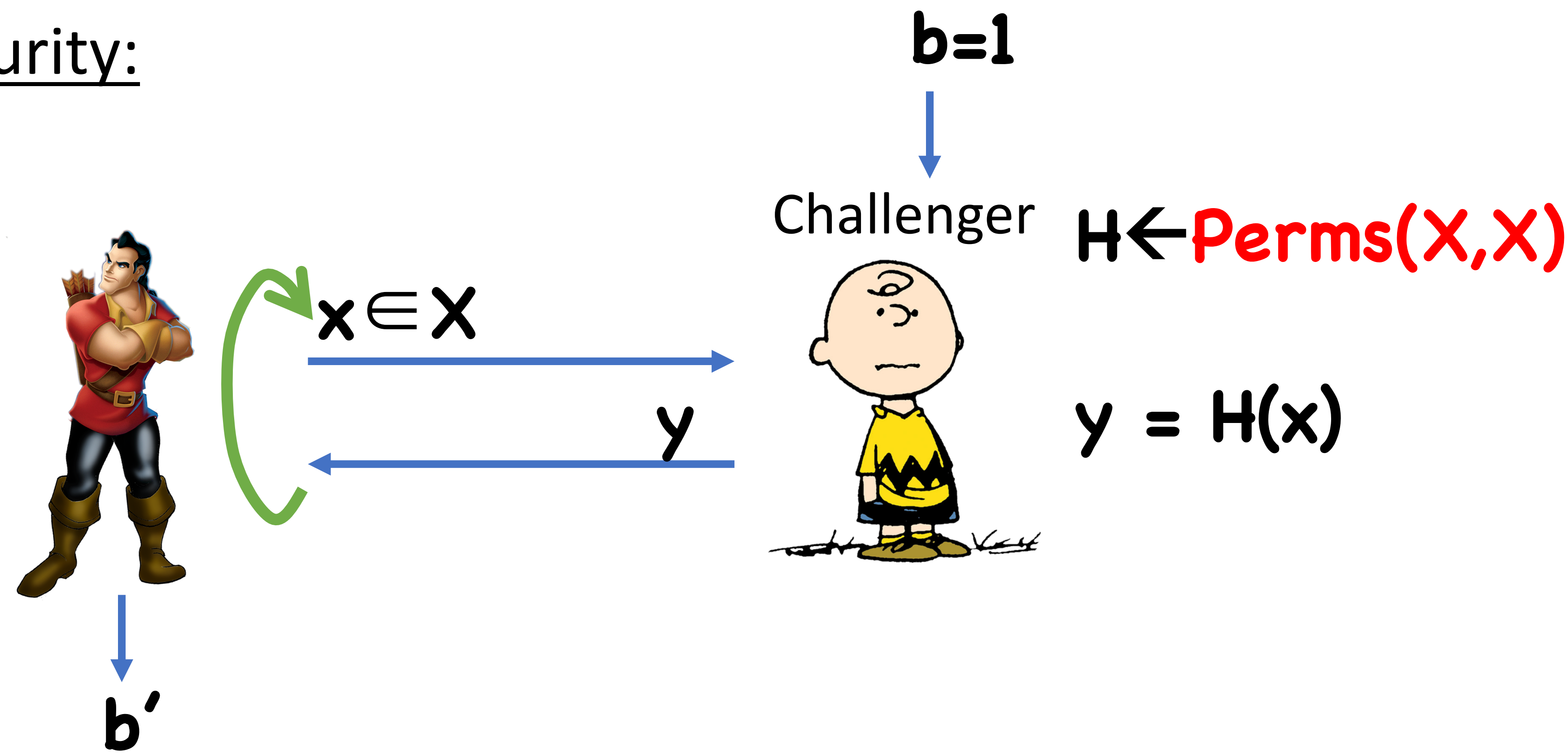
<u>Security:</u>



PRF-Exp$_0$()

Pseudorandom Permutations

Security:



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

PRF Security Definition

Definition: F is a **(t,q,ε)-**secure PRP if, for all  running in time at most **t** and making at most **q** queries,

| Pr[1←PRF-Exp$_0$( **)]**

− Pr[1←PRF-Exp$_1$( **)] | ≤ ε**

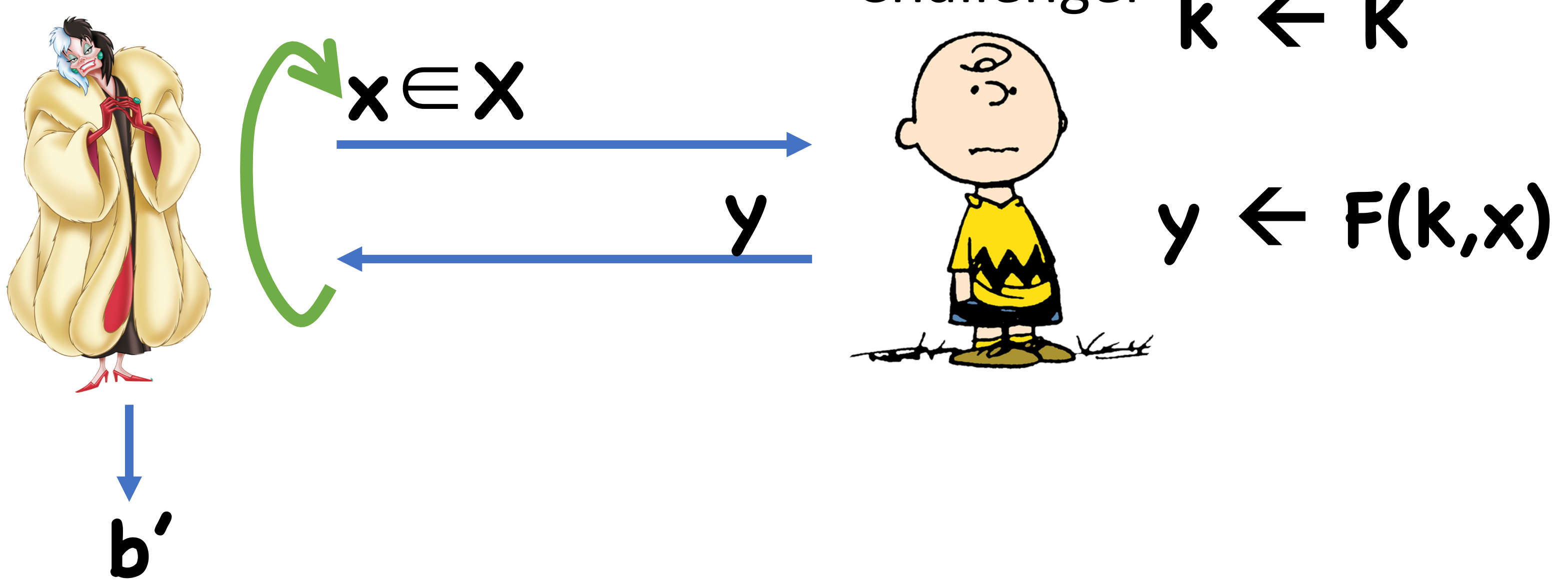
Theorem: A PRP (F,F^{-1}) is (t,q,ε)-secure iff F is $(t,q,\varepsilon+q^2/2|X|)$-secure as a PRF

Proof

Secure as PRP $\Rightarrow$ Secure as PRF

- Assume , hybrids

<u>Hybrid 0:</u>

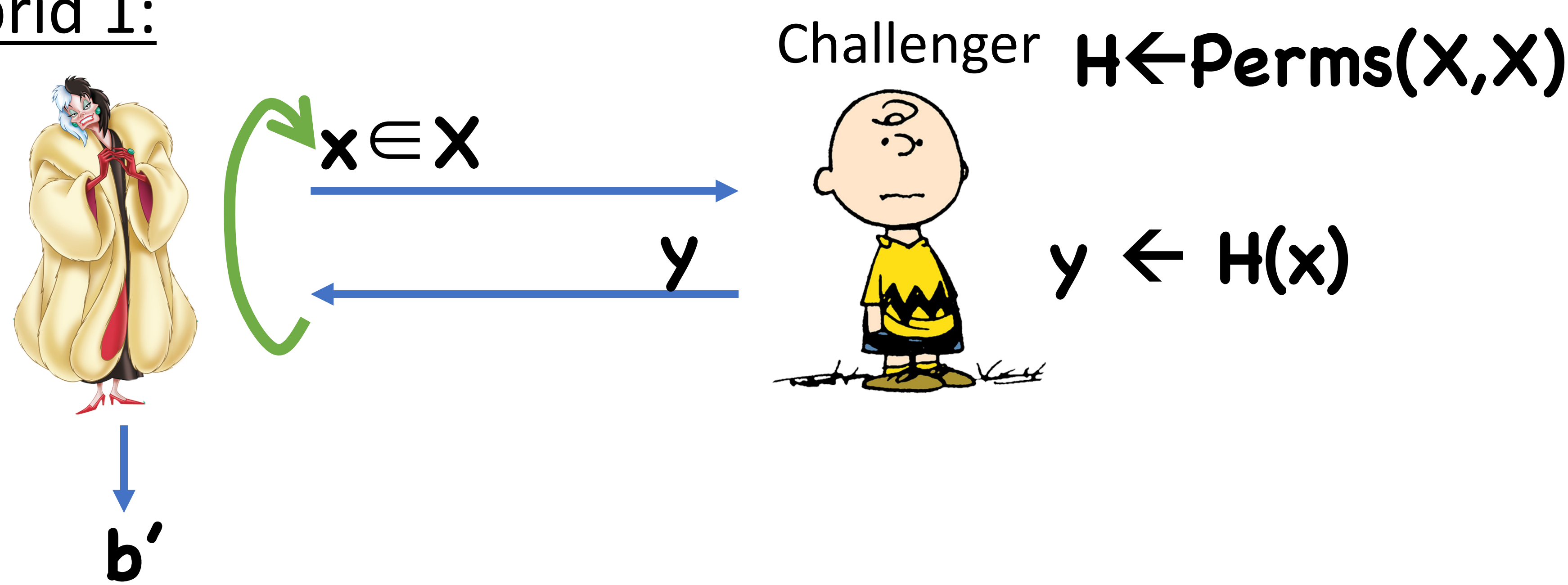


Proof

Secure as PRP ⇒ Secure as PRF

- Assume , hybrids

Hybrid 1:

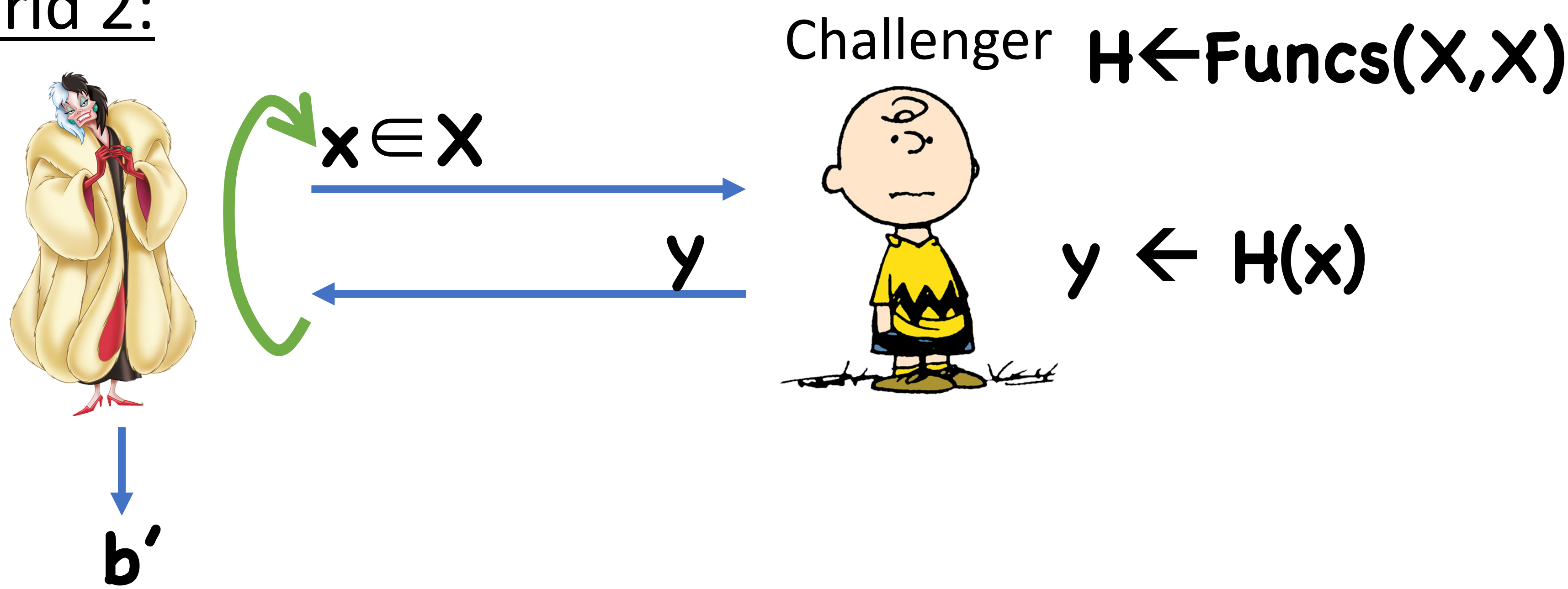


Proof

Secure as PRP $\Rightarrow$ Secure as PRF

- Assume , hybrids

<u>Hybrid 2:</u>



Proof

Secure as PRP $\Rightarrow$ Secure as PRF

- Assume , hybrids

Hybrids 0 and 1 are indistinguishable by PRP security

Hybrids 1 and 2?

- In Hybrid 1,  sees random **distinct** answers
- In Hybrid 2,  sees random answers
- Except with probability $\approx \mathbf{q^2/2|X|}$, random answers will be distinct anyway

Proof

Secure as PRF $\Rightarrow$ Secure as PRP

- Assume , hybrids

Proof essentially identical to other direction

Suppose (F, F^{-1}) is a secure PRP

Is (F^{-1}, F) also a secure PRP?

Counter Example

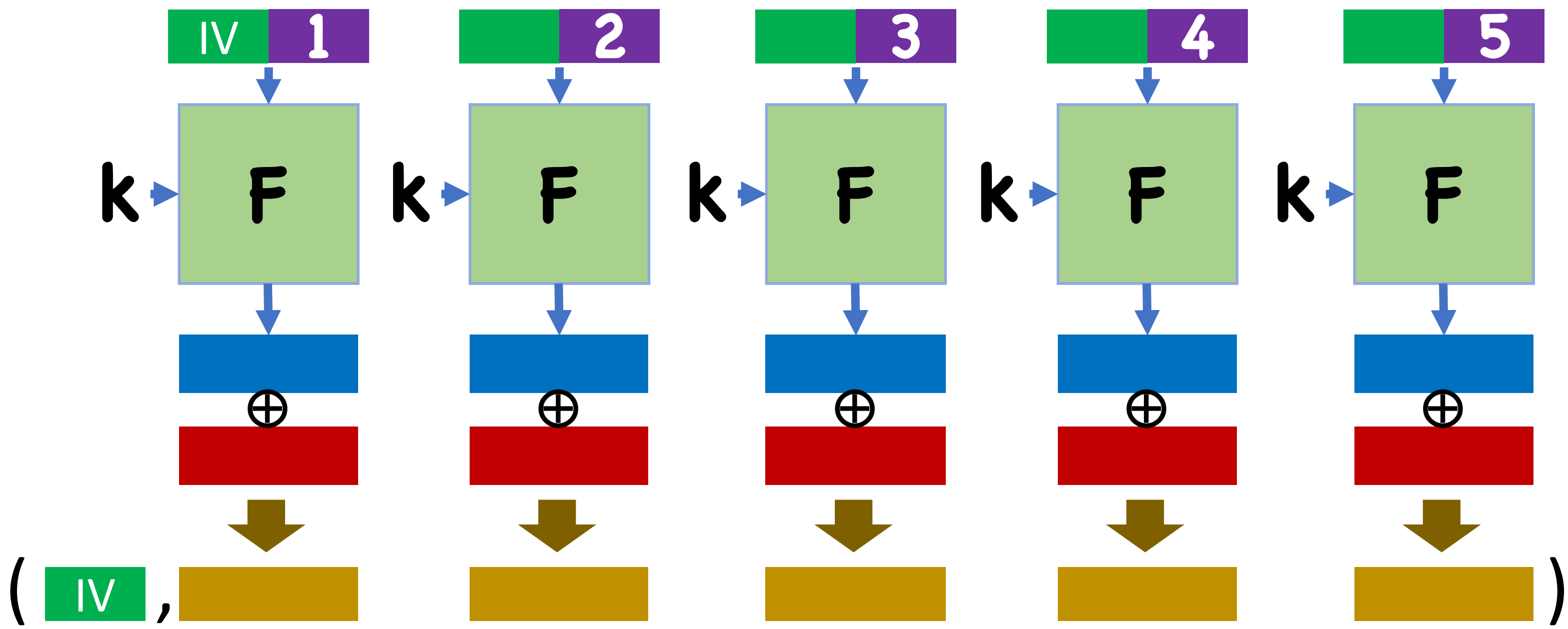
Suppose (F, F^{-1}) is a secure PRP. Assume $X = \{0,1\}^n$

Define (H, H^{-1}) as follows:

- Given k, let i be smallest input such that $F^{-1}(i)$ begins with a 0
- Let $x_0 = F^{-1}(0^n)$, $x_1 = F^{-1}(i)$
- $H(k,x) = \begin{cases} 0^n \text{ if } x = x_1 \\ i \text{ if } x = x_0 \\ F(k,x) \text{ otherwise} \end{cases}$

How to use block ciphers for encryption

Counter Mode (CTR)



Electronic Code Book (ECB)

Enc(k, m):

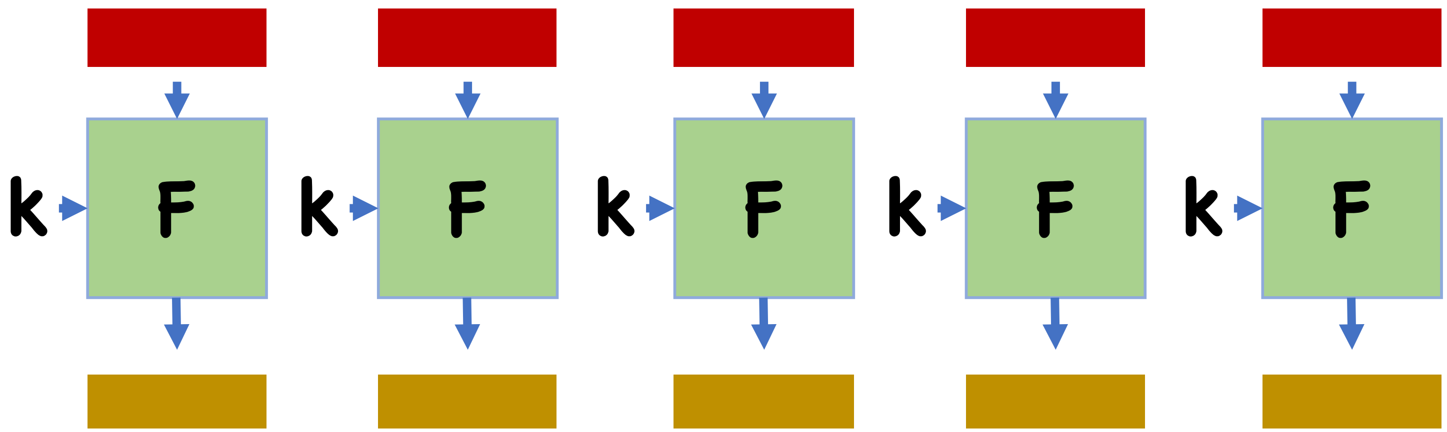
- Break **m** into **t** blocks m_i of **n** bits
- For each block m_i, let $c_i = F(k, m_i)$
- Output $c = (c_1, \ldots, c_t)$

Dec(k, c):

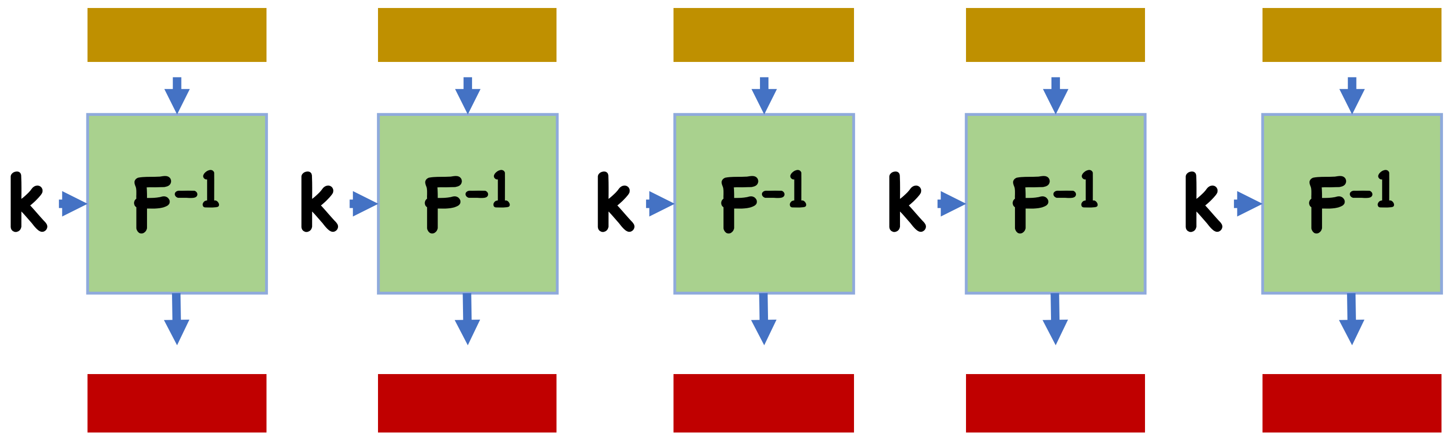
- Break **c** into **t** blocks c_i of **n** bits
- For each block c_i, let $m_i = F^{-1}(k, c_i)$
- Output $m = (m_1, \ldots, m_t)$

substitution cipher for **n**-bit alphabet

Electronic Code Book (ECB)



ECB Decryption



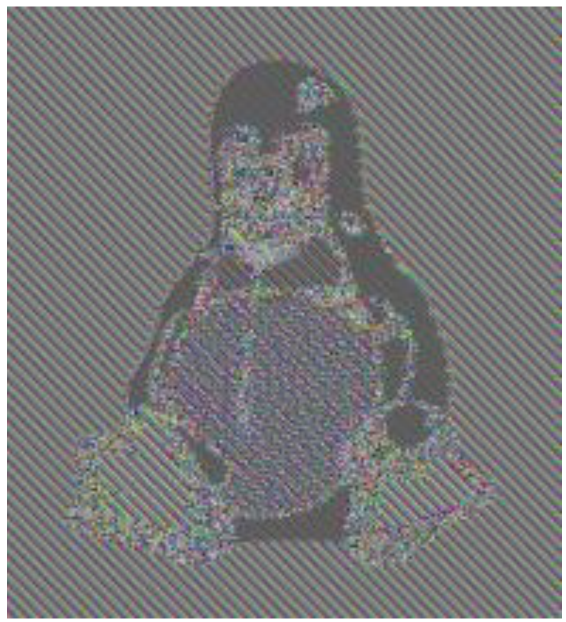
Security of ECB?

Is ECB mode CPA secure?

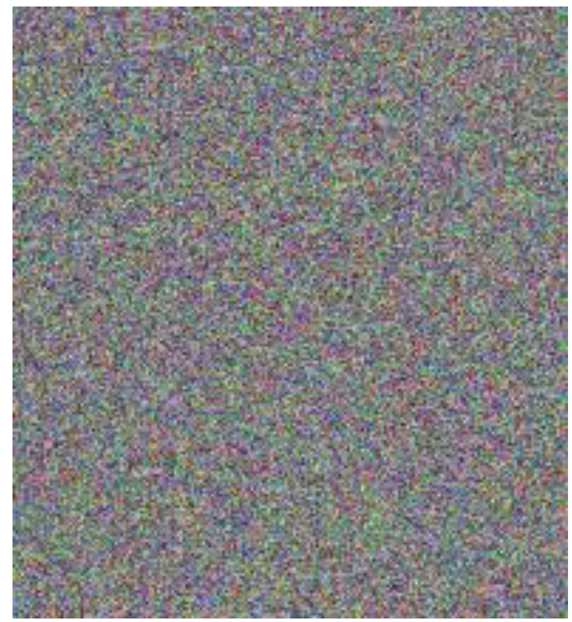
Is ECB mode *one-time* secure?

Security of ECB

Plaintex

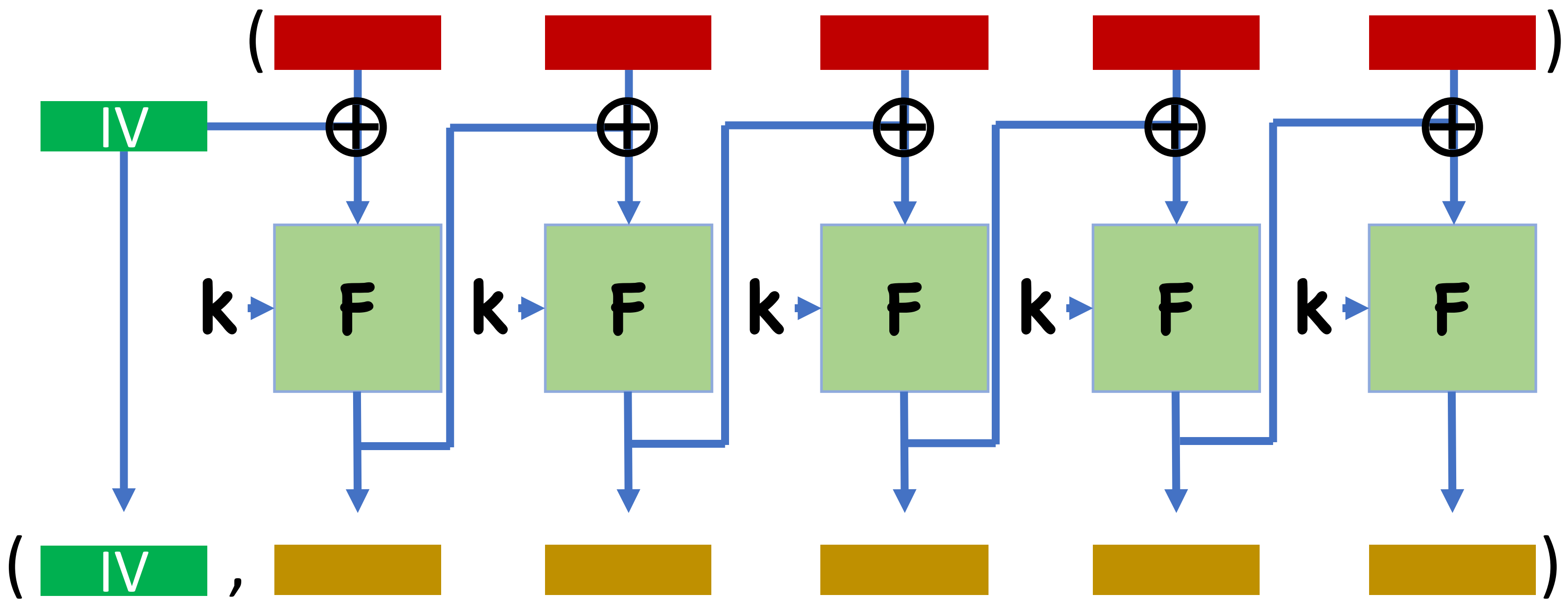


Ciphertext



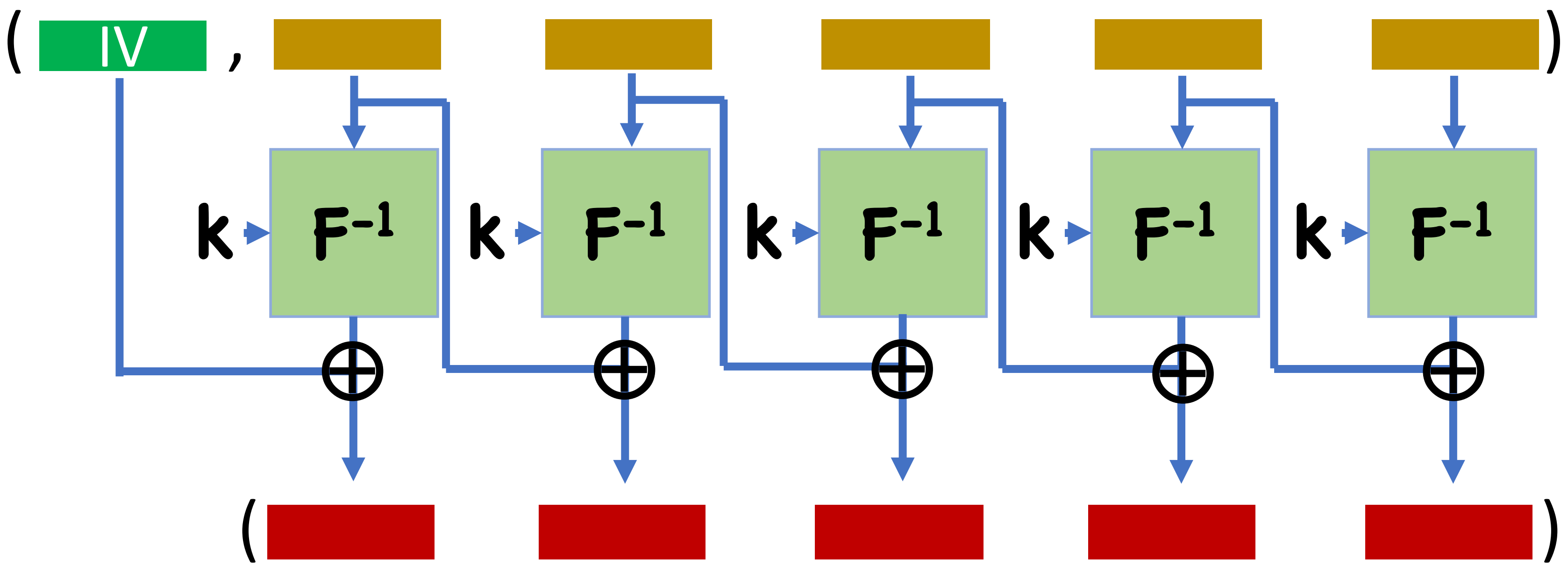
Ideal

Cipher Block Chaining (CBC) Mode



(For now, assume all messages are multiples of the block length)

CBC Mode Decryption



Theorem: If (F,F^{-1}) is a (t,q,ε)-secure pseudorandom permutation, then CBC mode encryption is $(t-t', q/n, 2\varepsilon+q^2/|X|)$ CPA secure for messages of length up to n.

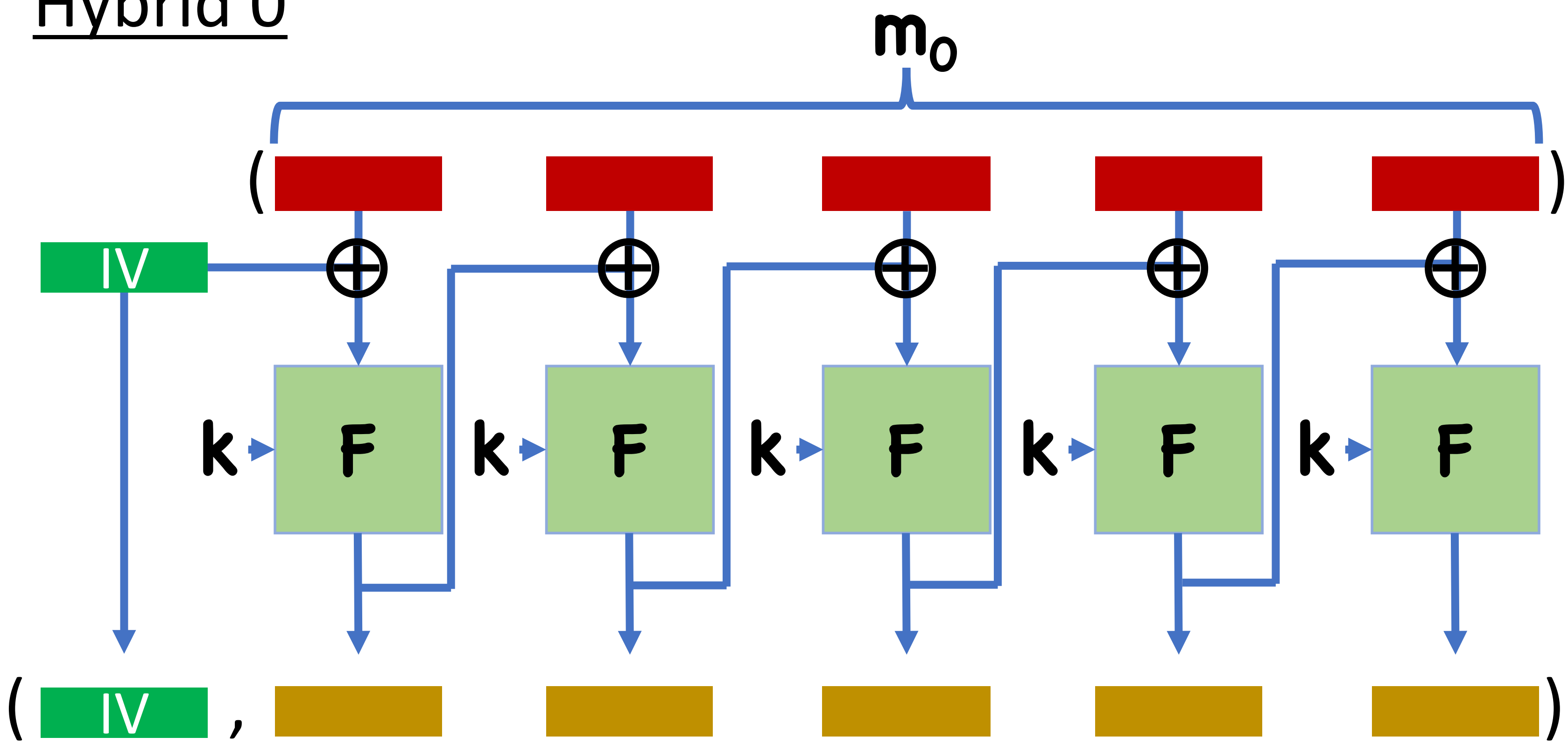
Proof Sketch

Assume toward contradiction an adversary  for CBC mode

Hybrids…

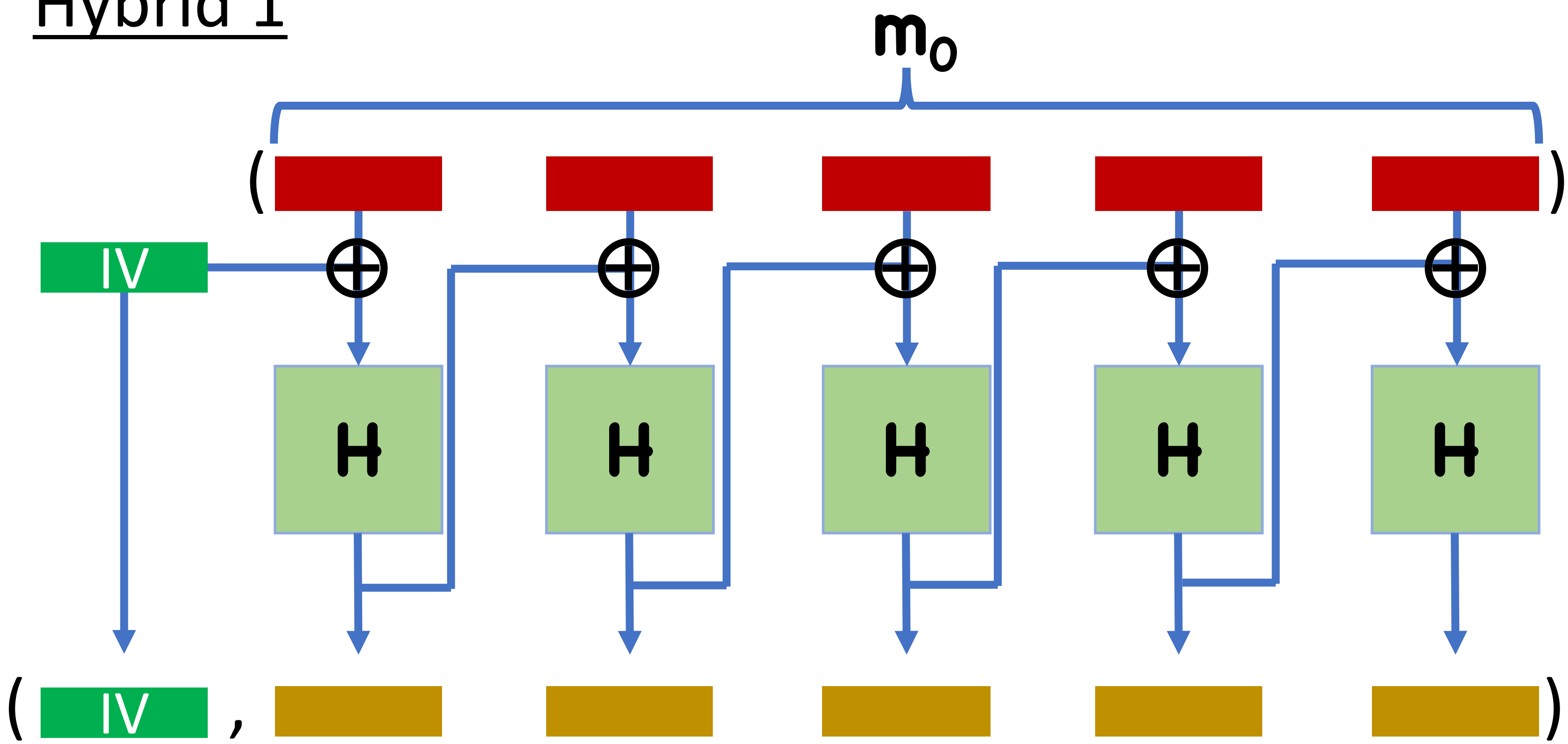
Proof Sketch

Hybrid 0



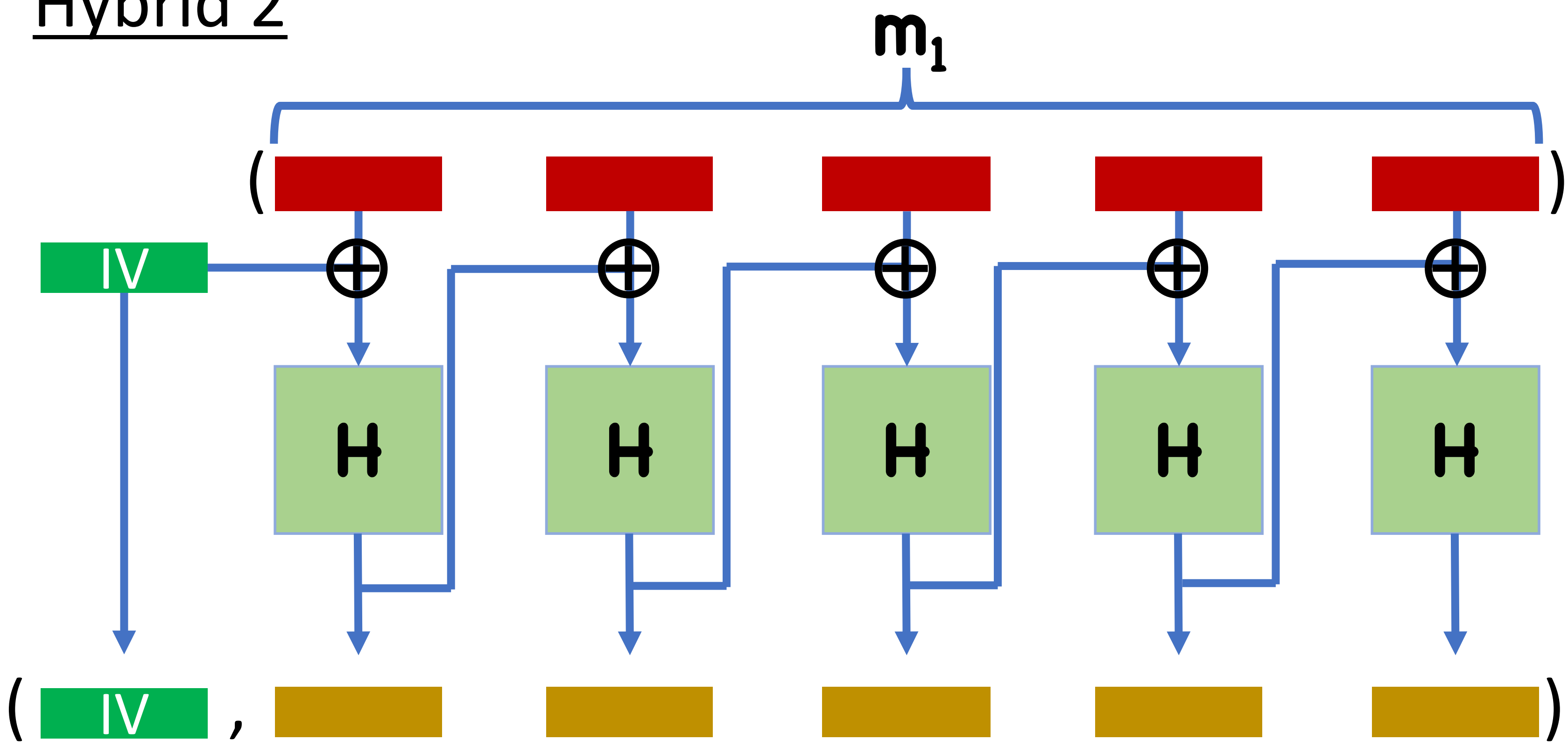
Proof Sketch

<u>Hybrid 1</u>



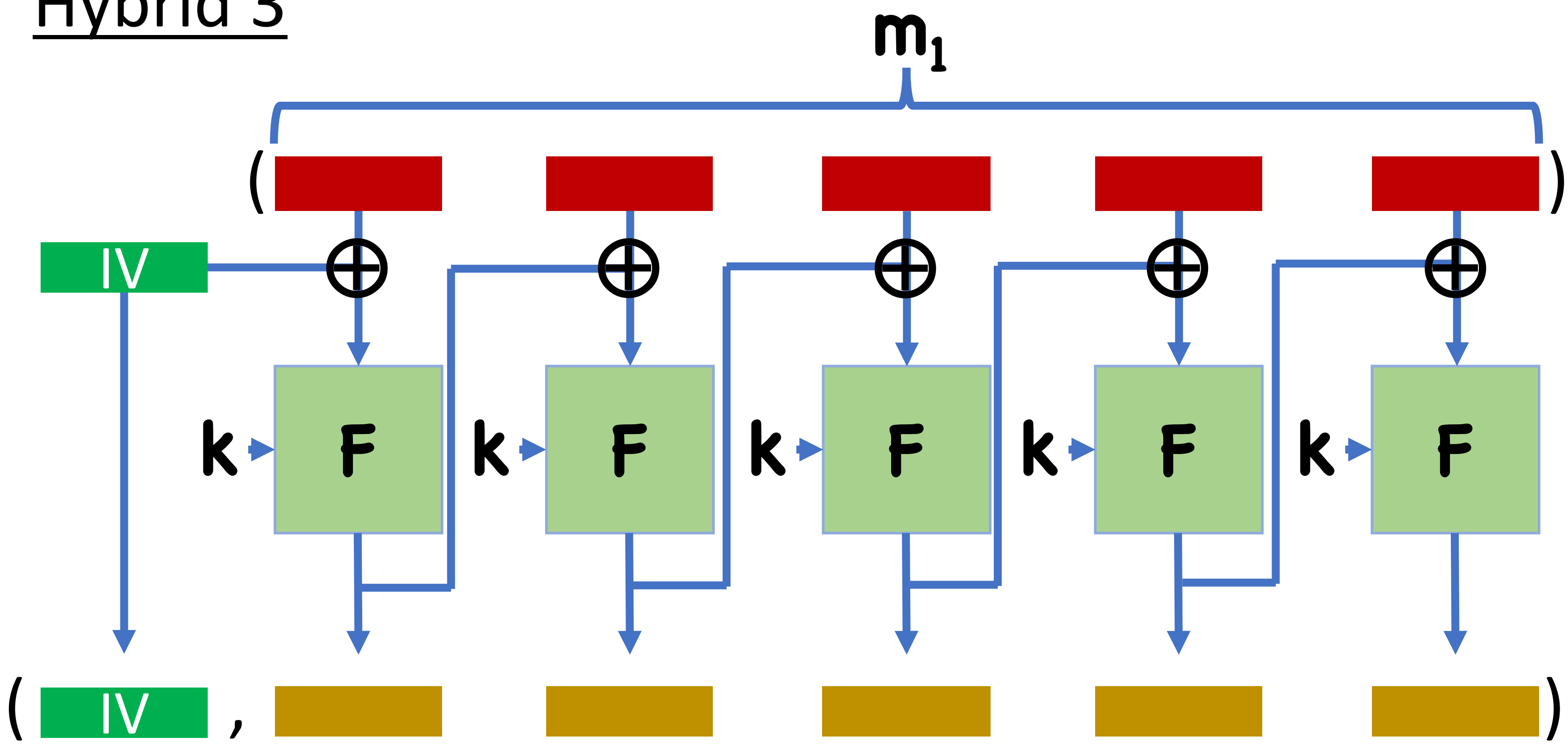
Proof Sketch

Hybrid 2



Proof Sketch

Hybrid 3



Proof Sketch

Hybrid 0,1 differ by replacing calls to **F** with calls to random permutation **H**

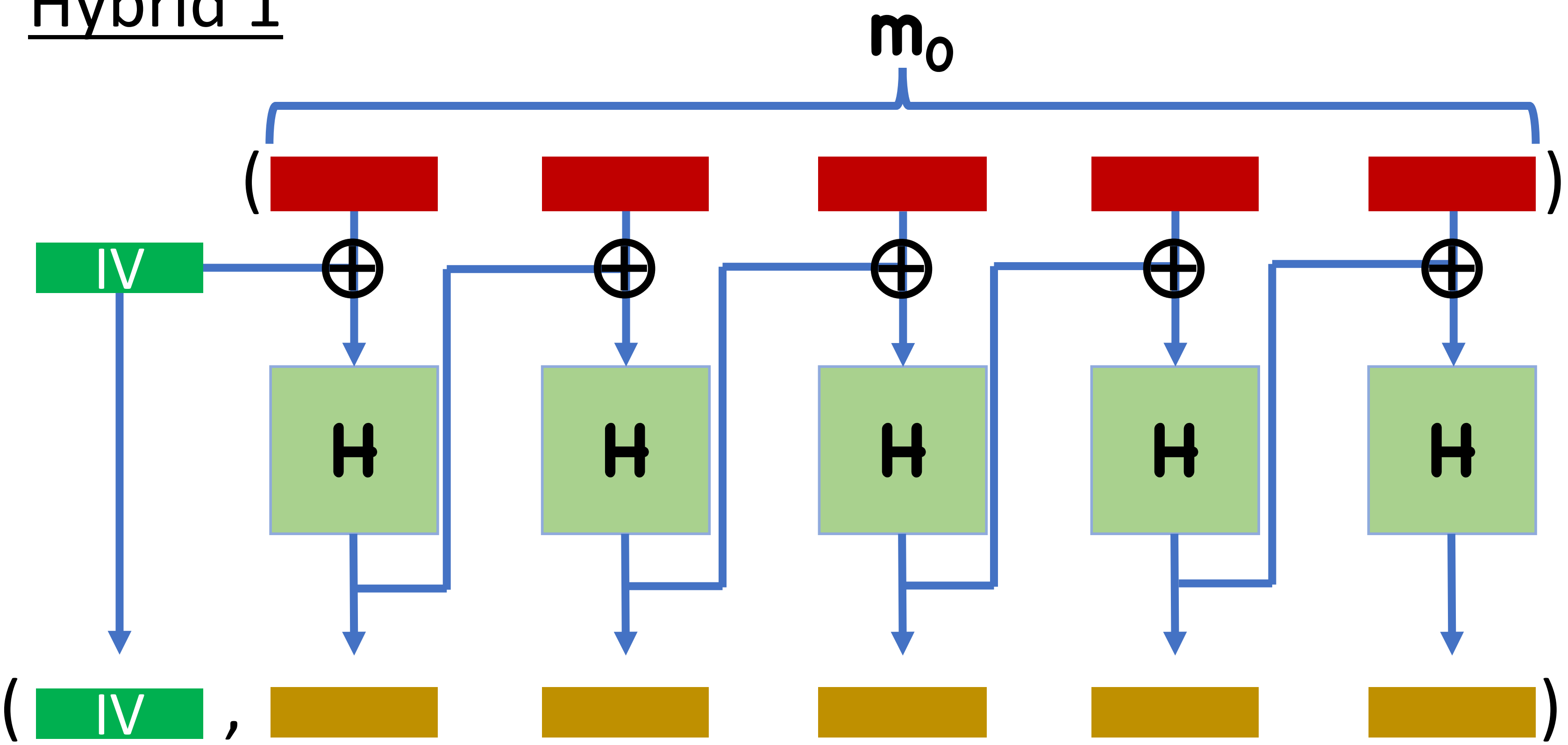
- Indistinguishable by PRP security

Same for Hybrids 2,3

All that is left is to show indistinguishability of 1,2

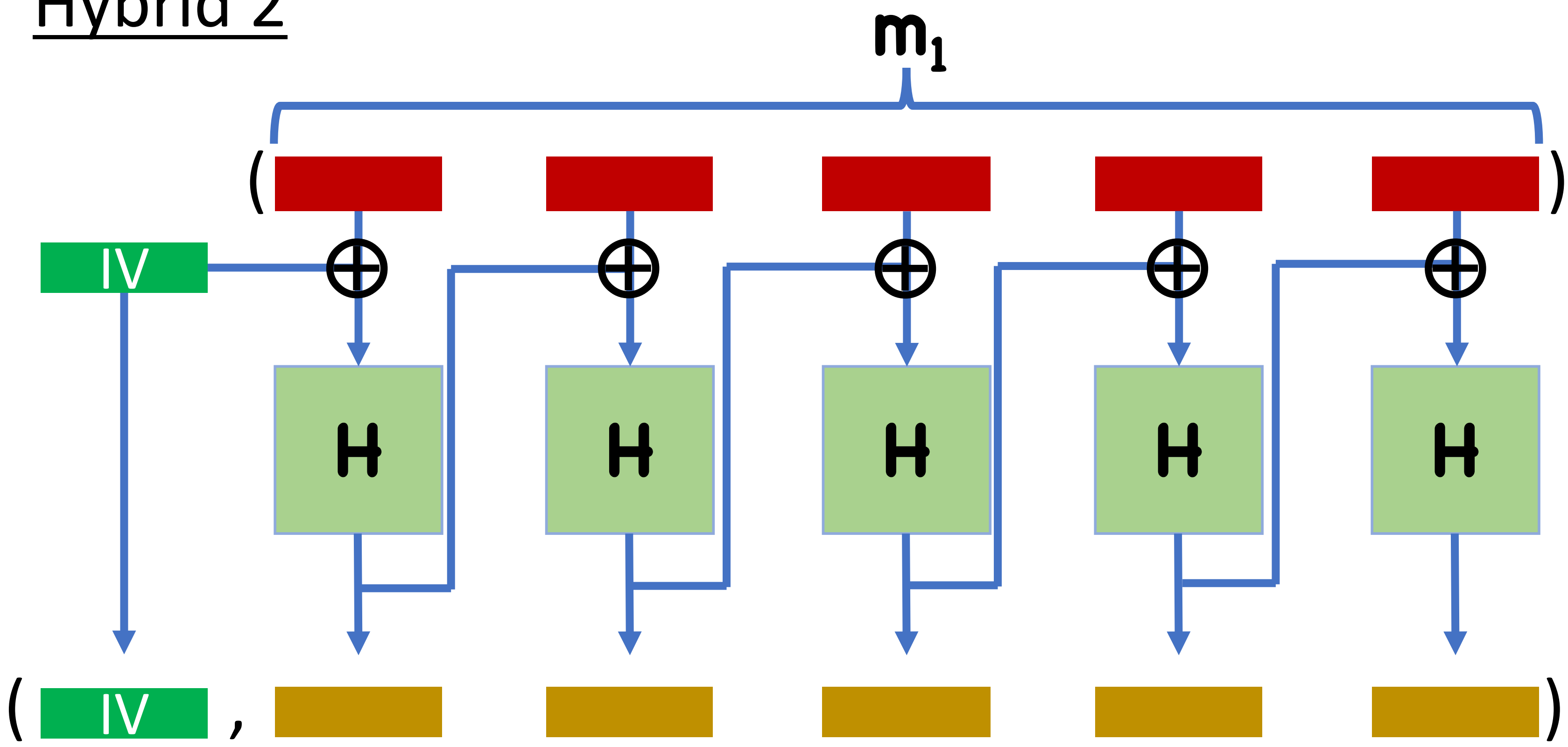
Proof Sketch

Hybrid 1



Proof Sketch

Hybrid 2



Proof Sketch

Idea:

- As long as, say, the sequence of left messages queried by  does not result in two calls to **F** on the same input, all outputs will be random (distinct) outputs
- For each message, first query to **F** will be uniformly random
- Second query gets XORed with output of first query to **F** $\Rightarrow$ ≈ uniformly random

Proof Sketch

Idea:

- Since queries to **F** are (essentially) uniformly random, probability of querying same input twice is exponentially small
- Ciphertexts will be essentially random
- True regardless of encrypting $\mathbf{m_0}$ or $\mathbf{m_1}$

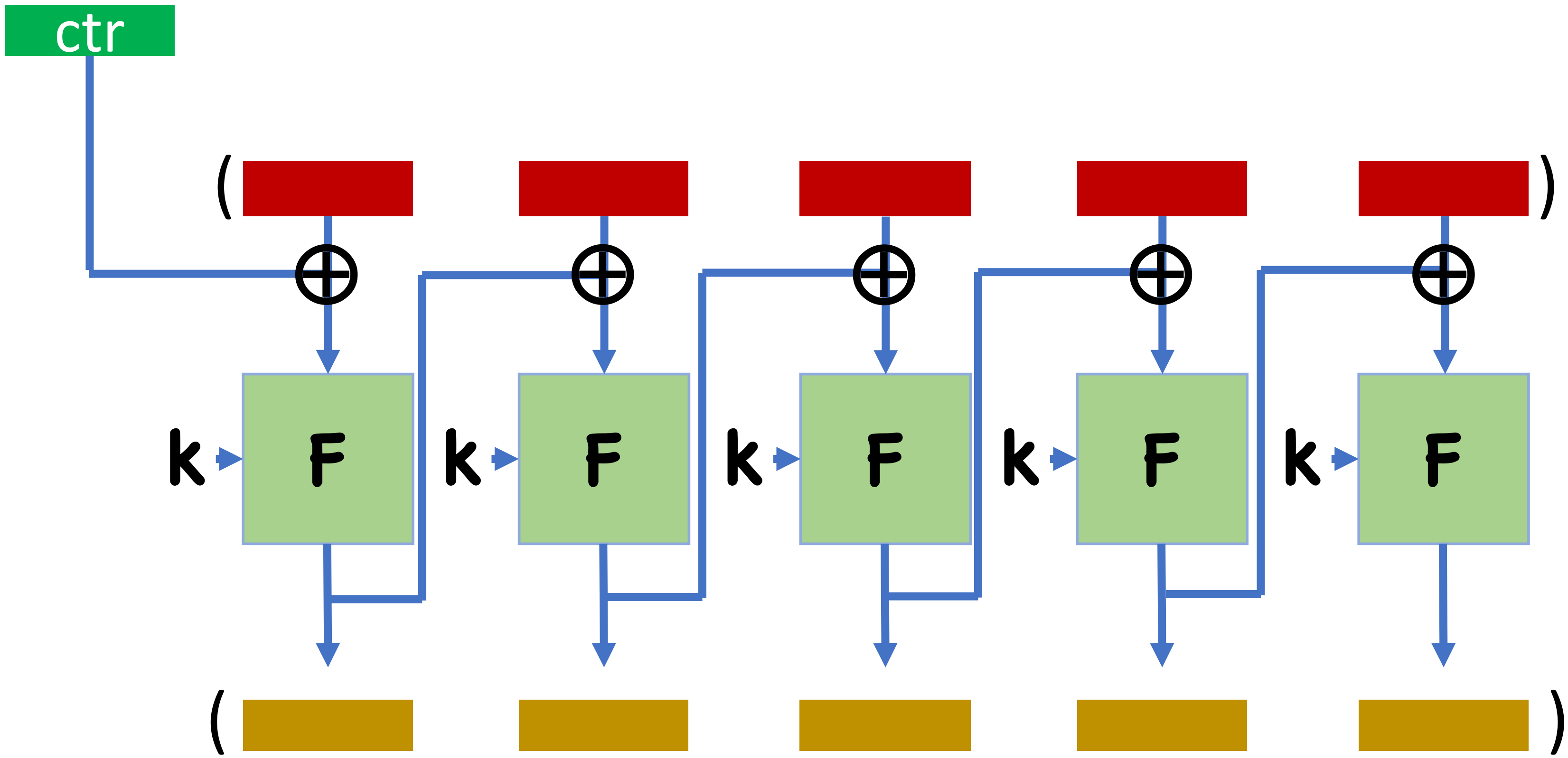
Stateful Variants of CBC

Chained CBC
- IV is set to last block of previous ciphertext

Deterministic IV
- Sender keeps a counter
- To encrypt, IV is set to counter, and counter is incremented

Both variants mean no need to send IV

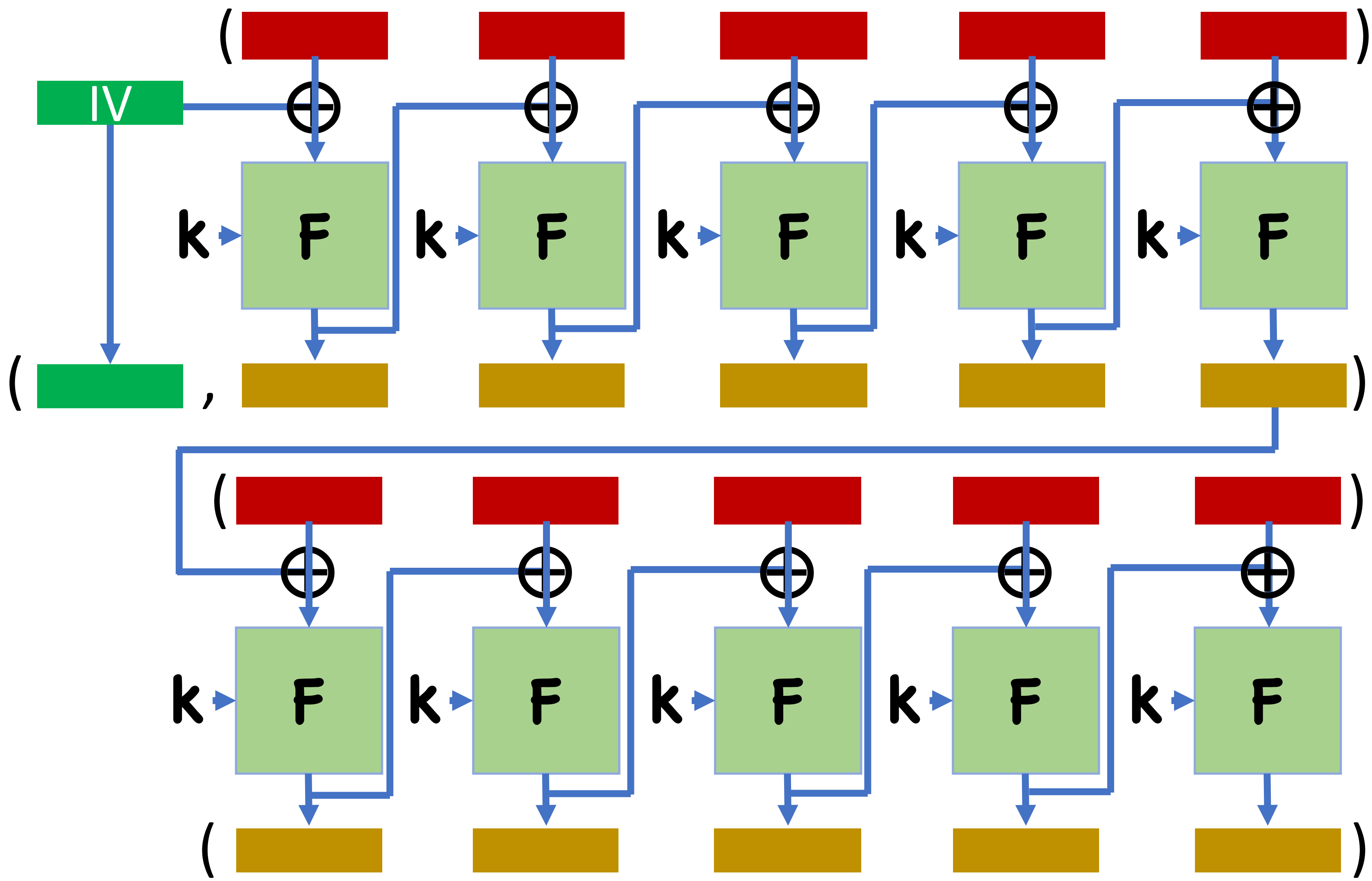
Deterministic IV



ctr++

Is Deterministic IV Secure?

Chained CBC



Is Chained CBC Secure?

CBC Mode with Predictable IV

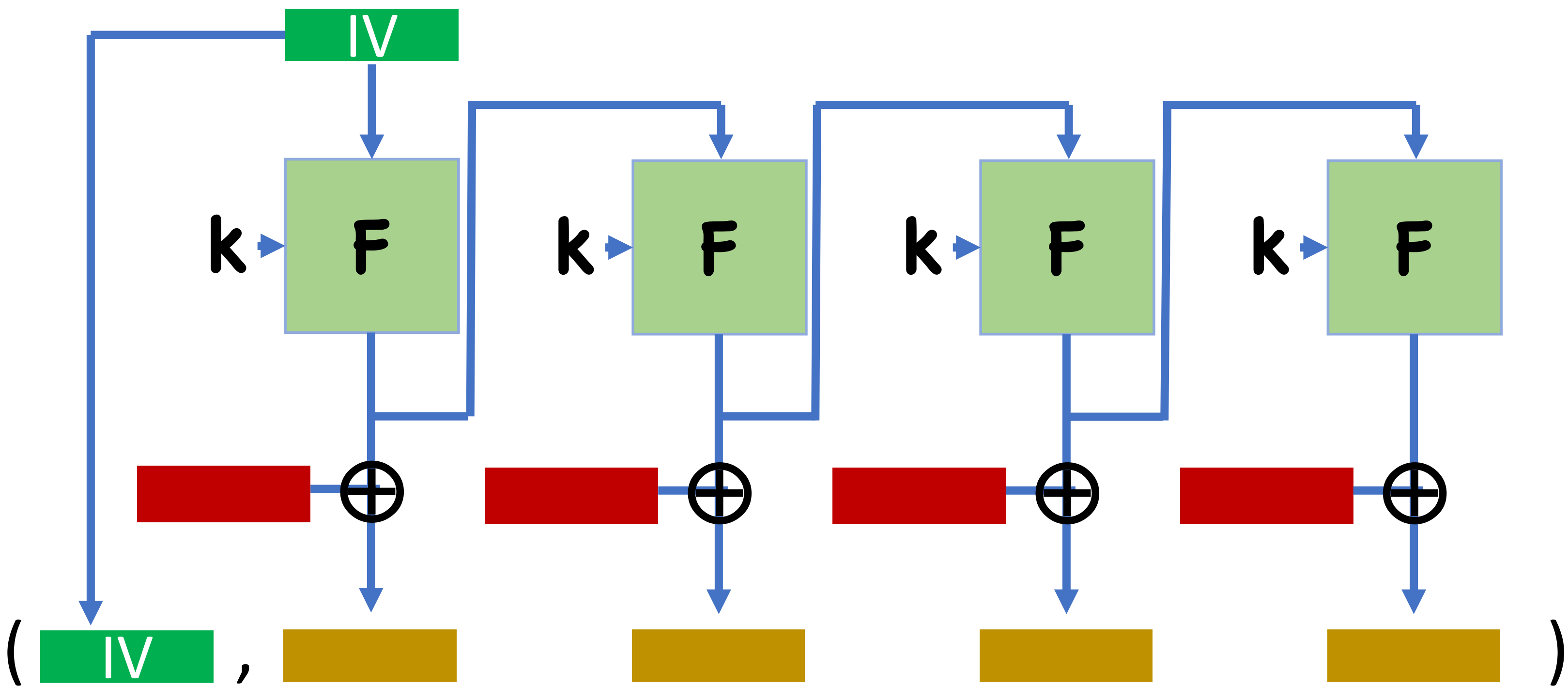
In general, if you can predict the **IV** of the next message, you can break CBC-mode encryption

Idea:

- Set first block of next message to be the next **IV**
- Then **F** will be applied to **0**
- First block of ciphertext will be **F(k,0)**

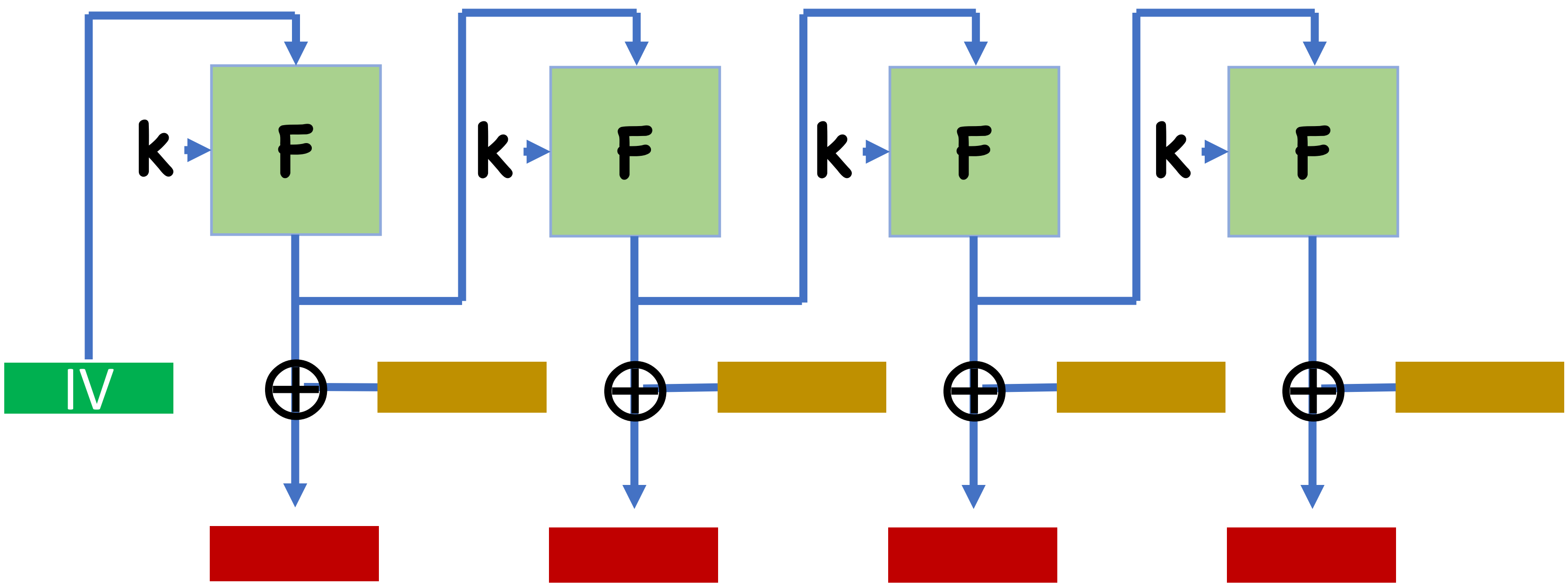
So if we set left messages in this way, all first blocks will be the same

Output Feedback Mode (OFB)



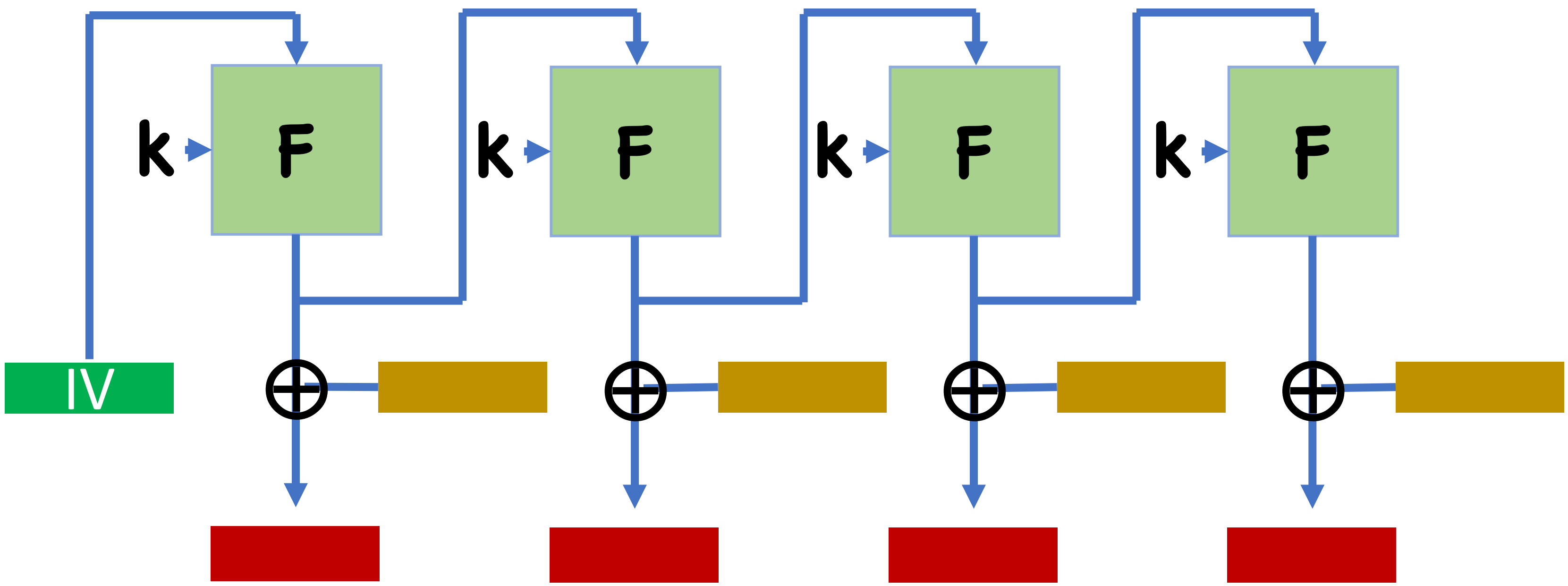
Turn block cipher into stream cipher

OFB Decryption



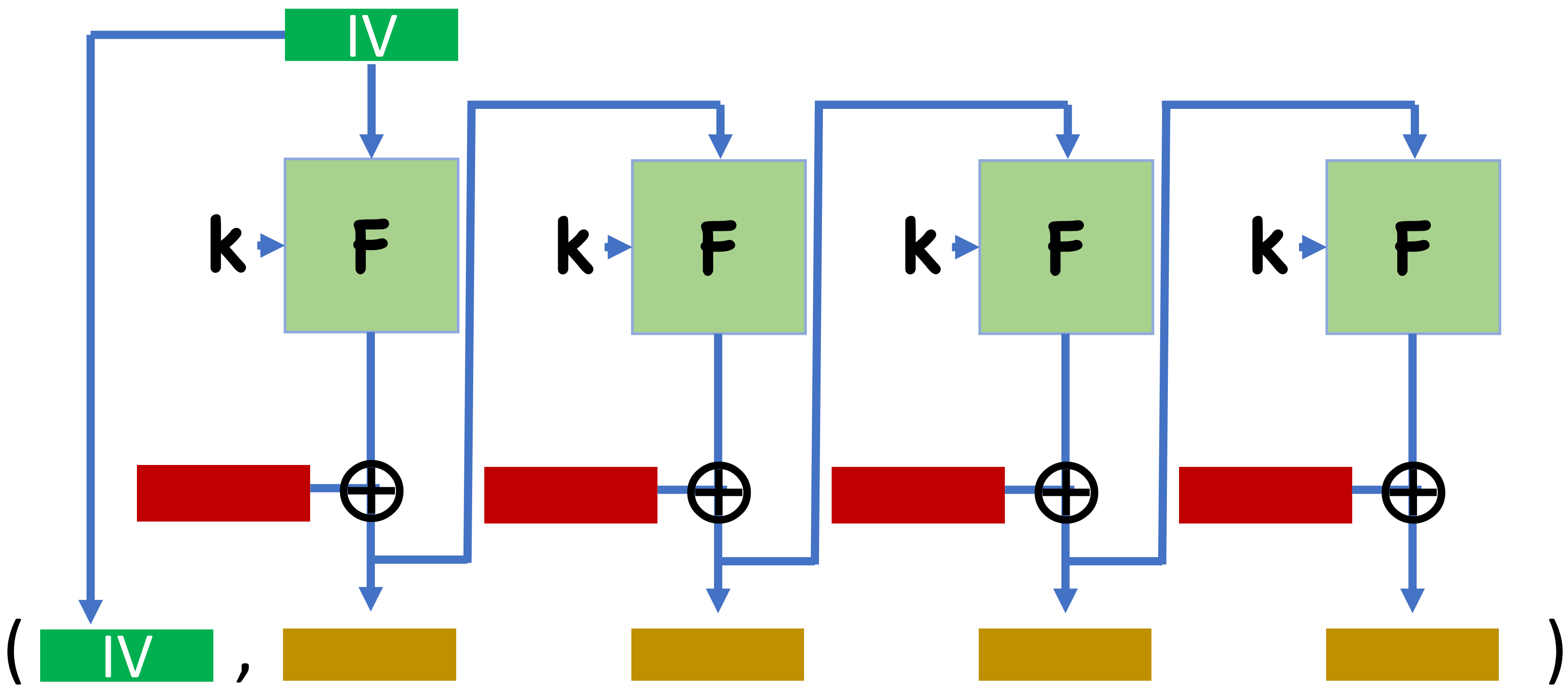
What happens if a block is lost in transmission?

OFB decryption:



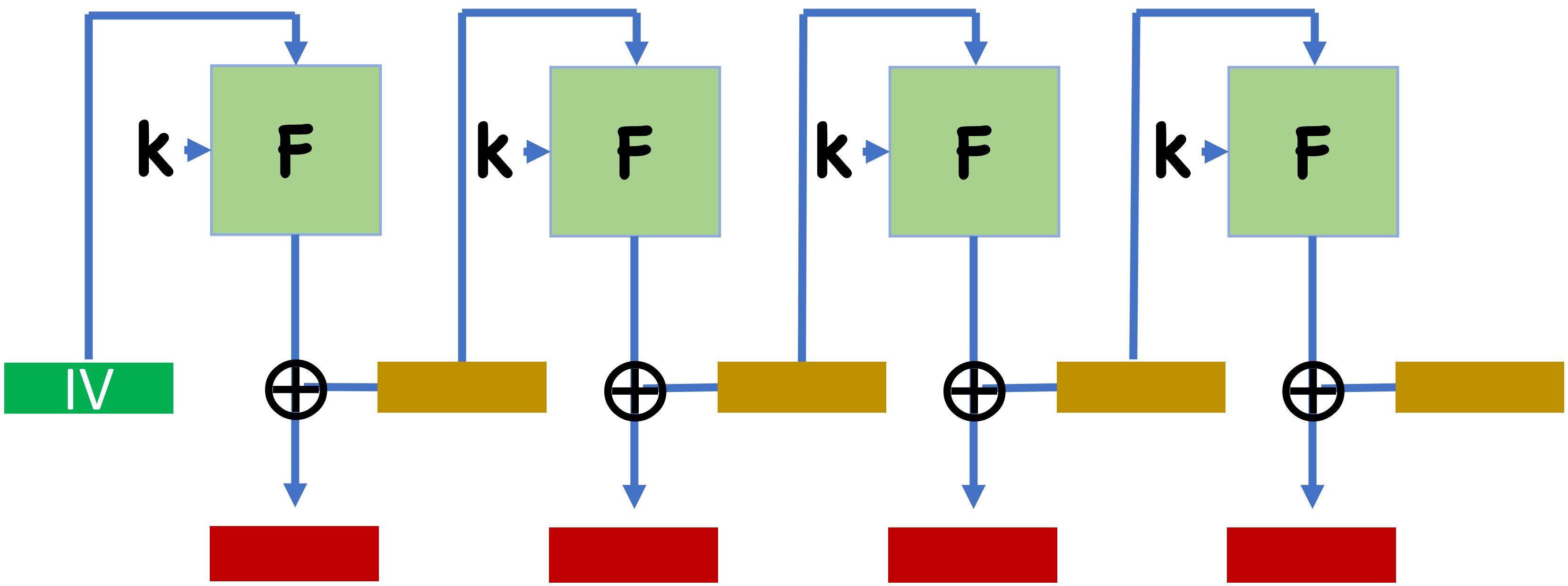
Same goes for CTR mode

Cipher Feedback (CFB)



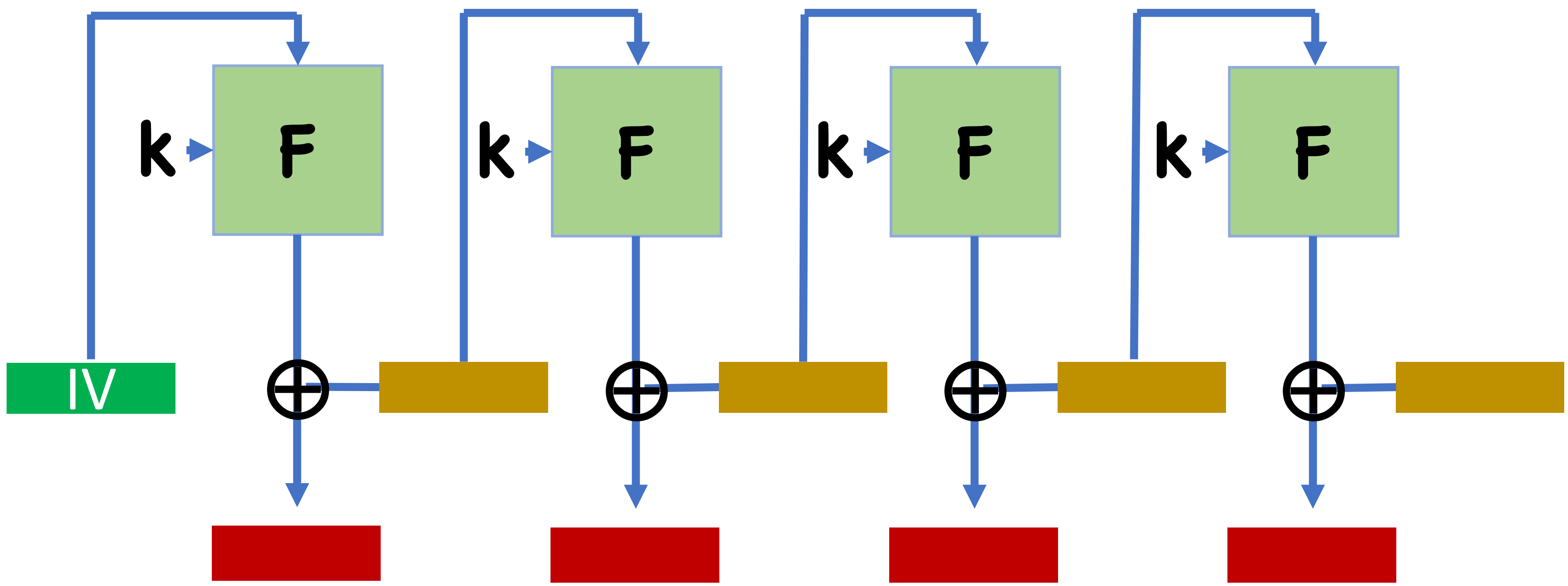
Turn block cipher into **self-synchronizing** stream cipher

CFB Decryption



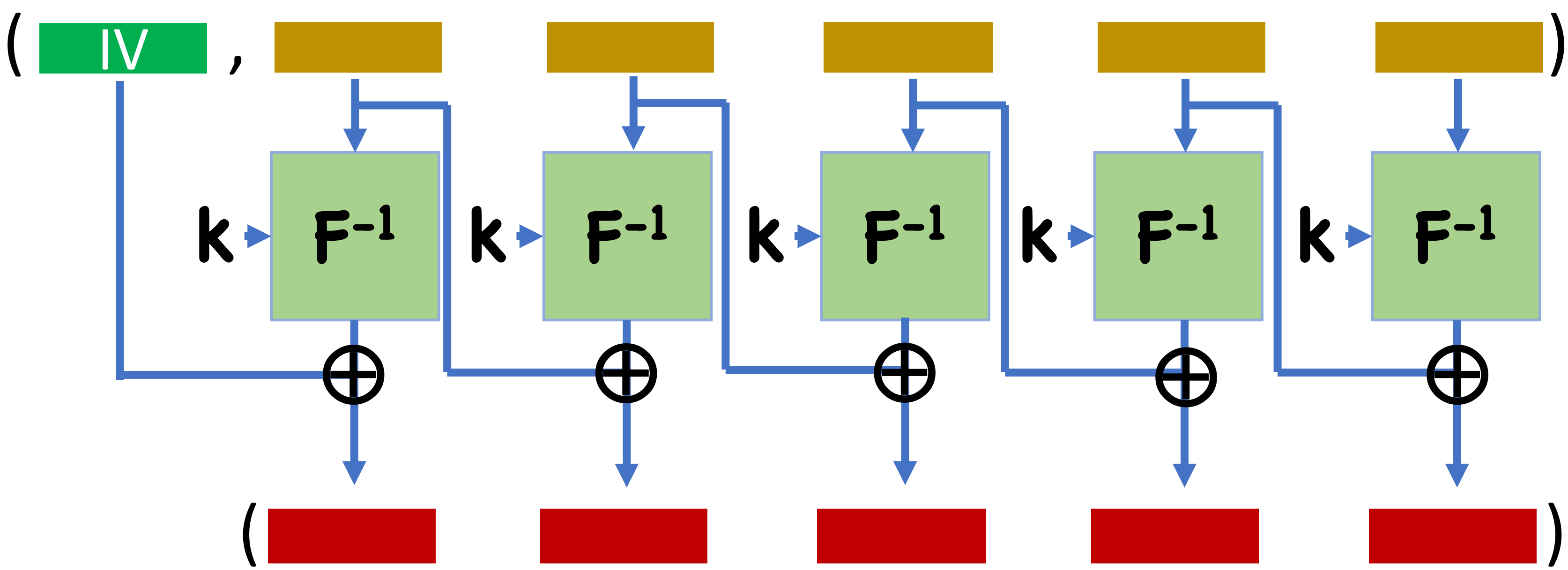
What happens if a block is lost in transmission?

CFB decryption:



What happens if a block is lost in transmission?

What about CBC?



Security of OFB, CFB modes

Security very similar to CBC

Define 4 hybrids

- 0: encrypt left messages
- 1: replace PRP with random permutation
- 2: encrypt right messages
- 3: replace random permutation with PRP

0,1 and 2,3 are indistinguishable by PRP security

1,2 are indistinguishable since ciphertexts are essentially random

Summary

PRPs/Block Ciphers

Modes of operations: ECB, Counter, CBC, OFB, CFB

Next Time

Designing PRPs/PRFs

Reminders

My OH today are delayed until 5pm

- Resume normal schedule next week

HW2 due tomorrow

Project 1 due next week