COS433/Math 473: Cryptography

Mark Zhandry

Princeton University

Spring 2020

Announcements

OH decided:

- Ben: Mondays 3pm
- Jiaxin: Wednesdays 1pm
- Me: Fridays 10am

Normal OH will start **NEXT WEEK**

This week only:

I will have OH 10am on Wednesday 2/12

Announcements

HW1 posted on course website

- Due Feb 20, 11:59pm
- Submission instructions TBA

Previously on COS 433…

Takeaway: Crypto is Hard

Designing crypto is hard, even experts get it wrong

- Just because I don’t know how to break it doesn’t mean someone else can’t

Unexpected attack vectors

- Known/chosen plaintext attack
- Chosen *ciphertext* attack
- Timing attack
- Power analysis
- Acoustic cryptanalysis

Takeaway: Need for Formalism

For most of history, cipher design and usage based largely on intuition

- Intuition in many cases false

Instead, need to formally define the usage scenario

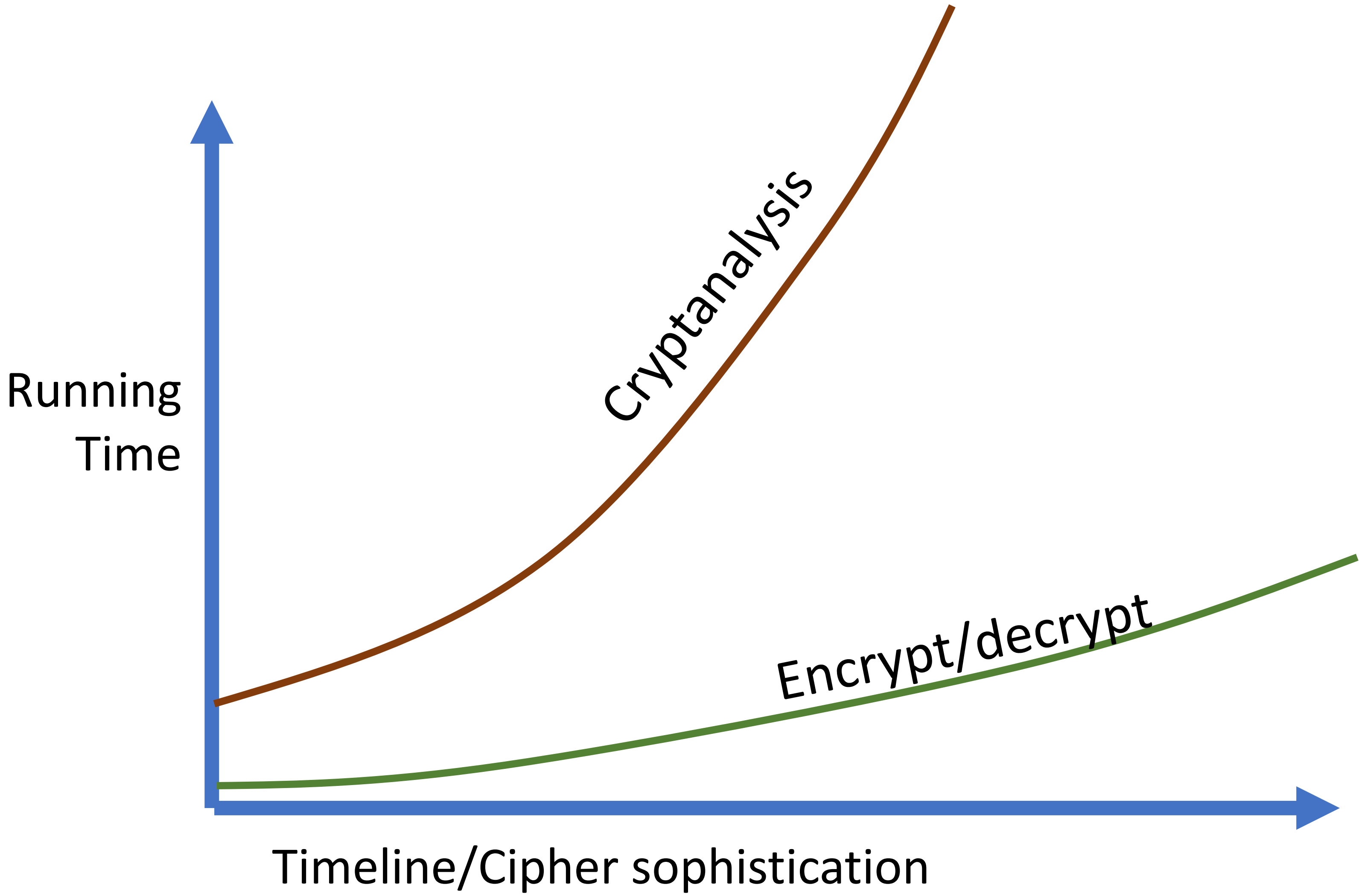
- Prove that scheme is secure in scenario
- Only use scheme in that scenario

Takeaway: Kerckhoffs's Principle

Kerckhoffs's Principle: A cryptosystem should be secure even if everything about the system, except the key, is public knowledge.

- Leaks happen. Should only have to update key, not redesign entire system
  - Even worse, cipher can potentially be reconstructed from ciphertexts
- More eyes means more likely to be secure
- Necessary for formalizing crypto

Takeaway: Importance of Computers



Modern Cryptography

Basics of Defining Crypto

Usually three pieces:

1. **Syntax:** what algorithms are there, what are the inputs/outputs
2. **Correctness/completeness:** how do the algorithms interact
3. **Security:** what should an adversary be permitted/prevented from doing

Formalizing Encryption

Syntax:

- Key space $\mathbf{K}$
- Message space $\mathbf{M}$
- Ciphertext space $\mathbf{C}$
- $\mathbf{Enc: K \times M \rightarrow C}$
- $\mathbf{Dec: K \times C \rightarrow M}$

Correctness:

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

Example: One-Time Pad

K? $\{0,1\}^n$
M? $\{0,1\}^n$
C? $\{0,1\}^n$

$\mathbf{Enc(k,m) = m \oplus k}$

$\mathbf{Dec(k,c) = c \oplus k}$

Correctness: $\mathbf{m' = c \oplus k = (m \oplus k) \oplus k = m}$

(Perfect) Semantic Security

Definition: A scheme **(Enc,Dec)** is **(perfectly) semantically secure** if, for all:

- Distributions **D** on **M** ← Plaintext distribution
- Functions $\mathbf{I:M \rightarrow \{0,1\}^*}$ ← Info adv gets
- Functions $\mathbf{f:M \rightarrow \{0,1\}^*}$ ← Info adv tries to learn
- Functions $\mathbf{A:C \times \{0,1\}^* \rightarrow \{0,1\}^*}$ ← Adversary

"Simulator"

There exists a function $\mathbf{S:\{0,1\}^* \rightarrow \{0,1\}^*}$ such that

$$\Pr[\ A(\ Enc(k,m)\ ,\ I(m)\ ) = f(m)\ ]$$
$$= \Pr[\ S(\ I(m)\ ) = f(m)\ ]$$

where probabilities are taken over $\mathbf{k \leftarrow K,\ m \leftarrow D}$

Perfect Secrecy [Shannon'49]

Definition: A scheme **(Enc,Dec)** has **perfect secrecy** if, for any two messages $\mathbf{m_0}, \mathbf{m_1} \in \mathbf{M}$

$$\mathbf{Enc(K,\ m_0)} \overset{d}{=} \underbrace{\mathbf{Enc(K,\ m_1)}}$$

Random variable corresponding to uniform distribution over **K**

Random variable corresponding to encrypting $\mathbf{m_1}$ using a uniformly random key

Semantic Security = Perfect Secrecy

Theorem: A scheme **(Enc,Dec)** is semantically secure if and only if it has perfect secrecy

Proper Use Case for Perfect Security

- Message can come from any distribution ✓
- Adversary can know anything about message ✓
- Encryption hides anything ✓

- But, definition only says something about an adversary that sees a single message ✗

  $\Longrightarrow$ If two messages, no security guarantee

- Assumes no side-channels ✗
- Assumes key is uniformly random ✗

Today: Weaknesses of Perfect Security

Perfect Security of One-Time Pad

Fix any message $\mathbf{m} \in \{0,1\}^n$, ciphertext $\mathbf{c} \in \{0,1\}^n$

$$\begin{aligned} \Pr_k[\mathrm{Enc}(k,m)=c] &= \Pr_k[k \oplus m = c] \\ &= \Pr_k[k = m \oplus c] \\ &= 2^{-n} \end{aligned}$$

Therefore, for any $\mathbf{m}$, $\mathrm{Enc}(K,\ m)$ $=$ uniform dist over $\mathbf{C}$

In particular, for any $\mathbf{m_0},\mathbf{m_1}$,

$$\mathrm{Enc}(K,\ m_0) \stackrel{d}{=} \mathrm{Enc}(K,\ m_1)$$

Variable Length Messages

Variable-Length Messages

OTP has message-length $\{0,1\}^n$ where n is the key length

In practice, fixing the message size is often unreasonable

So instead, will allow for smaller messages to be encrypted

Variable-Length OTP?

Does the variable length OTP have perfect secrecy according to our definition?

Ciphertext Size

Theorem: For scheme with perfect secrecy, the expected ciphertext size for any message, $\mathbb{E}[\ |\mathrm{Enc}(K,m)|\ ]$, is at least $(\log_2 |M|) - 3$

Proof

Fix a key $\mathbf{k}$.

Let $\mathbf{C_{k,m}}$ be set of ciphertexts $\mathbf{c}$ s.t. $\mathbf{Pr[Enc(k,m)=c]>0}$

By correctness, each $\mathbf{C_{k,m}}$ as $\mathbf{m}$ varies are disjoint and non-empty

- If $\mathbf{c} \in \mathbf{C_{k,m}}$ and $\mathbf{c} \in \mathbf{C_{k,m'}}$, then $\mathbf{m'=Dec(k,c)=m}$

Therefore, therefore $|\cup_{\mathbf{m}} \mathbf{C_{k,m}}| \geq |\mathbf{M}|$

Proof

$$|\cup_m C_{k,m}| \geq |M|$$

Therefore, if we encrypt a random message, the expects size of a ciphertext is at least

$$\Sigma_m \min(|c| : c \in C_{k,m}) / |M|$$

$\min(|c| : c \in C_{k,m}) = t$ for at most 2^t different m

Proof

Let $\mathbf{r} = \text{floor}(\log_2|M|)$

$$\Sigma_m \min(\ |c|\ :\ c \in C_{k,m})\ /\ |M|$$

$$= (1\times0+2\times1+4\times2+\ldots+2^{r-1}\times(r-1)$$

$$+(|M|-(2^r-1))\times r\ )/|M|$$

$$= (2^r(r-2)+2\ +\ (|M|-(2^r-1))\times r\ )\ /\ |M|$$

$$= (r-2(2^r-1)\ +\ |M|\times r)\ /\ |M|$$

$$\geq (0-2|M|\ +\ |M|\times r)/|M|\ =\ r-2$$

Proof

Therefore, for a random message, the expected ciphertext length for any key is at least $\log_2|M|-3$

Must also be true for a random key k

By perfect secrecy, for any messages m_0,m_1

$$\mathbb{E}_K[\ |Enc(K,m_0)|\ ] = \mathbb{E}_K[\ |Enc(K,m_1)|\ ]$$

Therefore,

$$\mathbb{E}_K[\ |Enc(K,m_0)|\ ]$$
$$= \mathbb{E}_{K,M}[\ |Enc(K,M)|\ ] \geq \log_2|M|-3$$

Variable-Length Messages

For perfect secrecy of variable length messages, must have expected ciphertext length for short messages almost as long as longest messages

In practice, very undesirable

- What if I want to either send a **100mb** attachment, or just a message "How are you?"

Therefore, we usually allow message length to be revealed

(Perfect) Semantic Security for Variable Length Messages

Definition: A scheme **(Enc,Dec)** is **(perfectly) semantically secure** if, for all:

- Distributions $\mathbf{D}$ on $\mathbf{M}$
- (Probabilistic) Functions $\mathbf{I:M \rightarrow \{0,1\}^*}$
- (Probabilistic) Functions $\mathbf{f:M \rightarrow \{0,1\}^*}$
- (Probabilistic) Functions $\mathbf{A:C \times \{0,1\}^* \rightarrow \{0,1\}^*}$

There exists (probabilistic) func $\mathbf{S:\{0,1\}^* \rightarrow \{0,1\}^*}$ s.t.

$$\Pr[\ A(\ Enc(k,m)\ ,\ I(m)\ ) = f(m)\ ]$$
$$= \Pr[\ S(\ I(m),\ |m|\ ) = f(m)\ ]$$

where probabilities are taken over $\mathbf{k \leftarrow K,\ m \leftarrow D}$

Perfect Secrecy For Variable Length Messages

Definition: A scheme **(Enc,Dec)** has **perfect secrecy** if, for any two messages $\mathbf{m_0}$, $\mathbf{m_1}$ where $|\mathbf{m_0}| = |\mathbf{m_1}|$,

$$\mathbf{Enc(K,\ m_0)} \overset{d}{=} \mathbf{Enc(K,\ m_1)}$$

Easy to adapt earlier proof to show:

Theorem: A scheme **(Enc,Dec)** is semantically secure if and only if it has perfect secrecy

Variable-Length OTP

Key space $\mathcal{K} = \{0,1\}^n$
Message space $\mathcal{M} = \{0,1\}^{\leq n}$
Ciphertext space $\mathcal{C} = \{0,1\}^{\leq n}$

$$\mathrm{Enc}(k, m) = k_{[\,1,\ |m|\,]} \oplus m$$
$$\mathrm{Dec}(k, c) = k_{[\,1,\ |m|\,]} \oplus c$$

Theorem: Variable-Length OTP has perfect secrecy

Encrypting Multiple Messages

Re-using the OTP

What if we have a **100mb** long key **k**, but encrypt only **1mb**?

Can't use first **1mb** of **k** again, but remaining **99mb** is still usable

However, basic OTP definition does not allow us to re-use the key ever

Syntax for Stateful Encryption

Syntax:

- Key space **K**, Message space **M**, Ciphertext space **C**
- State Space **S**
- **Init: {} → S**
- **Enc: K×M×S → C×S**
- **Dec: K×C×S → M×S**

$\mathbf{State_0 \leftarrow Init()}$
$\mathbf{(c_0, state_1) \leftarrow Enc(k, m_0, state_0)}$
$\mathbf{(c_1, state_2) \leftarrow Enc(k, m_1, state_1)}$
…

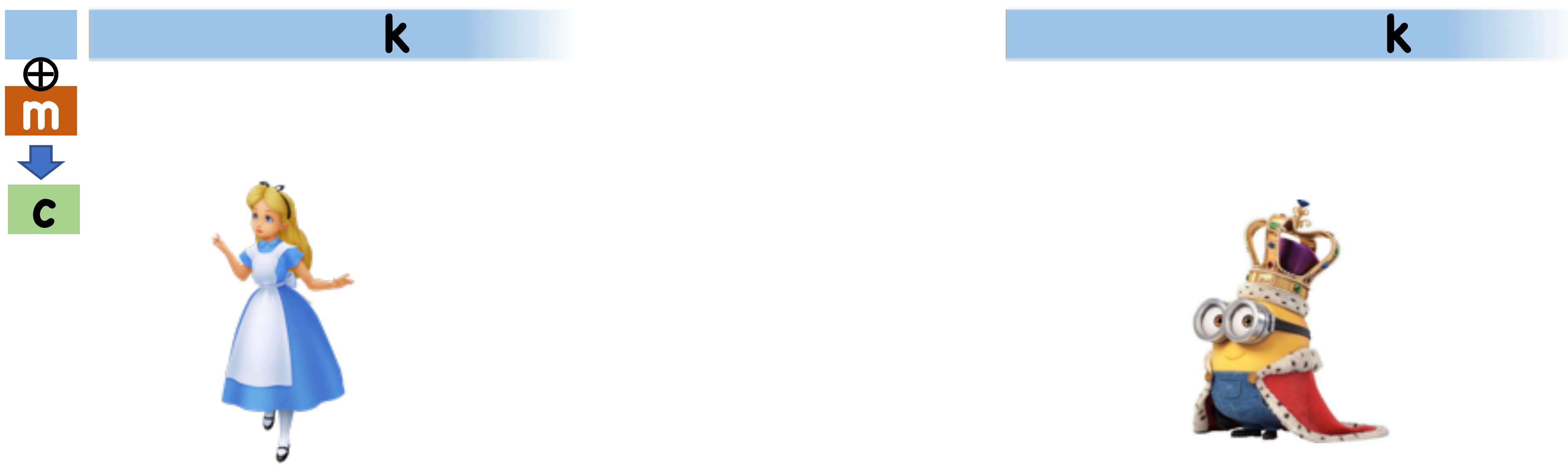
Reusing the OTP

k

k

m

Reusing the OTP



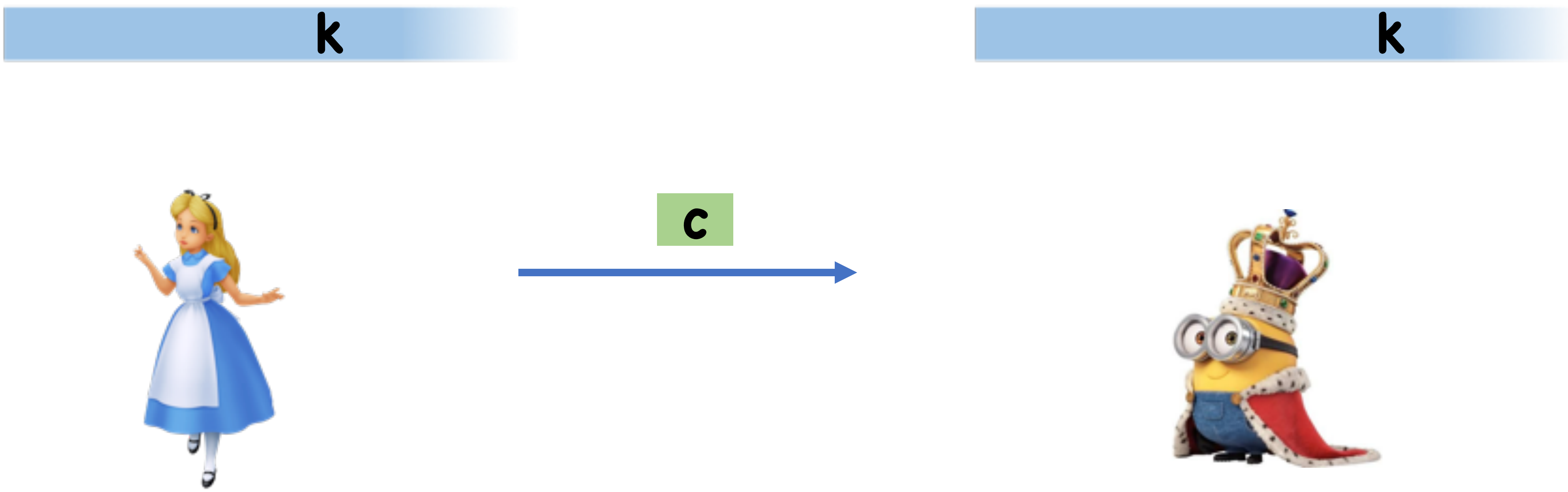
Reusing the OTP

k

k

c

Reusing the OTP



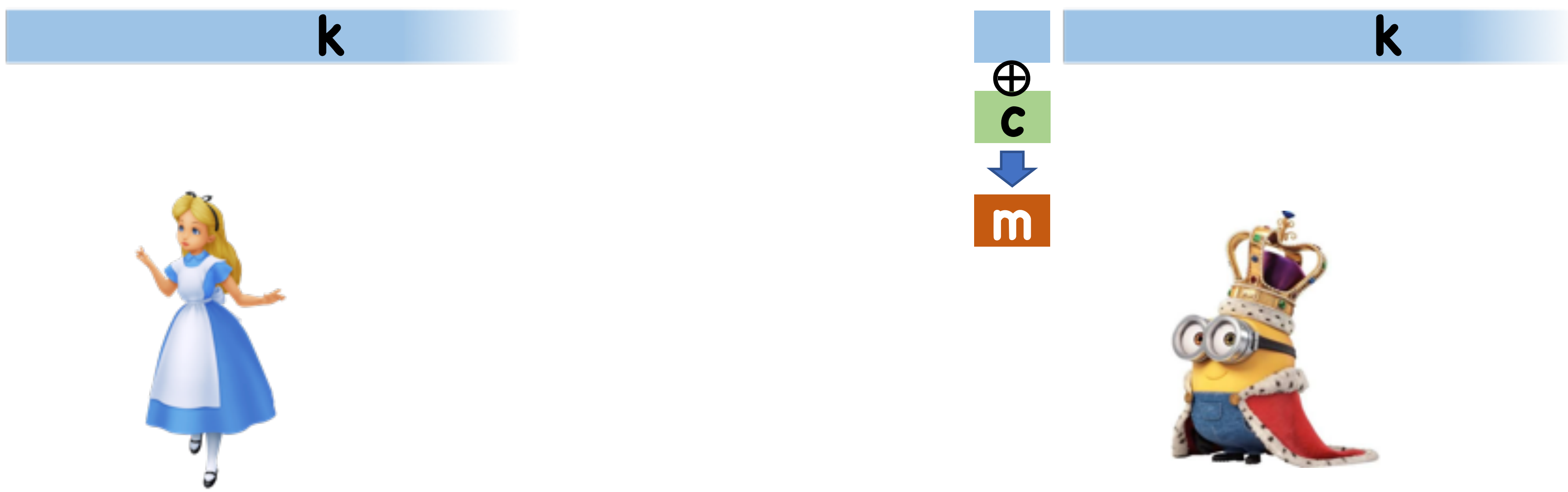
Reusing the OTP

k

k

c

Reusing the OTP



Reusing the OTP

k

k

m

Reusing the OTP

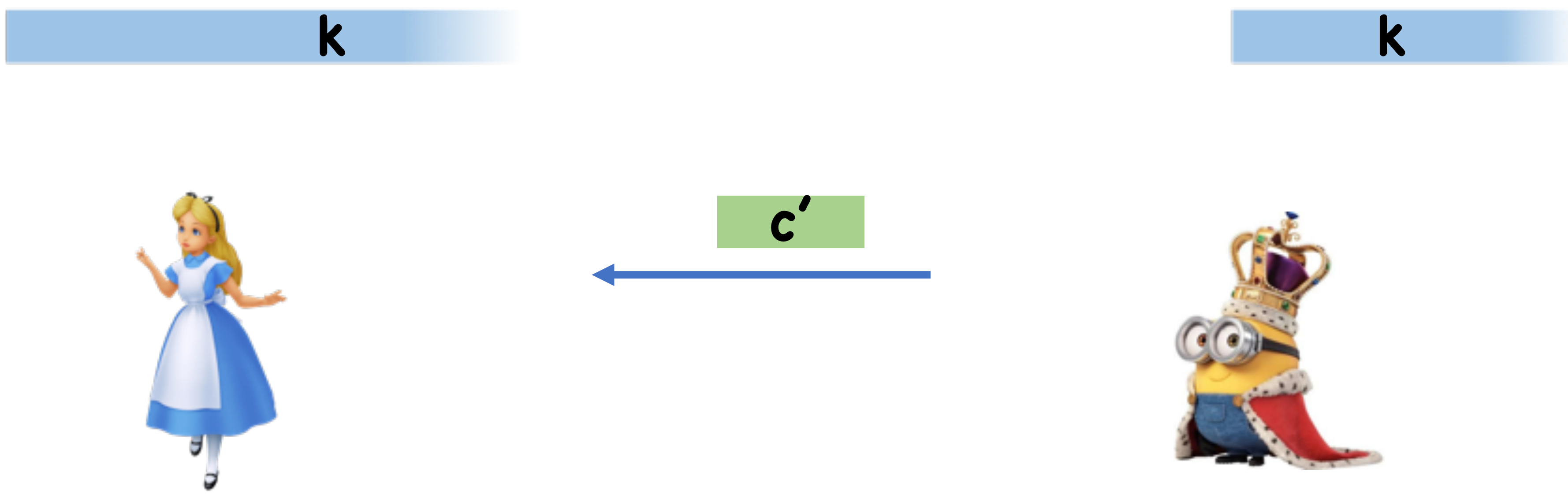
k

k

m′

Reusing the OTP

Reusing the OTP



Reusing the OTP

k

c′

k

Reusing the OTP

Problem

In real world, messages aren't always synchronous

What happens if Alice and Bob try to send message at the same time?

They will both use the same part of the key!

Problem

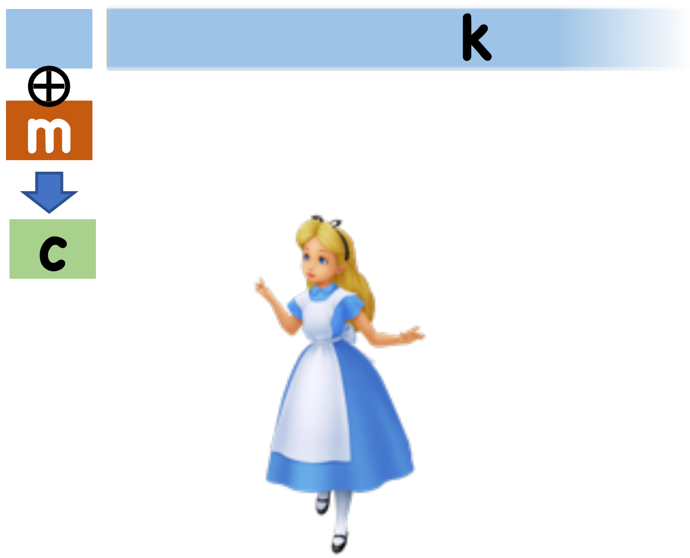
k

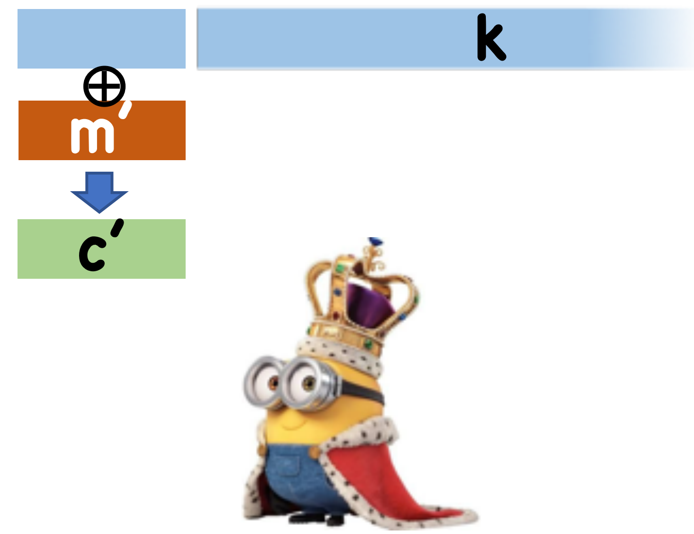
m

k

m'

Problem





Problem

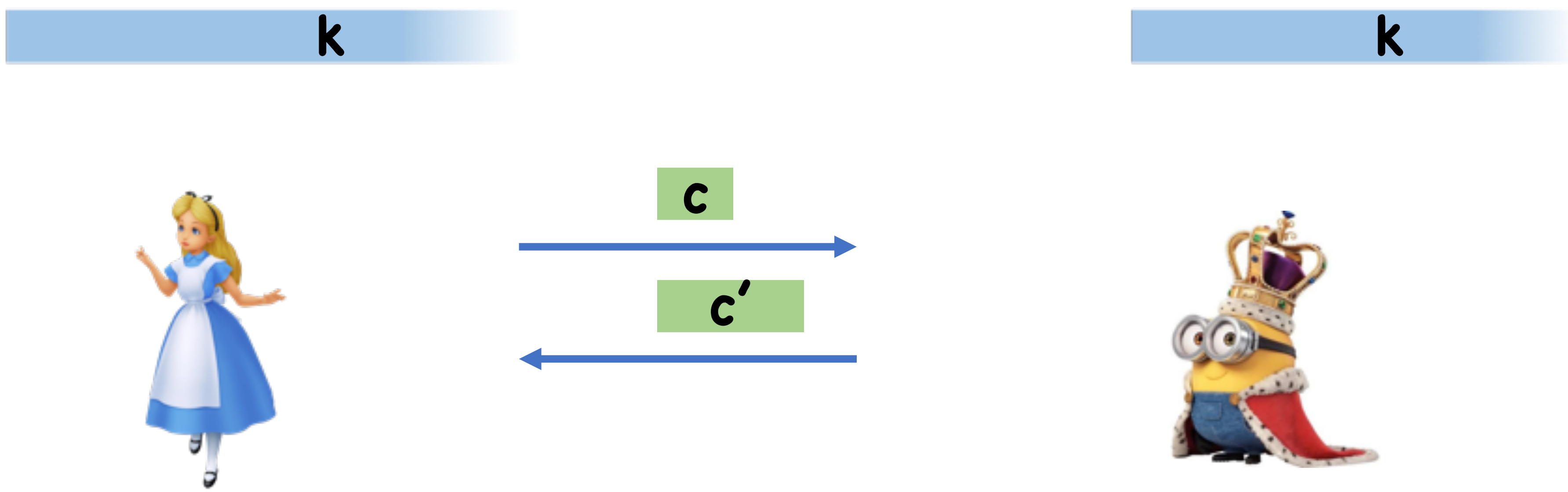
k

c

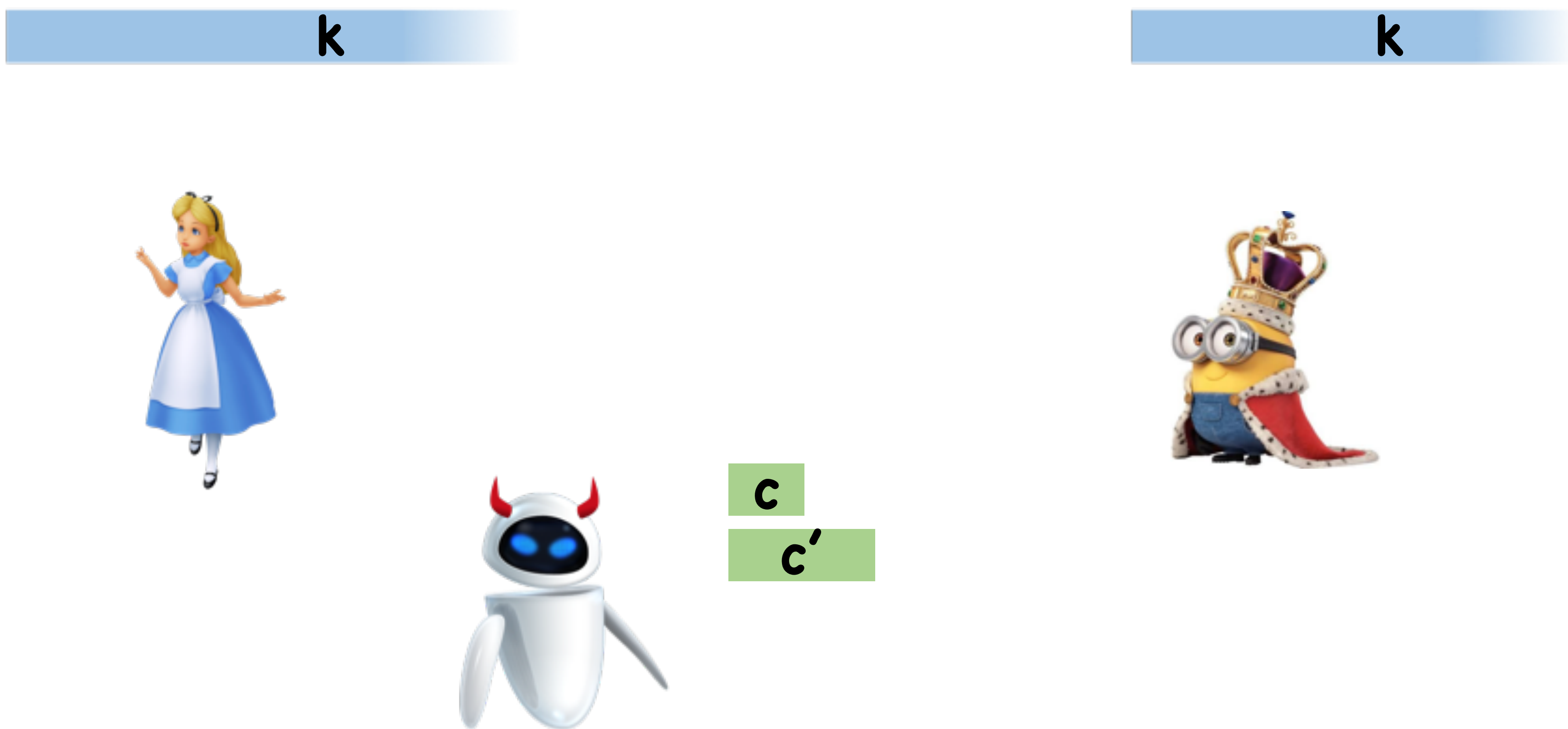
k

c′

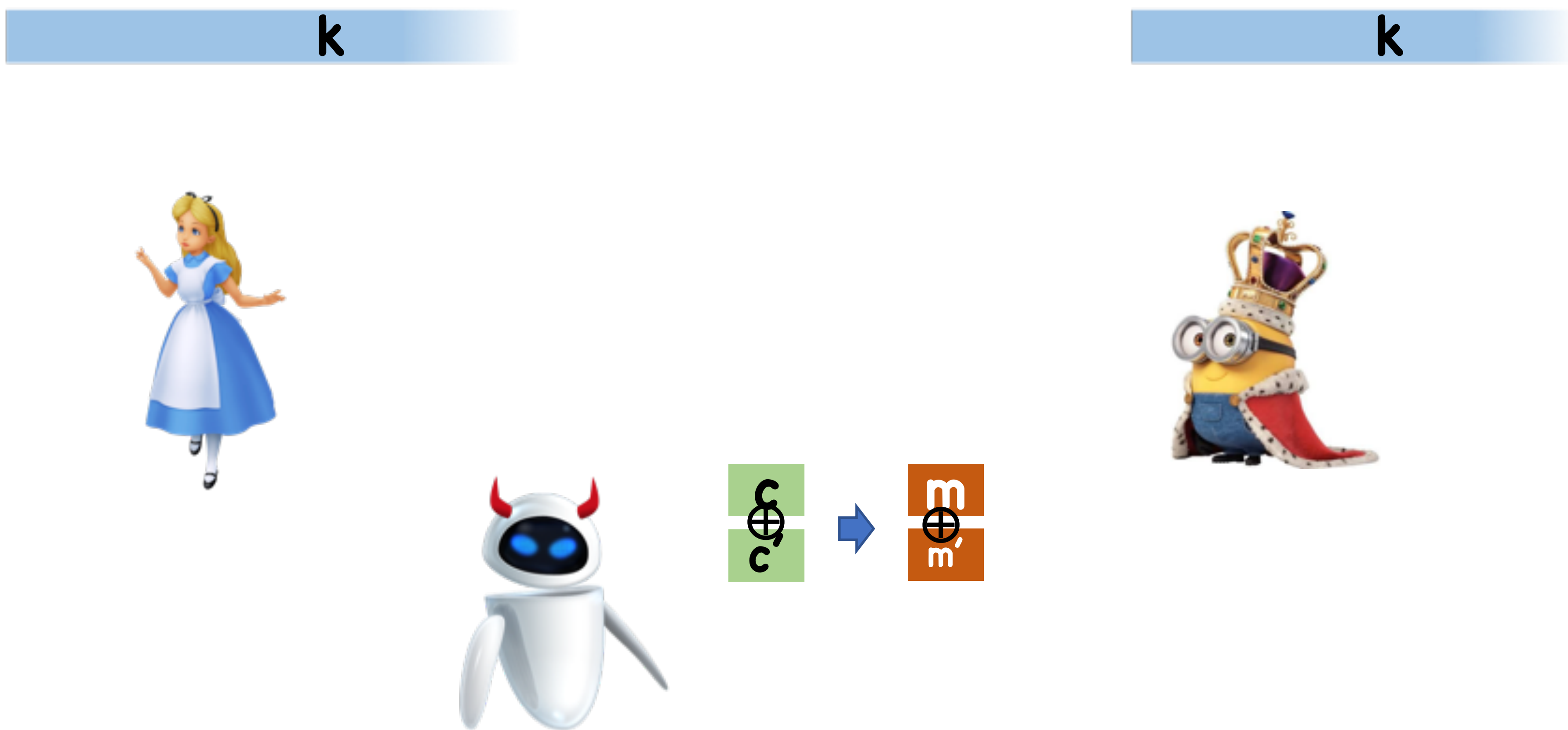
Problem



Problem



Problem



Solution?

Reusing the OTP

$k_{A \to B}$

$k_{B \to A}$

$k_{A \to B}$

$k_{B \to A}$

Still A Problem

In real world, messages aren’t always synchronous

Also, sometimes messages arrive out of order or get dropped

- Need to be very careful to make sure decryption succeeds

These difficulties exist in any stateful encryption

- For this course, we will generally consider only **stateless** encryption schemes

Perfect Security for Multiple Messages?

Stateless Encryption with Multiple Messages

Ex:

M = C
K = Perms(M) (never mind that key is enormous)
Enc(K, m) = K(m)
Dec(K, c) = K^{-1}**(c)**

Q: Is this perfectly secure for two messages?

Theorem: No stateless deterministic encryption scheme can have perfect security for multiple messages

Randomized Encryption

Syntax:

- Key space $\mathbf{K}$
- Message space $\mathbf{M}$
- Ciphertext space $\mathbf{C}$
- $\mathbf{Enc:\ K \times M \rightarrow C}$, potentially probabilistic
- $\mathbf{Dec:\ K \times C \rightarrow M}$ (usually deterministic)

Correctness:

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

Randomized Encryption

Syntax:

- Key space **K**
- Message space **M**
- Ciphertext space **C**
- **Enc: K×M → C**, potentially probabilistic
- **Dec: K×C → M** (usually deterministic)

Correctness:

- For all $\mathbf{k} \in \mathbf{K}$, $\mathbf{m} \in \mathbf{M}$,

$$\Pr[\ \mathrm{Dec}(k, \mathrm{Enc}(k,m)\ ) = m\ ] \quad = \quad 1$$

Stateless Encryption with Multiple Messages

Ex:

$r \leftarrow R$

$C = M \times R$

$K = \text{Perms}(C)$

$\text{Enc}(K, m) = K(m, r)$

$\text{Dec}(K, c) = (m', r') \leftarrow K^{-1}(c)$, output m'

Q: Is this perfectly secure for two messages?

Proof of Easy Case

Let **(Enc,Dec)** be stateless, deterministic

Let $m_0^{(0)} = m_0^{(1)}$
Let $m_1^{(0)} \neq m_1^{(1)}$

Consider distributions of encryptions:

- $(c^{(0)} , c^{(1)}) = (\mathrm{Enc}(K, m_0^{(0)}), \mathrm{Enc}(K, m_0^{(1)}))$

  $\Rightarrow c^{(0)} = c^{(1)}$ (by determinism)

- $(c^{(0)} , c^{(1)}) = (\mathrm{Enc}(K, m_1^{(0)}), \mathrm{Enc}(K, m_1^{(1)}))$

  $\Rightarrow c^{(0)} \neq c^{(1)}$ (by correctness)

Generalize to Randomized Encryption

Let **(Enc,Dec)** be stateless, ~~deterministic~~

Let $\mathbf{m_0^{(0)} = m_0^{(1)}}$
Let $\mathbf{m_1^{(0)} \neq m_1^{(1)}}$

Consider distributions of encryptions:

- $\mathbf{(c^{(0)} , c^{(1)}) = (Enc(K, m_0^{(0)}), Enc(K, m_0^{(1)}))}$
  $\Rightarrow$ **????**
- $\mathbf{(c^{(0)} , c^{(1)}) = (Enc(K, m_1^{(0)}), Enc(K, m_1^{(1)}))}$
  $\Rightarrow \mathbf{c^{(0)} \neq c^{(1)}}$ (by correctness)

Generalize to Randomized Encryption

$(c^{(0)} , c^{(1)}) = (\mathrm{Enc}(K, m), \mathrm{Enc}(K, m))$

$\Pr[c^{(0)} = c^{(1)}]$?

- Fix **k**
- Conditioned on k, $c^{(0)}$, $c^{(1)}$ are two independent samples from same distribution $\mathrm{Enc}(k, m)$

> **Lemma:** Given any distribution **D** over a finite set **X**, $\Pr[Y=Y' : Y \leftarrow D, Y' \leftarrow D] \geq 1/|X|$

- Therefore, $\Pr[c^{(0)} = c^{(1)}]$ is non-zero

Generalize to Randomized Encryption

Let **(Enc,Dec)** be stateless, deterministic

Let $m_0^{(0)} = m_0^{(1)}$
Let $m_1^{(0)} \neq m_1^{(1)}$

Consider distributions of encryptions:

- $(c^{(0)} , c^{(1)}) = (\mathrm{Enc}(K, m_0^{(0)}), \mathrm{Enc}(K, m_0^{(1)}))$
  $\Rightarrow \Pr[c^{(0)} = c^{(1)}] > 0$
- $(c^{(0)} , c^{(1)}) = (\mathrm{Enc}(K, m_1^{(0)}), \mathrm{Enc}(K, m_1^{(1)}))$
  $\Rightarrow \Pr[c^{(0)} = c^{(1)}] = 0$

What do we do now?

Reminders

Normal OH will start **NEXT WEEK**

This week only:

I will have OH 10am on Wednesday 2/12

HW1 posted on course website

- Due Feb 20, 11:59pm
- Submission instructions TBA