CIS 5560

Cryptography
Lecture 1

Course website:

pratyushmishra.com/classes/cis-5560-s24/

Slides adapted from Dan Boneh and Vinod Vaikuntanathan

Course Staff

Instructor: Pratyush Mishra (me!)

prat@upenn.edu

TAs:

Jack Hourigan (hojack@upenn.edu)

Tushar Mopuri (tmopuri@upenn.edu)

Alireza Shirzad (alrshir@upenn.edu)

Matan Shtepel (matan.shtepel@gmail.com)

Course Format

- **Lecture:** Tues/Thurs 1:45-3:15PM Fagin Hall 118
- **Grading:**
  - Participation: 5%
  - HW: 40%
  - Midterm: 25%
  - Final: 30%
- **Important dates:**
  - Midterm: 03/14/24
  - Final: TBD

Homeworks

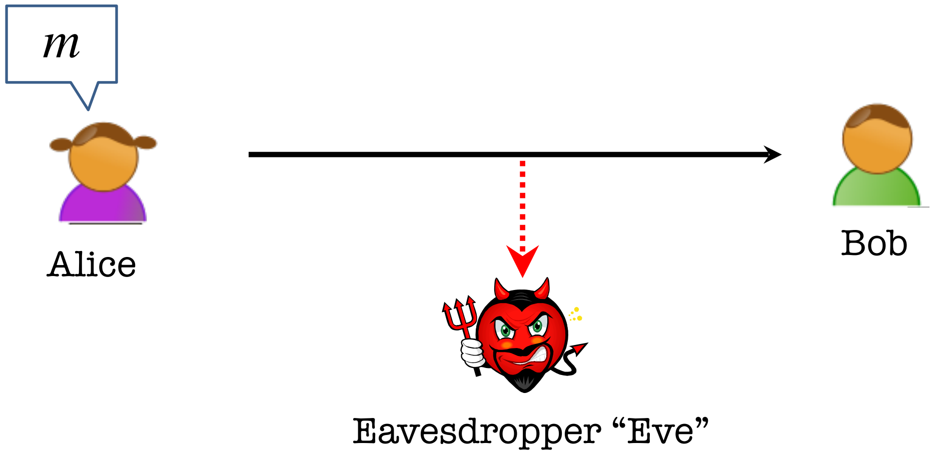
- Usually, 1 per week
- Released on Tuesdays
- Due Monday 5PM
- Drop 2 lowest scores
- Mostly proof-based, with perhaps one programming oriented homework

Important Links

- Class website (WIP): pratyushmishra.com/classes/cis-5560-s24
- EdStem: edstem.org/us/courses/53008
- Canvas: canvas.upenn.edu/courses/1771710/
- Gradescope: gradescope.com/courses/704354

What is Cryptography?

Confidential Communication

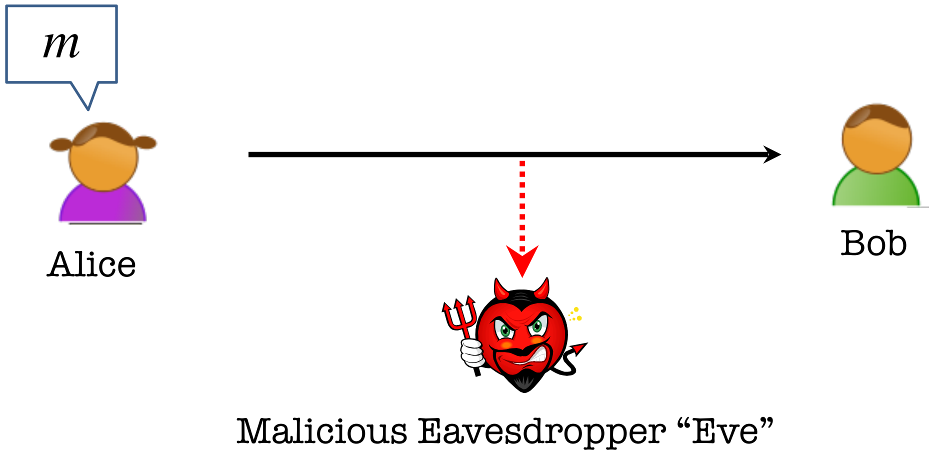


Alice wants to send a message m to Bob without revealing it to Eve.

Tool: Encryption schemes

Eg: Caesar Cipher (broken!!), AES, DES, RSA, etc

Confidential Communication with *Integrity*

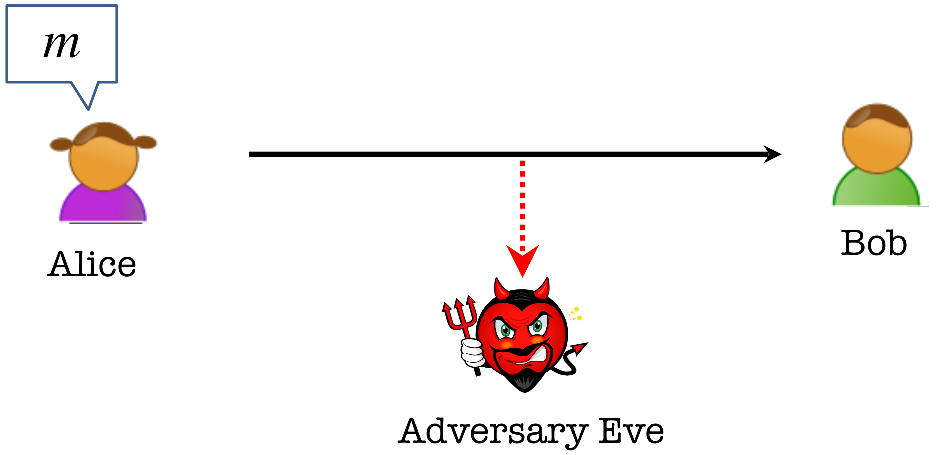


Eve can tamper with messages now

Alice wants to send a message m to Bob without Eve changing it.

Tool: Message Authentication Codes

Communication with *Authenticity*

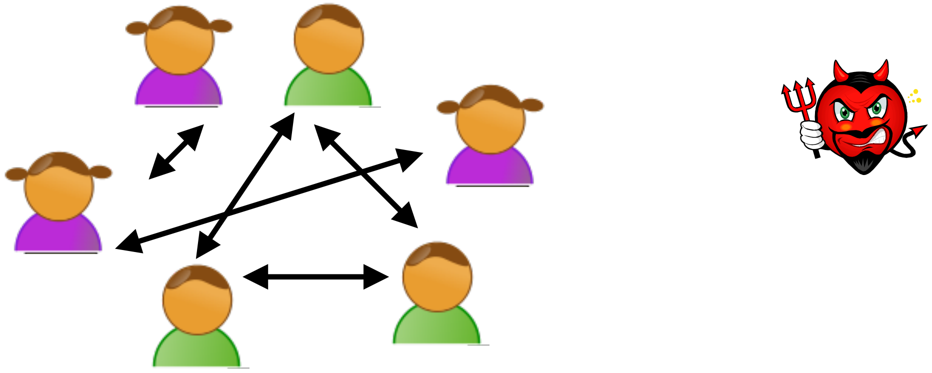


Eve can tamper with messages now

Bob wants guarantee that *only* Alice sent m.

Tool: Digital signatures

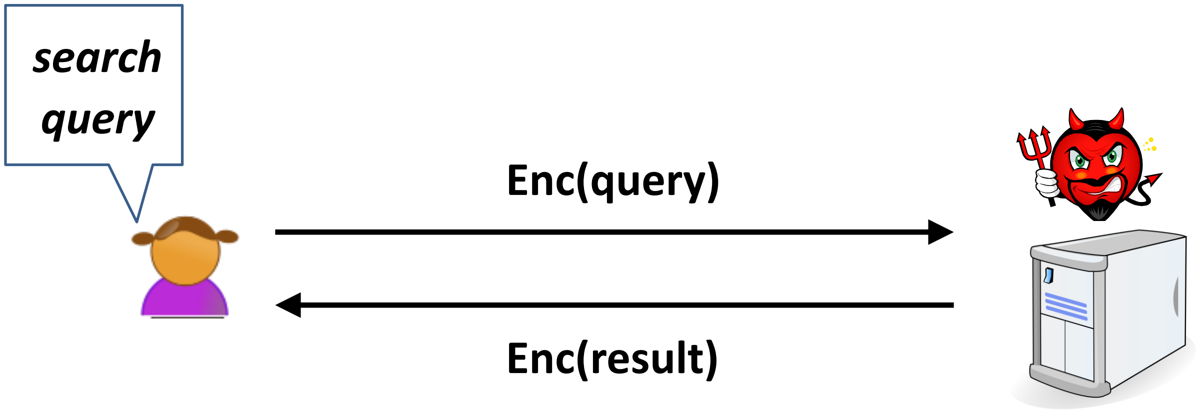
Anonymous Communication



Eve should not be able to tell who is talking to whom

Tool: dining cryptographer networks, onion encryption, etc

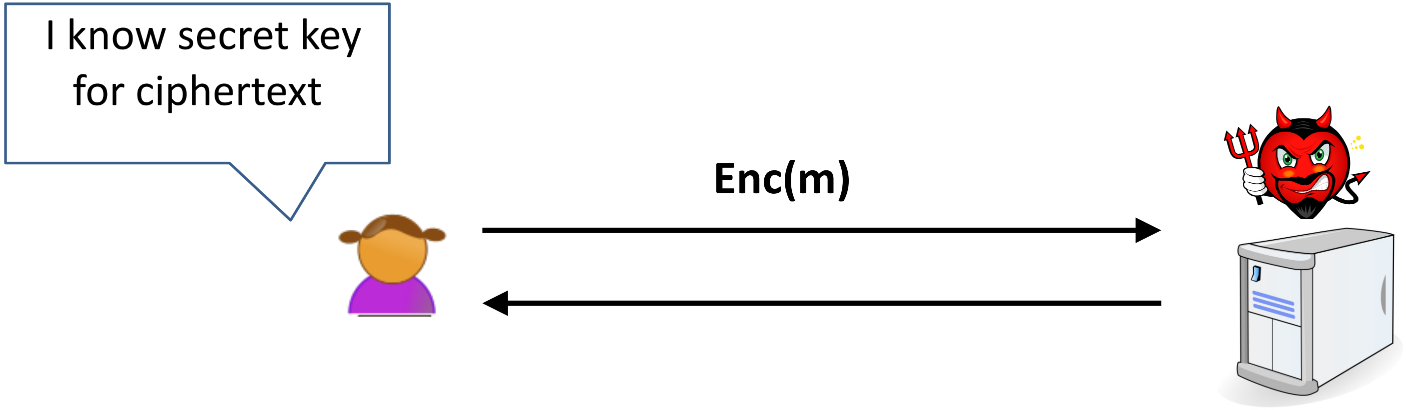
Computation on Secret Data



Eve's server should run computation without learning Alice's data

Tool: Homomorphic encryption, multiparty computation

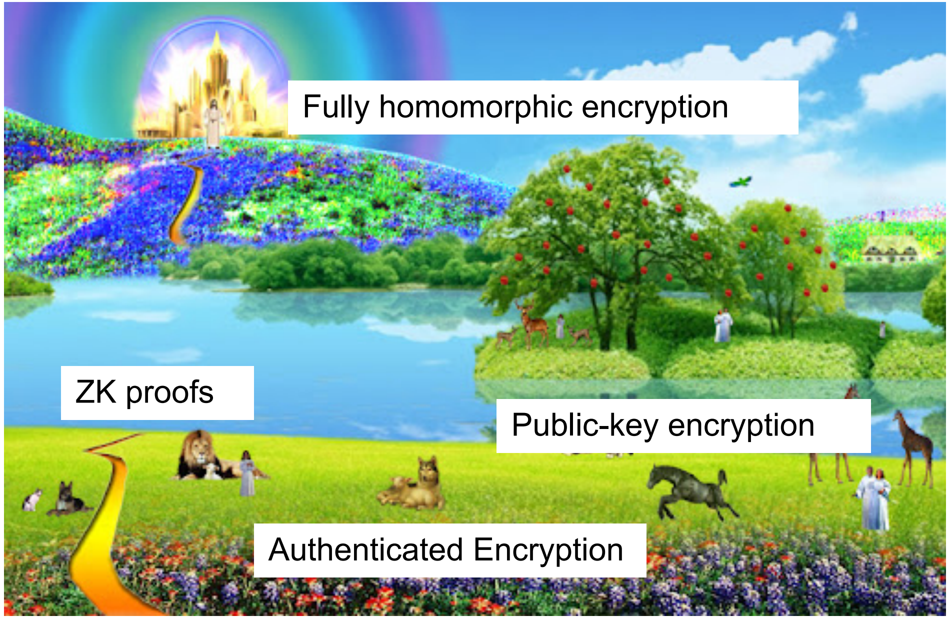
Proofs about Secret Data



Eve's server should be convinced about Alice's claim without learning Alice's secrets.

Tool: Zero knowledge proofs

Crypto is a magical land!



How do we get there? Not magic, but science!

The three steps in cryptography:

- Precisely specify threat model
- Propose a construction
- Prove that breaking construction under threat model will solve an underlying hard problem

Things to remember

Cryptography is:

- A tremendous tool
- The basis for many security mechanisms

Cryptography is not:

- The solution to all security problems
- Reliable unless implemented and used properly
- Something you should try to invent yourself
    - many many examples of broken ad-hoc designs

Discrete Probability Primer

- **Probability distribution** P over a finite set S is a function $P : S \to [0,1]$ such that $\sum_{x \in S} P(x) = 1$
- **Support** of P is set $\mathrm{Supp}(P) \subseteq S$ s.t. $\forall x \in \mathrm{Supp}(P), P(x) \neq 0$
- **An event** is a set $A \subseteq S$; $\Pr[A] = \sum_{x \in A} P(x) \in [0,1]$
- **Union bound:** For events A_1 and A_2, $\Pr[A_1 \cup A_2] \leq \Pr[A_1] + \Pr[A_2]$
- A **random variable** X is a fn $X : S \to V$ that induces a dist. on V
- Events A and B are **independent** if $\Pr[A \text{ and } B] = \Pr[A] \cdot \Pr[B]$
- RVs X and Y are **ind.** if $\Pr[X = a \text{ and } Y = b] = \Pr[X = a] \cdot \Pr[Y = b]$

- $S = \{0,1\}^2$
- **Example distribution:** Uniform: for all $x \in S, P(x) = 1/|S|$
- **Example event:** $A = \{x \in S \mid \text{lsb}(x) = 1\}$. $\Pr[A] = 1/2$
- **Example RV:** $X = \text{lsb}$. Here $V = \{0,1\}$, and induced distribution is $\Pr[X = 0] = 1/2 \ ; \ \Pr[X = 1] = 1/2$
- **Example independent RVs:** $X = \text{lsb}$ and $Y = \text{msb}$
$\Pr[X(x) = 0 \text{ and } Y(x) = 0] = \Pr[x = 00] = \frac{1}{4} = Pr[X(x) = 0] \Pr[Y(x) = 0]$

Uniform RV

- A **Uniform RV** is $R : S \to S$ that induces a uniform dist on S.
- That is, for all $x \in S$, $\Pr[R = x] = 1/|S|$

Randomized algorithms

- Deterministic algorithm: $y \leftarrow A(m)$
- Randomized algorithm: $y \leftarrow A(m; R)$ where $R \xleftarrow{\$} \{0,1\}^n$
  - Output is a random variable $y \xleftarrow{\$} A(m)$

An important property of XOR

Thm: Y is an RV over $\{0,1\}^n$, X is a uniform ind. RV over $\{0,1\}^n$

Then $Z := Y \oplus X$ is uniform var. on $\{0,1\}^n$

Proof: (for n=1)

Pr[Z=0] = $\Pr[(x,y)=(0,0) \text{ or } (x,y)=(1,1)] =$

$= \Pr[(x,y)=(0,0)] + \Pr[(x,y)=(1,1)] =$

$= \frac{p_0}{2} + \frac{p_1}{2} = \frac{1}{2}$

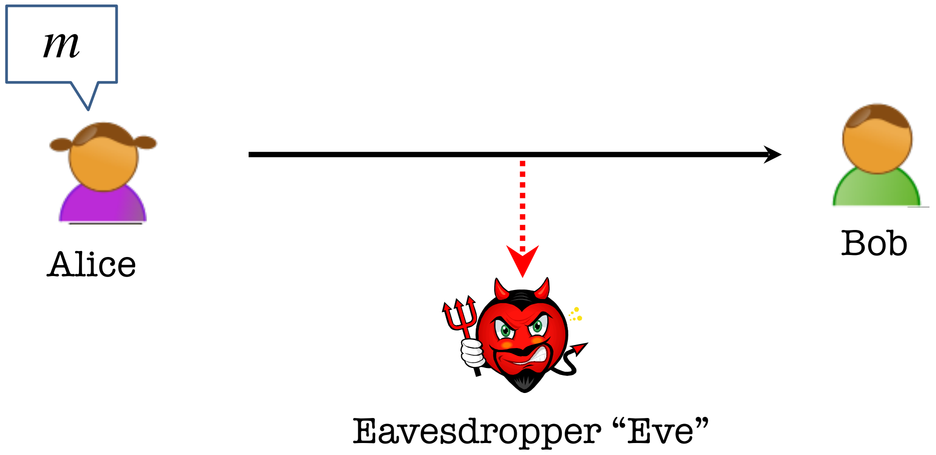
Y	Pr
0	p_0
1	p_1

X	Pr
0	1/2
1	1/2

X	Y	Pr
0	0	$p_0/2$
0	1	$p_1/2$
1	0	$p_0/2$
1	1	$p_1/2$

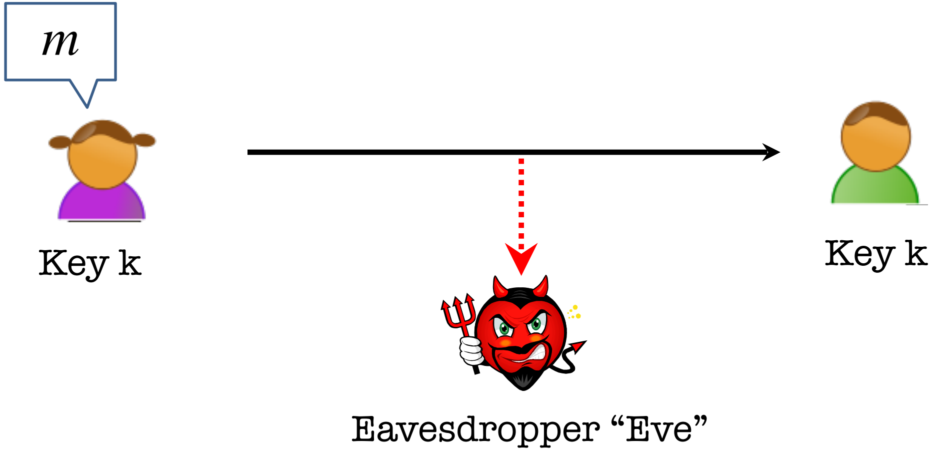
Our First Definition: Symmetric Key Encryption

Secure Communication



Alice wants to send a message m to Bob without revealing it to Eve.

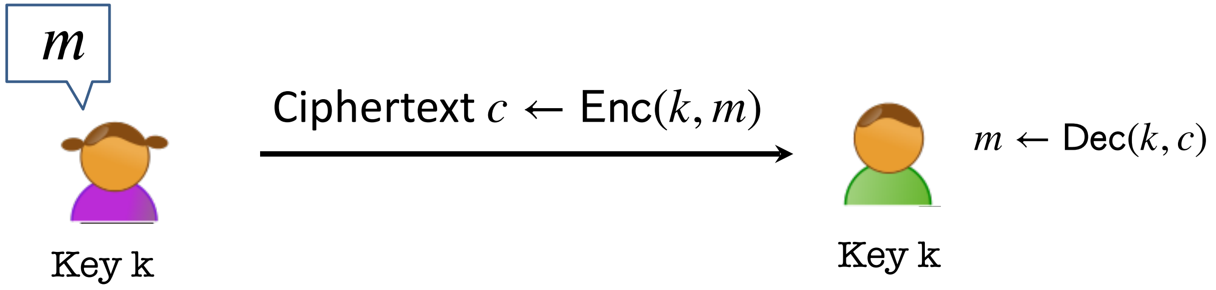
Secure Communication



SETUP: Alice and Bob meet beforehand to agree on a secret key k.

Key Notion: Secret-key Encryption

(or Symmetric-key Encryption)



Three (possibly randomized) polynomial-time algorithms:

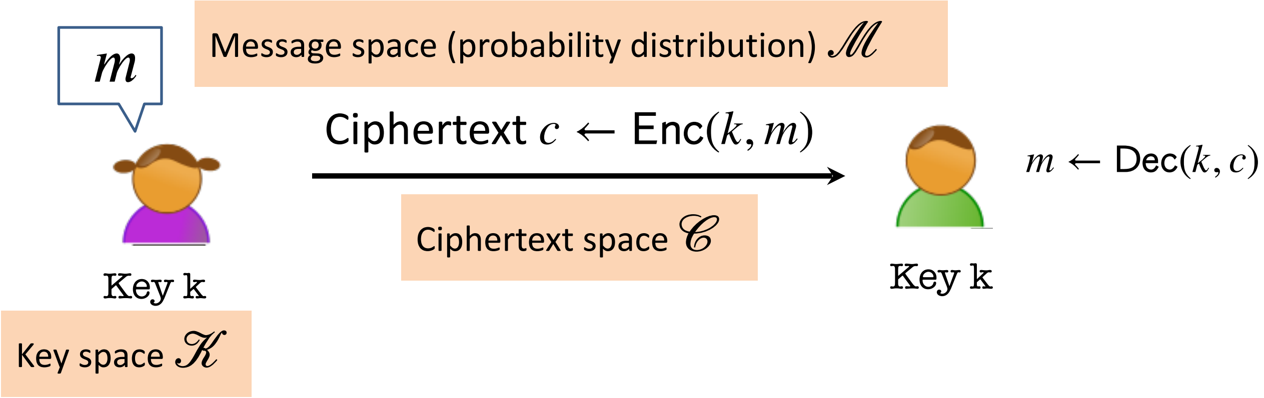
- **Key Generation Algorithm:** $\text{Gen}(1^k) \rightarrow k$

  Has to be randomized (why?)

- **Encryption Algorithm:** $\text{Enc}(k, m) \rightarrow c$

- **Decryption Algorithm:** $\text{Dec}(k, c) \rightarrow m$

Key Property 1: Correctness



- $\forall k \in \mathsf{Supp}(\mathsf{Gen}), \forall m \in \mathscr{M}, \mathsf{Dec}(k, \mathsf{Enc}(k, m)) = m$
- **Most basic property: if Bob gets incorrect answer, scheme is useless!**

The Worst-case Adversary

- An arbitrary computationally *unbounded* algorithm **EVE**.*

- Knows Alice and Bob's algorithms Gen, Enc and Dec but does not know the key nor their internal randomness.

  (*Kerckhoff's principle or Shannon's maxim*)

- Can see the ciphertexts going through the channel
  (but cannot modify them… we will come to that later)

Security Definition: What is she trying to learn?

What is a secure encryption scheme?

Attacker's abilities: **CT only attack** (for now)

Possible security requirements:

attempt #1: **attacker cannot recover secret key**

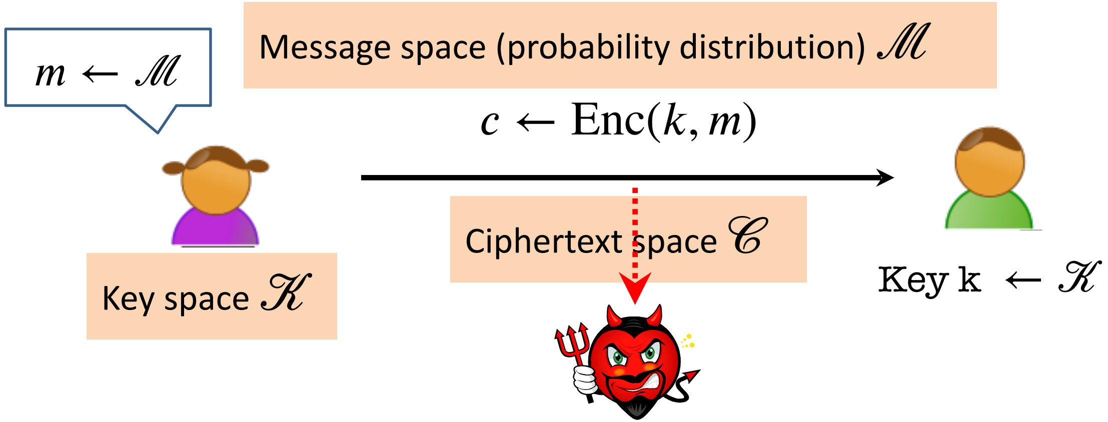
$\text{Enc}(k, m) = m$ would be secure

attempt #2: **attacker cannot recover all of plaintext**

$\text{Enc}(k, (m_1, m_2)) = \text{Enc}(k, m_1) \,||\, m_2$ would be secure

Shannon's idea: **CT should reveal no "info" about PT**

Shannon's Perfect Secrecy Definition



What Eve knows after looking at c

=

What Eve knew before looking at c

$$\forall m \in \mathrm{supp}(\mathscr{M}), \forall c \in \mathscr{C}, M \text{ is a RV } \sim \mathscr{M}$$

$$\underbrace{\Pr[M = m \,|\, \mathrm{Enc}(\mathscr{K}, m) = c]}_{\text{after}} = \underbrace{\Pr[M = m]}_{\text{before}}$$

Shannon's Perfect Secrecy Definition

What Eve knows after looking at c

=

What Eve knew before looking at c

$$\forall m \in \mathsf{supp}(\mathscr{M}), \forall c \in \mathscr{C}, M \text{ is a RV } \sim \mathscr{M}$$

$$\underbrace{\Pr[M = m \,|\, \mathsf{Enc}(\mathscr{K}, m) = c]}_{\text{after}} = \underbrace{\Pr[M = m]}_{\text{before}}$$

✓ **CT reveals no info about PT**

But this def is difficult to work with:

How to prove that ciphertext reveals no info?

Alternate Def: Perfect Indistinguishability

$$\forall m, m' \in \mathrm{supp}(\mathscr{M}), \quad c \in \mathrm{Supp}(\mathscr{C}):$$
$$\Pr[\mathsf{Enc}(\mathscr{K}, m) = c] = \Pr[\mathsf{Enc}(\mathscr{K}, m') = c]$$

World 0:

$k \leftarrow \mathscr{K}$

$c = E(k, m)$

World 1:

$k \leftarrow \mathscr{K}$

$c' = E(k, m')$

is a **distinguisher** that gets c and tries to guess which world she's in

The Two Definitions are Equivalent

THEOREM: An encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ satisfies perfect secrecy IFF it satisfies perfect indistinguishability.

PROOF (next class): Simple use of conditional prob.

Perfect Secrecy is Achievable

The One-time Pad Construction:

Gen: *Choose an n-bit string* k *at random, i.e.* $k \leftarrow \{0,1\}^n$

Enc(k, m) *with* $\mathscr{M} = \{0,1\}^n$*: Output* $c = m \oplus k$

Dec(k, c)*: Output* $m = c \oplus k$

Perfect Secrecy is Achievable

The One-time Pad Construction:

Gen: *Choose an n-bit string k at random, i.e.* $k \leftarrow \{0,1\}^n$

Enc(k, m) *with* $\mathcal{M} = \{0,1\}^n$: *Output* $c = m \oplus k$

Dec(k, c): *Output* $m = c \oplus k$

<u>Correctness</u>: $c \oplus k = m \oplus k \oplus k = m$

Perfect Secrecy is Achievable

The One-time Pad Construction:

Gen: *Choose an n-bit string* k *at random, i.e.* $k \leftarrow \{0,1\}^n$

Enc(k, m) *with* $\mathscr{M} = \{0,1\}^n$*: Output* $c = m \oplus k$

Dec(k, c): *Output* $m = c \oplus k$

Claim: One-time Pad achieves Perfect Indistinguishability (and therefore perfect secrecy).

Proof: For any $m, c \in \{0,1\}^n$,

$$\Pr[\mathsf{Enc}(K, m) = c] = \Pr[k \oplus m = c] = \Pr[k = c \oplus m] = 1/2^n$$

Perfect Secrecy is Achievable

The One-time Pad Construction:

Gen: *Choose an n-bit string k at random, i.e.* $k \leftarrow \{0,1\}^n$

$\text{Enc}(k, m)$ *with* $\mathscr{M} = \{0,1\}^n$: *Output* $c = m \oplus k$

$\text{Dec}(k, c)$: *Output* $m = c \oplus k$

Claim: One-time Pad achieves Perfect Indistinguishability (and therefore perfect secrecy).

Proof: For any $m, m', c \in \{0,1\}^n$

So, $\Pr[\text{Enc}(K, m) = c] = \Pr[\text{Enc}(K, m') = c]$.

QED.

Perfect Secrecy has its Price

THEOREM: For any perfectly secure encryption scheme,

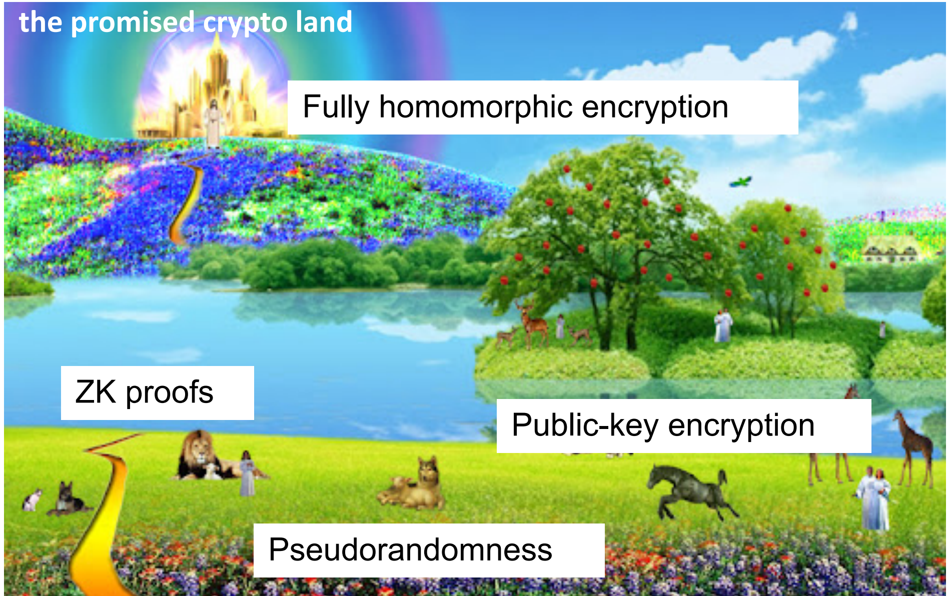
$$|\mathscr{K}| \geq |\mathscr{M}|$$

So, what are we to do?

RELAX the definition:

EVE is an arbitrary *computationally bounded* algorithm.

+ number theory/geometry/combinatorics



To Summarize…

- **Secure Communication**: a quintessential problem in cryptography.
- We saw two equivalent definitions of security:

  Shannon's perfect indistinguishability and perfect secrecy
- **One-time pad achieves perfect secrecy.**
- **A Serious Limitation:** Any perfectly secure encryption scheme needs keys that are at least as long as the messages.
- **Next Lecture: Overcoming the limitation** with Computationally Bounded Adversaries.