CIS 5560

Cryptography
Lecture 6

Course website:

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

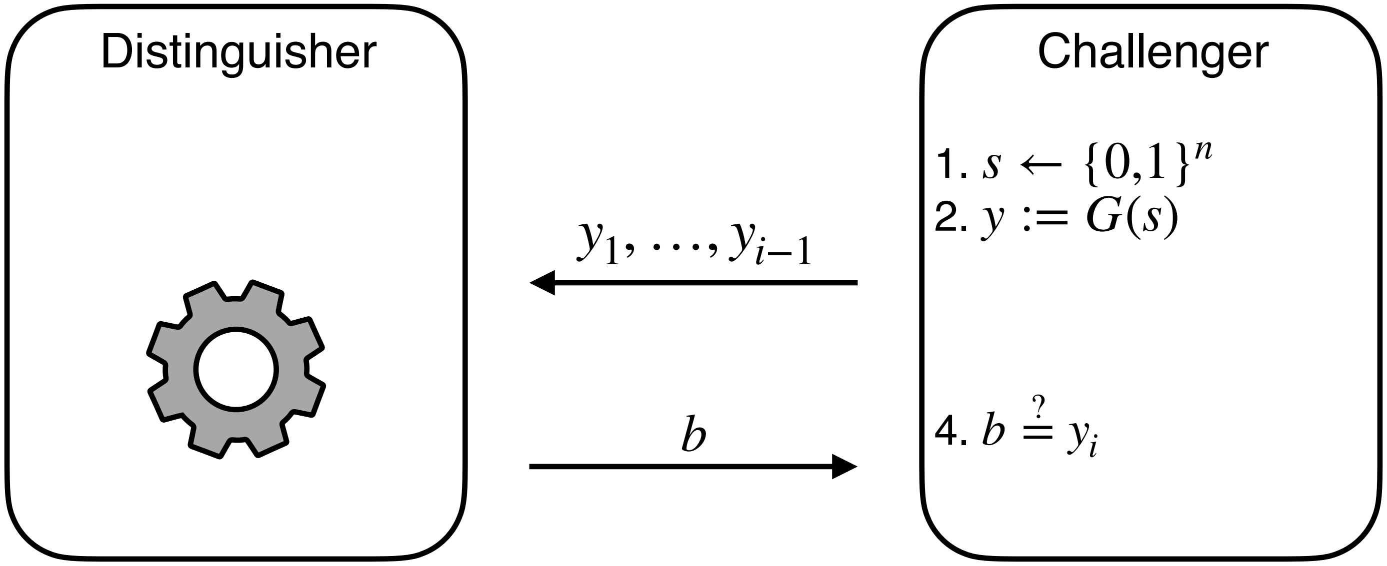
Slides adapted from Dan Boneh and Vinod Vaikuntanathan

Announcements

- **HW 3 out after lecture**
  - Due **Tuesday**, Feb 13 at 1PM on Gradescope
  - Covers PRGs, OWFs, and PRFs

Recap of last lecture

PRG Next-Bit Unpredictability



$$\Pr\left[A(y_1, \ldots, y_{i-1}) = y_i \,\middle|\, \begin{array}{r} s \leftarrow \{0,1\}^n \\ y \leftarrow G(s) \end{array}\right] = 1/2 + \varepsilon(n)$$

Hardcore Bits

HARDCORE PREDICATE

For any $F: \{0,1\}^n \to \{0,1\}^m,\ B: \{0,1\}^n \to \{0,1\}$ is a **hardcore predicate** if for every efficient A, there is a negligible function μ s.t.

$$\Pr\left[b = B(x) \,\middle|\, \begin{array}{c} x \leftarrow \{0,1\}^n \\ b \leftarrow A(F(x)) \end{array} \right] = 1/2 + \mu(n)$$

OWP $\Rightarrow$ PRG

Theorem

Let F be a one-way permutation, and let B be a hardcore predicate for F.

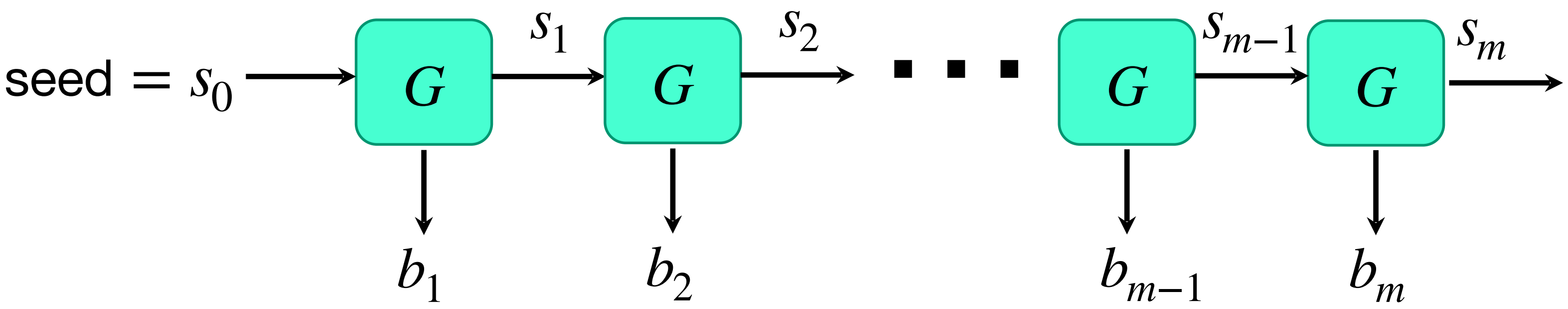
Then, $G(x) := F(x) \,||\, B(x)$ is a PRG.

Length extension: One bit to Many bits

PRG length extension.

Theorem: If there is a PRG G that stretches by one bit, there is one that stretches by many bits

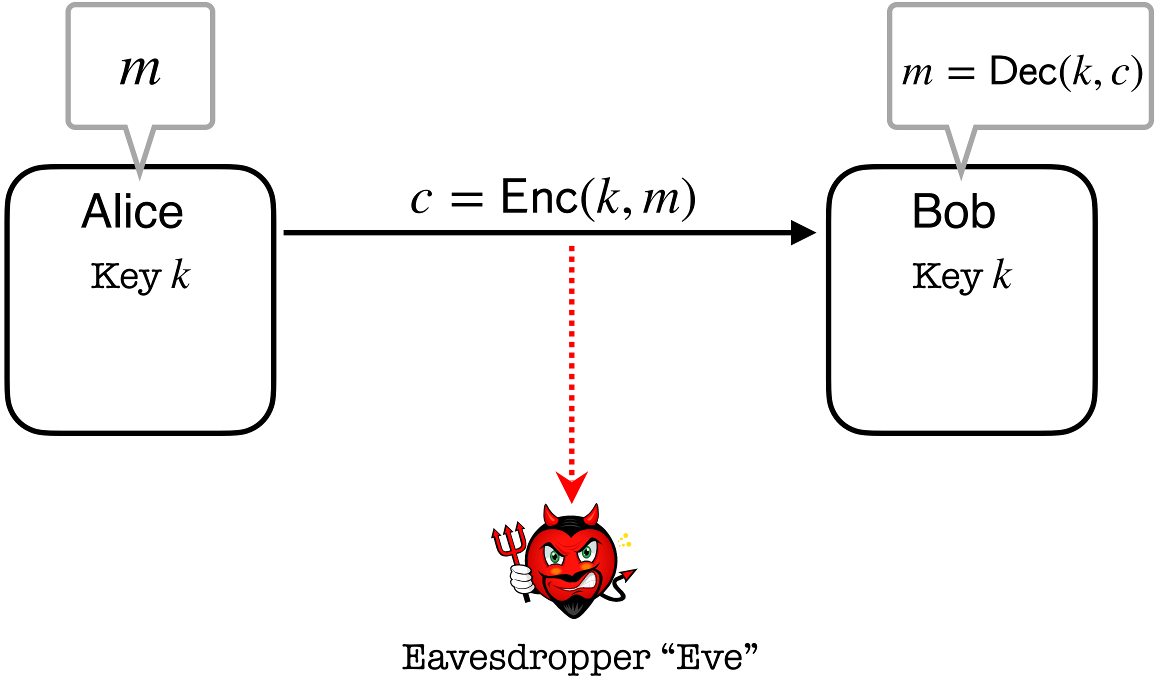
Construction of $G'(s_0)$



Today's Lecture

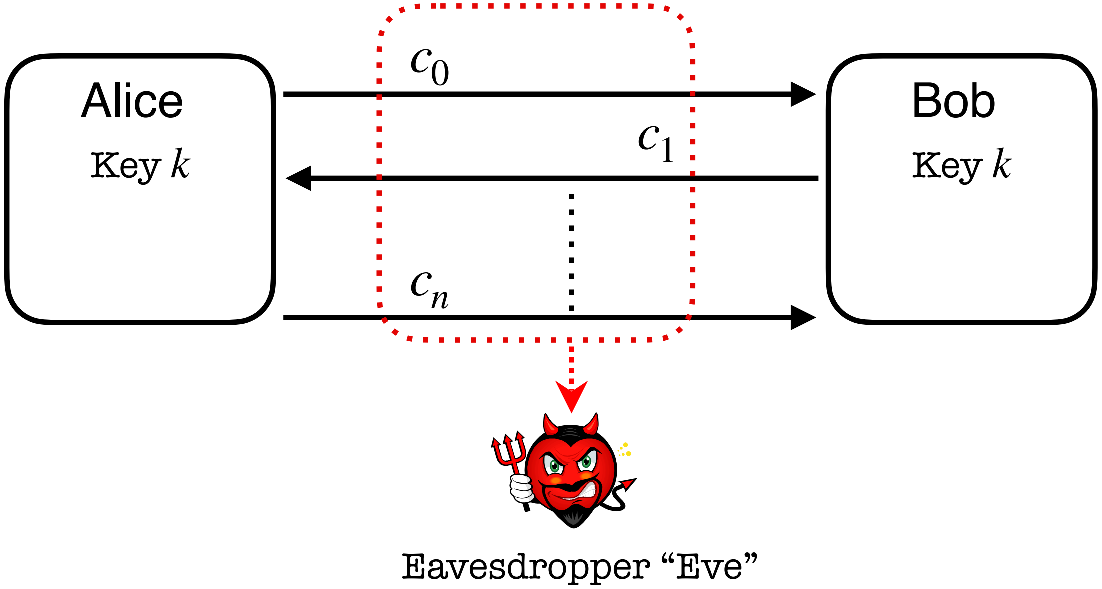
- Encryption for many messages
  - Definition
  - Attempted construction from PRGs
- PRFs
- PRPs
- Block ciphers

So far: Secure Communication for 1 Message



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

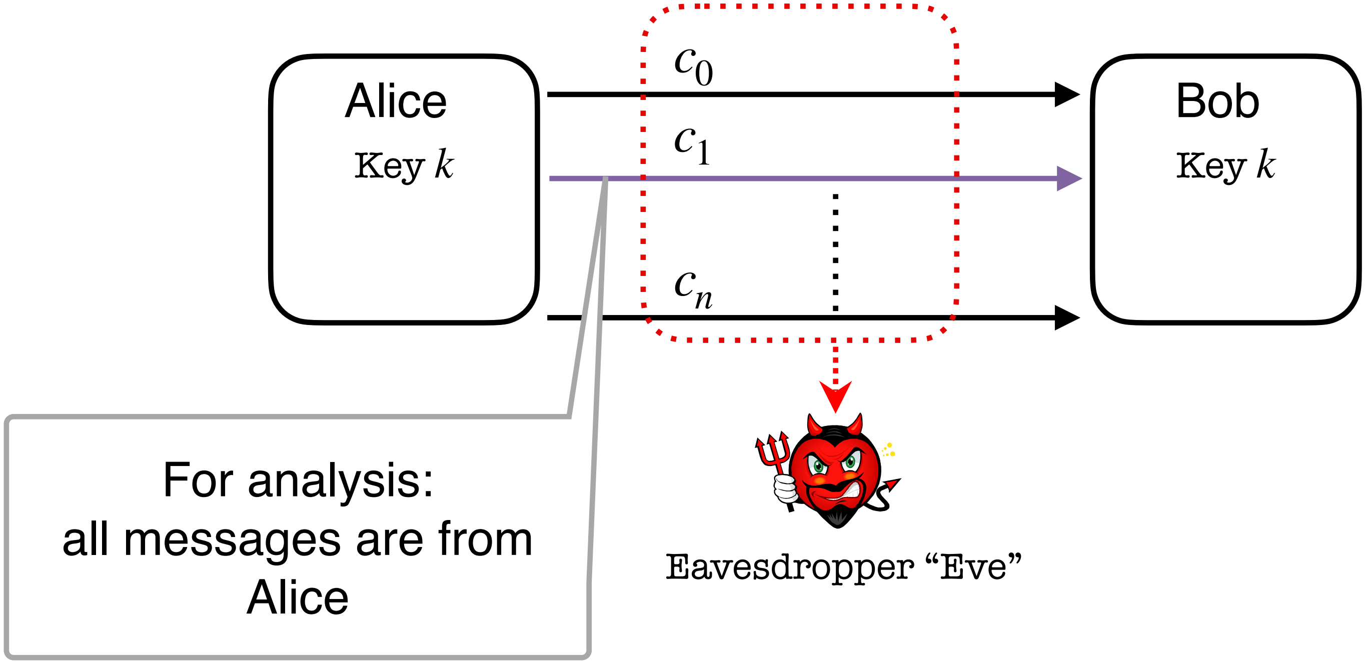
What about a secure *conversation*?



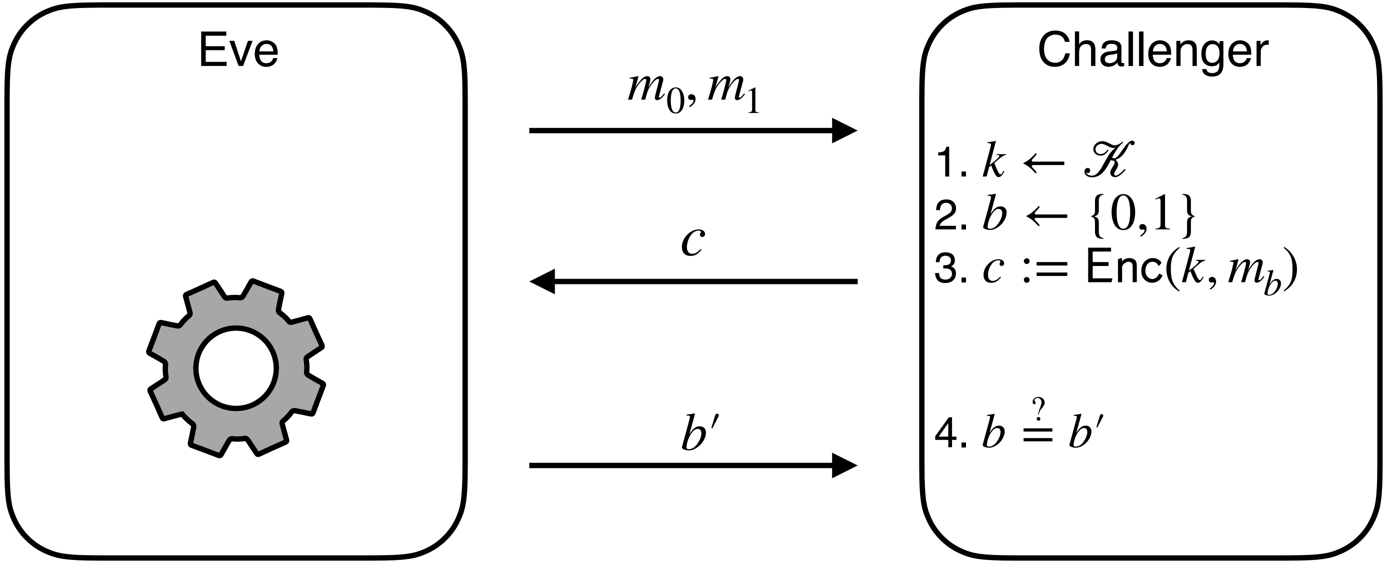
Alice and Bob want to send *many* messages to each other, without revealing *any* of them to Eve.

Requirement: Must use the same key!

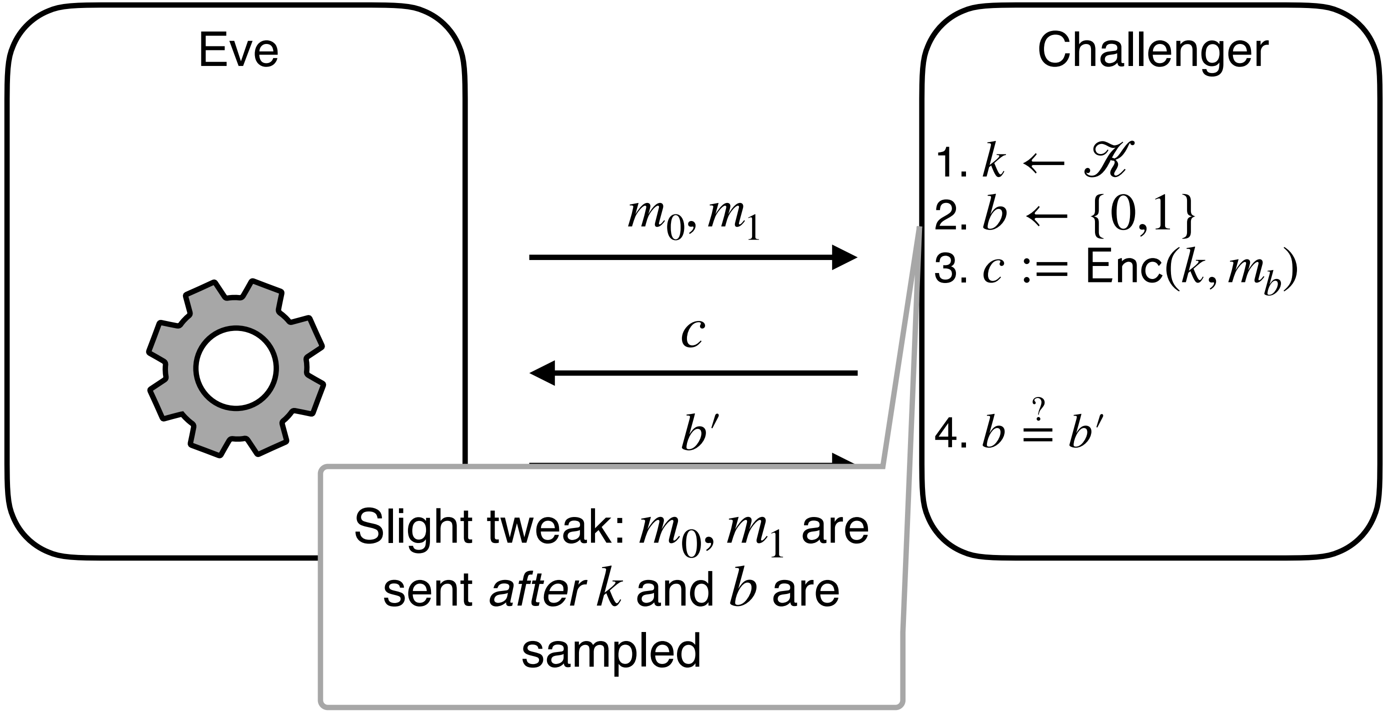
Simplification from Adversarial perspective



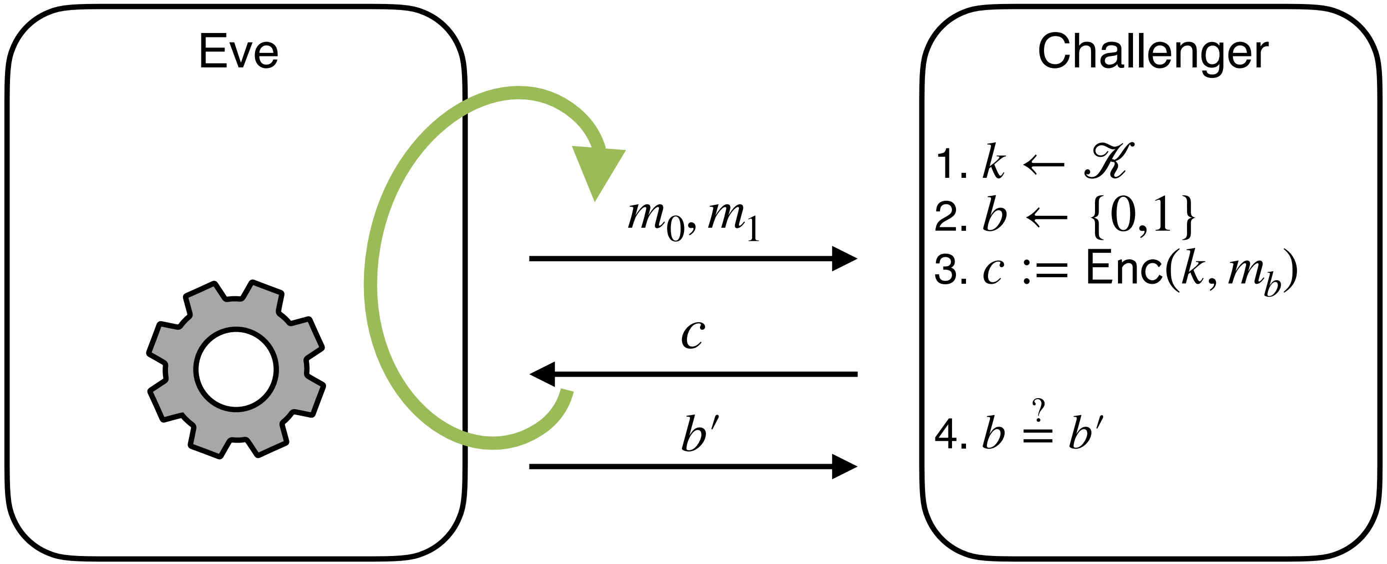
Semantic Security for 1 msg



Semantic Security for 1 msg

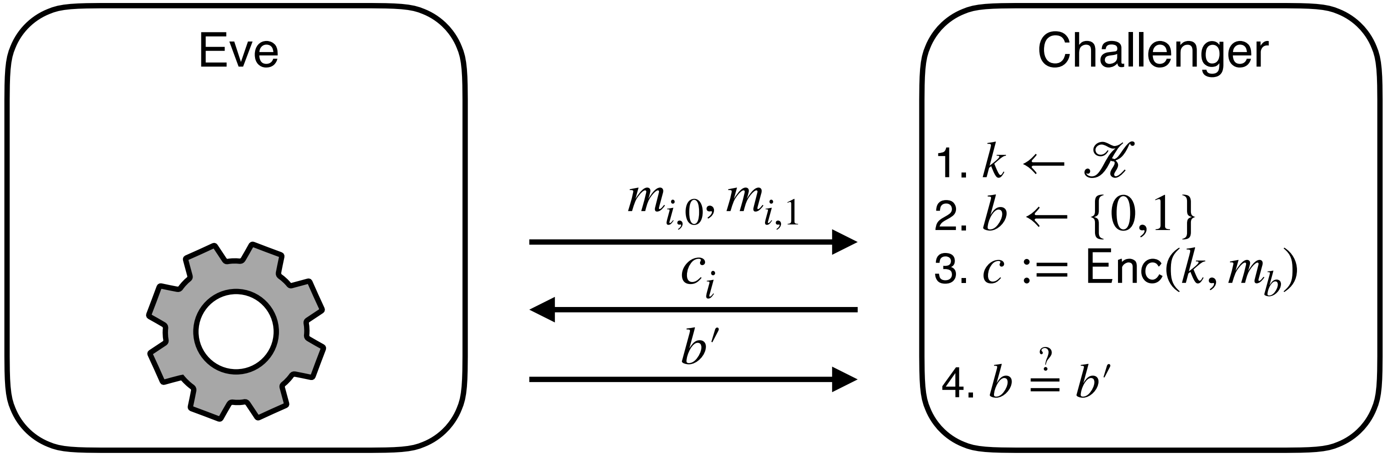


Semantic Security for many msgs?



Repeat experiment many times!

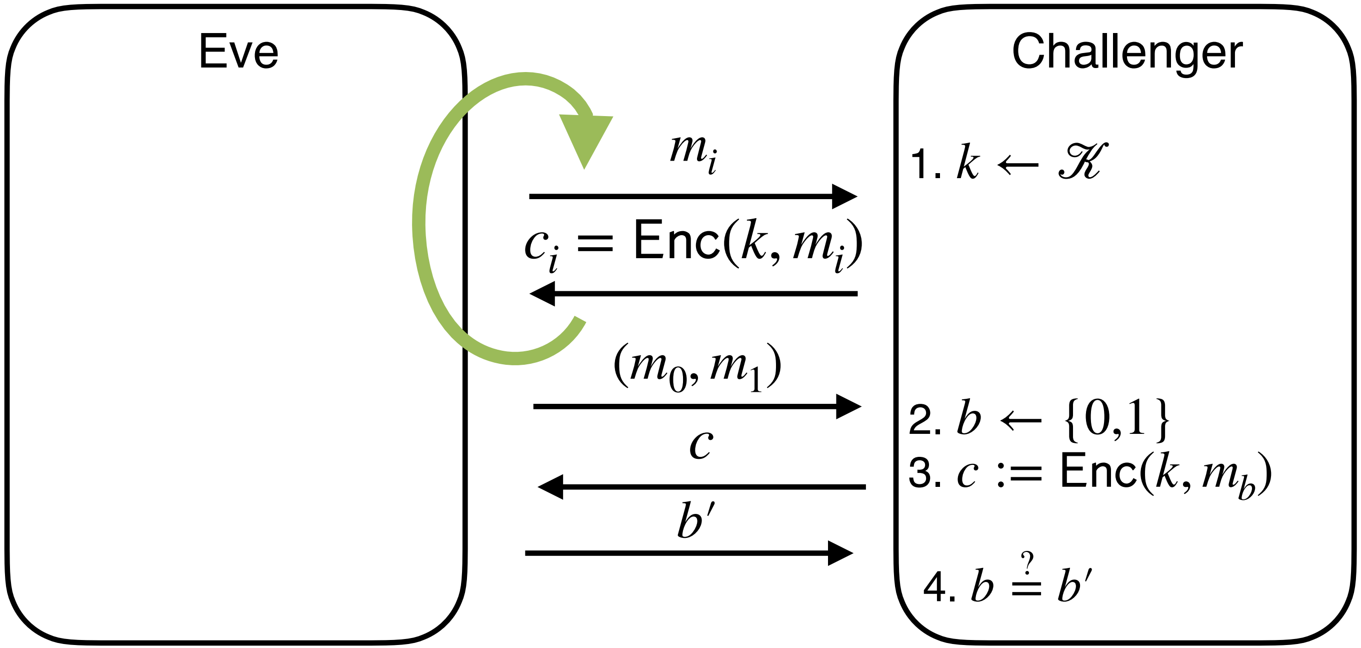
Semantic Security for Many Msgs



For every **PPT** Eve, there exists a negligible fn ε,

$$\Pr\left[\mathsf{Eve}(c_q) = b \;\middle|\; \begin{array}{r} k \leftarrow \mathcal{K} \\ b \leftarrow \{0,1\} \\ \text{For } i \text{ in } 1,\ldots,q: \\ (m_{i,0}, m_{i,1}) \leftarrow \mathsf{Eve}(c_{i-1}) \\ c_i = \mathsf{Enc}(k, m_{i,b}) \end{array} \right] < \frac{1}{2} + \varepsilon(n)$$

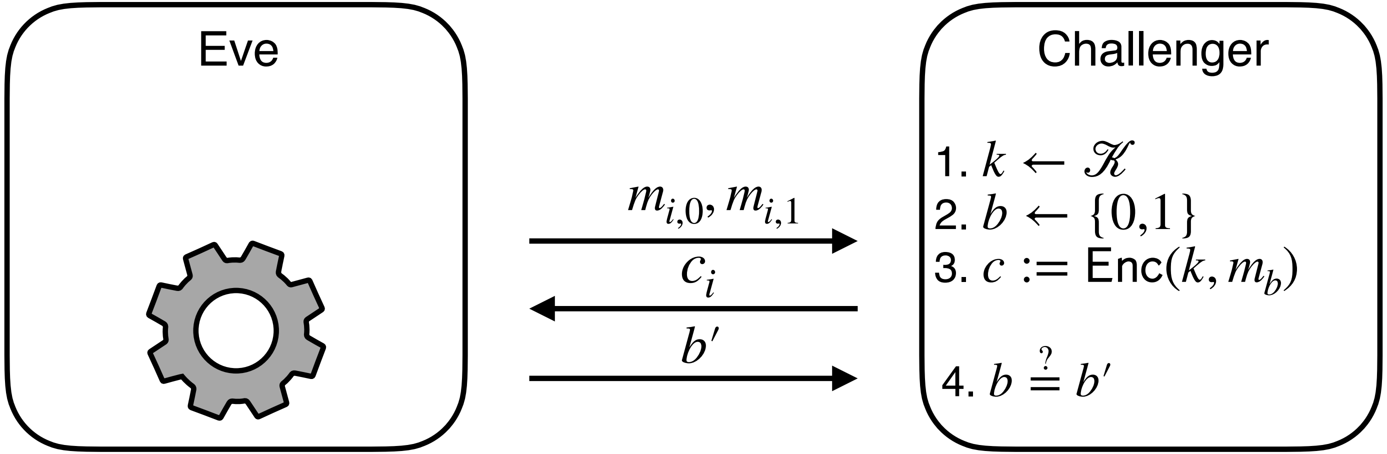
Alternate (Stronger?) definition



Also called “IND-CPA”: Indistinguishability under Chosen-Plaintext Attacks

Equivalent to previous definition: just set $m_{i,0} = m_{i,1} = m_i$

Semantic Security for Many Msgs



For every **PPT** Eve and q, there exists a negligible fn ε, such that

$$\Pr\left[\mathsf{Eve}(c_q) = b \;\middle|\; \begin{array}{r} k \leftarrow \mathscr{K} \\ b \leftarrow \{0,1\} \\ \text{For } i \text{ in } 1,\ldots,q: \\ (m_{i,0}, m_{i,1}) \leftarrow \mathsf{Eve}(c_{i-1}) \\ c_i = \mathsf{Enc}(k, m_{i,b}) \end{array} \right] < \frac{1}{2} + \varepsilon(n)$$

Construction Attempt #1: Stream Ciphers

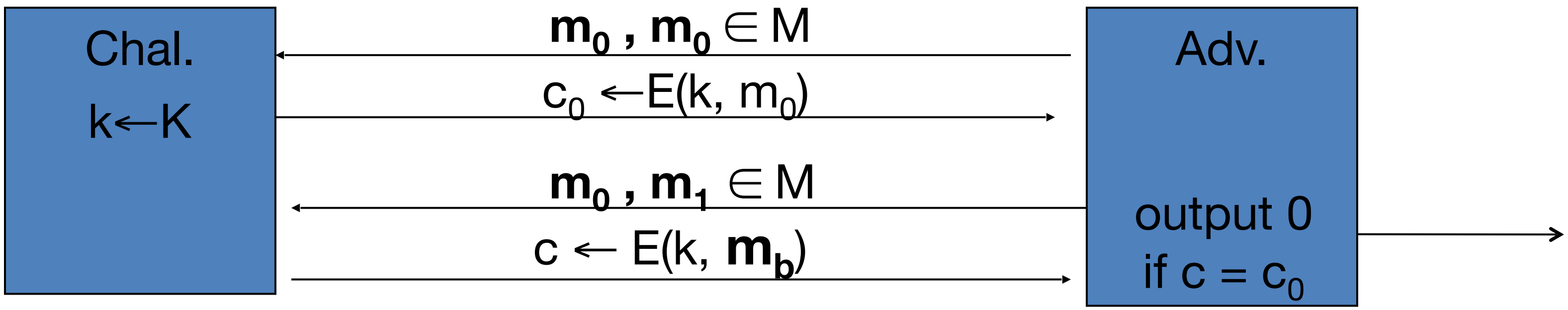
- Gen$(1^k) \rightarrow k$:
  - Sample an n-bit string at random.
- Enc$(k, m) \rightarrow c$:
  - Expand k to an $m(n)$-bit string using PRG: $s = G(k)$
  - Output $c = s \oplus m$
- Dec$(k, c) \rightarrow m$:
  - Expand k to an $m(n)$-bit string using PRG: $s = G(k)$
  - Output $m = s \oplus c$

Is this secure?

Stream Ciphers insecure under CPA

Problem: E(k,m) outputs same ciphertext for msg m.

Then:



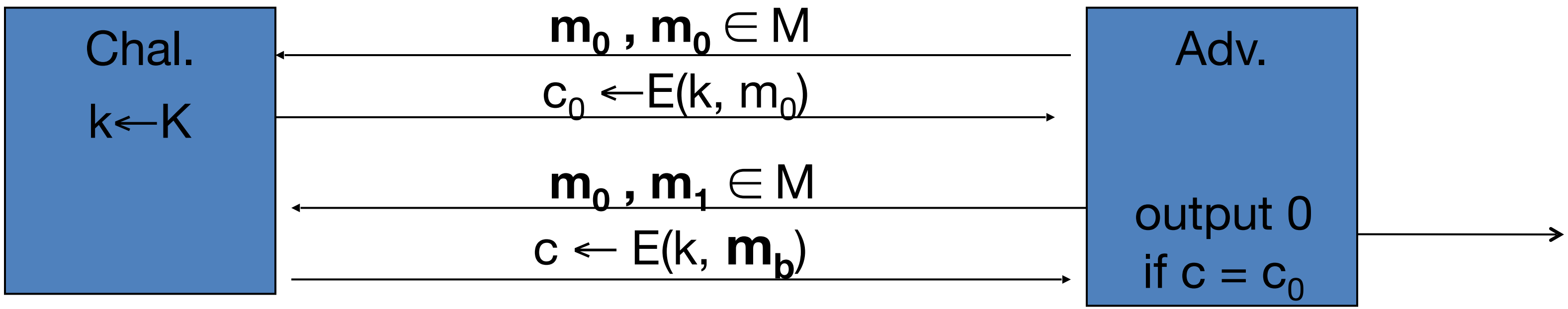
So what? an attacker can learn that two encrypted files are the same, two encrypted packets are the same, etc.

- Leads to significant attacks when message space M is small

Stream Ciphers insecure under CPA

Problem: E(k,m) always outputs same ciphertext for msg m.

Then:



If secret key is to be used multiple times $\Rightarrow$

given the same plaintext message twice, encryption must produce different outputs.

Ideas for multi-message encryption

- State? (e.g. counter of num msgs)
- Randomness?

Approach 1: Stateful encryption

- $\mathsf{Gen}(1^n) \to k$:
  - Sample an n-bit string at random.

- $\mathsf{Enc}(k, m, \mathbf{st}) \to c$:
  1. Interpret $\mathbf{st}$ as number ℓ of messages encrypted so far.
  2. Run PRG: $s = G(k)$
  3. Discard first ℓ bits of s to get s'
  4. Set $\ell := \ell + 1$
  5. Output $c = s' \oplus m$

- $\mathsf{Dec}(k, c, \mathbf{st}) \to m$:
  - Repeat steps 1 through 4 of Enc
  - Output $m = s' \oplus c$

Does this work?

Ans: Yes!

Exercise: reduce to PRG security

Pros:

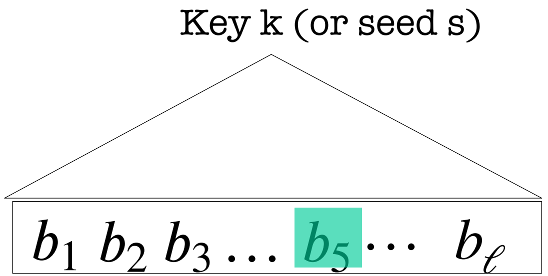
- Relies on existing tools
- Generally fast

Cons:

- Must maintain counter of encrypted messages
- Must rerun PRG from start every time
- Sequential encryption/decryption

Problem: PRGs are sequential

PRG $G(k)$



- With a PRG, accessing the ℓ-th bit takes time ℓ.
- How to get efficient *random access* into output?
- That is, we want some function such that $F(\ell) = \ell$-th bit

New tool:

Pseudorandom
Function

Pseudorandom Functions

Collection of functions $\mathcal{F}_\ell = \{F_k : \{0,1\}^\ell \rightarrow \{0,1\}^m\}_{k \in \{0,1\}^n}$

- indexed by a key k
- n: key length, ℓ: input length, m: output length.
- Independent parameters, all poly(sec-param) = poly(n)
- #functions in $\mathcal{F}_\ell \leq 2^n$ (singly exponential in n)

$\mathbf{Gen}(1^n)$: Generate a random n-bit key k.

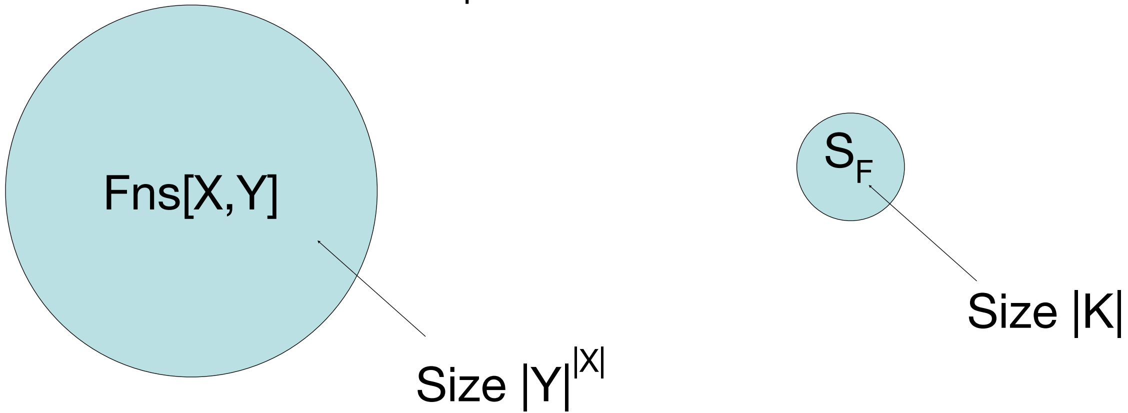
$\mathbf{Eval}(k, x)$ is a poly-time algorithm that outputs $F_k(x)$

Secure PRFs

- Let F: $K \times X \rightarrow Y$ be a PRF

$$\begin{cases} \text{Fns}[X,Y]: & \text{the set of } \underline{\textbf{all}} \text{ functions from } X \text{ to } Y \\ S_F = \{\ F(k,\cdot)\ \ \text{s.t.}\ \ k \in K\ \} & \subseteq\ \ \text{Funs}[X,Y] \end{cases}$$

- <u>Intuition</u>: a PRF is **secure** if
  a random function in Funs[X,Y] is indistinguishable from
  a random function in S_F



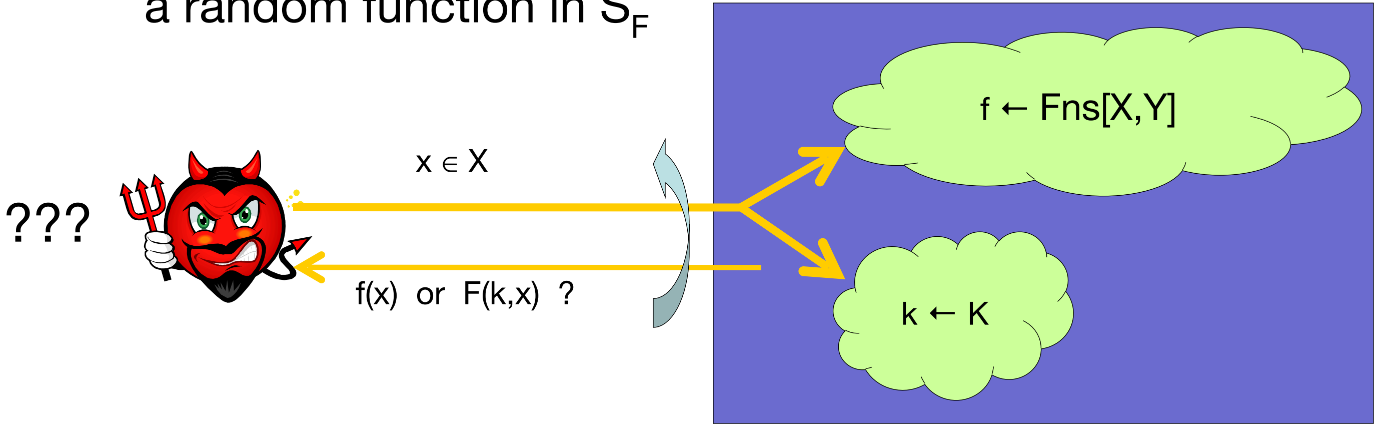
Secure PRFs

- Let $F:\ K \times X \rightarrow Y$ be a PRF

$$\begin{cases} \text{Fns[X,Y]:} & \text{the set of } \underline{\textbf{all}} \text{ functions from X to Y} \\ S_F = \{\ F(k,\cdot)\ \ \text{s.t.}\ \ k \in K\ \} & \subseteq \quad \text{Funs[X,Y]} \end{cases}$$

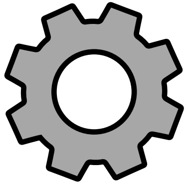
- Intuition: a PRF is **secure** if

  a random function in Fns[X,Y] is indistinguishable from
  a random function in S_F



PRF Security

Adv $\mathscr{A}$



x →

← y

b' →

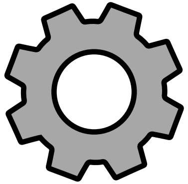
Challenger

1. $b \leftarrow \{0,1\}$
2. If b = 0
    1. Sample $f \leftarrow \mathsf{Fns}[X, Y]$
    2. Set $y := f(x)$
3. If b = 1
    1. Sample $k \leftarrow \mathscr{K}$
    2. Set $y := F_k(x)$
4. $b \stackrel{?}{=} b'$

$$\Pr[b = b'] = 1/2 + \mathsf{negl}(n)$$

PRF Security (Advantage defn)

Adv $\mathscr{A}$



x →

← y

b' →

Challenger

1. $b \leftarrow \{0,1\}$
2. If b = 0
    1. Sample $f \leftarrow$ Fns$[X, Y]$
    2. Set $y := f(x)$
3. If b = 1
    1. Sample $k \leftarrow \mathscr{K}$
    2. Set $y := F_k(x)$
4. $b \stackrel{?}{=} b'$

$$\left| \Pr[b' = 1 \,|\, b = 0] - \Pr[b' = 1 \,|\, b = 1] \right| = \mathsf{negl}(n)$$

An example

Let $K = X = \{0,1\}^n$.

Consider the PRF: $\boxed{F(k, x) = k \oplus x}$ defined over (K, X, X)

Let's show that F is insecure:

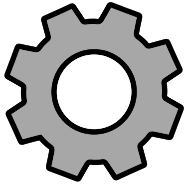
Adversary $\mathscr{A}$:
(1) choose arbitrary $x_0 \neq x_1 \in X$
(2) query for $y_0 = f(x_0)$ and $y_1 = f(x_1)$
(3) output `0' if $y_0 \oplus y_1 = x_0 \oplus x_1$ , else `1'

$$\Pr[EXP(0) = 0] = 1 \qquad \Pr[EXP(1) = 0] = 1/2^n$$

$$\Longrightarrow \quad Adv_{PRF}[\mathscr{A}, F] = 1 - (1/2n) \qquad \text{(not negligible)}$$

PRP Security

Adv $\mathscr{A}$



$x \longrightarrow$

$y \longleftarrow$

$b' \longrightarrow$

Challenger

1. $b \leftarrow \{0,1\}$
2. If b = 0
    1. Sample $f \leftarrow$ Perms$[X]$
    2. Set $y := f(x)$
3. If b = 1
    1. Sample $k \leftarrow \mathscr{K}$
    2. Set $y := F_k(x)$
4. $b \stackrel{?}{=} b'$

$$\Pr[b = b'] = 1/2 + \mathsf{negl}(n)$$

PRFs → multi-message encryption

Ideas for multi-message encryption

- State? (e.g. counter of num msgs)
- Randomness?